

**IBM DB2 10.1  
for Linux, UNIX, and Windows**

## **Text Search ガイド**





**IBM DB2 10.1  
for Linux, UNIX, and Windows**

## **Text Search ガイド**



**ご注意**

本書および本書で紹介する製品をご使用になる前に、275 ページの『付録 E. 特記事項』に記載されている情報をお読みください。

本書には、IBM の専有情報が含まれています。その情報は、使用許諾条件に基づき提供され、著作権により保護されています。本書に記載される情報には、いかなる製品の保証も含まれていません。また、本書で提供されるいかなる記述も、製品保証として解釈すべきではありません。

IBM 資料は、オンラインでご注文いただくことも、ご自分の国または地域の IBM 担当員を通してお求めいただくこともできます。

- オンラインで資料を注文するには、IBM Publications Center (<http://www.ibm.com/shop/publications/order>) をご利用ください。
- ご自分の国または地域の IBM 担当員を見つけるには、IBM Directory of Worldwide Contacts (<http://www.ibm.com/planetwide/>) をお調べください。

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。

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

原典： SC27-3888-00  
IBM DB2 10.1 for Linux, UNIX, and Windows  
Text Search Guide

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・サービス・センター

第1刷 2012.4

© Copyright IBM Corporation 2008, 2012.

# 目次

## 第 1 章 DB2 Text Search . . . . . 1

## 第 2 章 DB2 Text Search の鍵となるフ ィーチャーと概念 . . . . . 3

DB2 Text Search サーバーのデプロイメント・シナリ オ . . . . .	4
テキスト検索索引の作成、更新、およびプロパティ の変更 . . . . .	7
パーティション・データベース環境での DB2 Text Search . . . . .	10
DB2 Text Search 索引のインクリメンタル更新 . . . . .	13
DB2 Text Search の言語学的処理 . . . . .	14
シナリオ: 索引付けおよび検索 . . . . .	16
リッチ・テキストおよび専用フォーマットのサポー ト . . . . .	19

## 第 3 章 テキスト検索ソリューションの計 画 . . . . . 21

文書の特徴 . . . . .	21
DB2 Text Search でサポートされる文書フォー マット . . . . .	21
サポートされるデータ・タイプ . . . . .	21
サポートされない形式とデータ・タイプの変換 . . . . .	21
サポートされる言語およびコード・ページ . . . . .	22
文書のサイズに関する考慮事項 . . . . .	22
DB2 Text Search セキュリティーの概要 . . . . .	23
ユーザー・ロール . . . . .	25
アクセス・ポリシーと通信セキュリティー . . . . .	26
DB2 Text Search のキャパシティー・プランニング および最適化 . . . . .	27
DB2 Text Search サーバー構成 . . . . .	28
DB2 Text Search 索引の計画および最適化 . . . . .	32
DB2 Text Search システムのチューニング . . . . .	37
DB2 Text Search 照会の計画 . . . . .	39
DB2 Text Search の引数 . . . . .	39
DB2 Text Search における複数の述部 . . . . .	40
DB2 Text Search のロケールおよび言語 . . . . .	40
DB2 Text Search の SCORE 関数 . . . . .	41
DB2 Text Search の RESULTLIMIT 関数 . . . . .	41
DB2 Text Search のパーサー構成 . . . . .	42
DB2 Text Search の XML 名前空間 . . . . .	43

## 第 4 章 DB2 Text Search のインストー ルおよび構成 . . . . . 45

DB2 Text Search のハードウェアおよびソフトウェ アの要件 . . . . .	48
デフォルト構成の DB2 Text Search のインストー ル . . . . .	50
DB2 セットアップ・ウィザードによる DB2 Text Search のインストールおよび構成 . . . . .	50

応答ファイルによる DB2 Text Search のインスト ールおよび構成 . . . . .	50
db2_install を使用した DB2 Text Search のイン ストール (Linux および UNIX) . . . . .	51
初期構成なしの DB2 Text Search のインストー ル . . . . .	52
DB2 セットアップ・ウィザードを使用した DB2 データベース・サーバーのインストール (Windows) . . . . .	52
DB2 セットアップ・ウィザードによる DB2 サー バーのインストール (Linux および UNIX) . . . . .	55
DB2 の応答ファイル・インストールの概要 (Windows) . . . . .	59
DB2 の応答ファイル・インストールの概要 (Linux および UNIX) . . . . .	60
スタンドアロン・テキスト検索サーバーのインスト ールおよび構成 . . . . .	61
スタンドアロン・サーバーのインストール・スペ ース所要量 . . . . .	61
スタンドアロン DB2 Text Search サーバーのイン ストール . . . . .	61
スタンドアロン・サーバーを Windows サービス としてインストールおよび構成する . . . . .	63
スタンドアロン DB2 Text Search サーバーのアン インストール . . . . .	63

## 第 5 章 DB2 Text Search の構成 . . . . . 65

統合 DB2 Text Search サーバーの初期構成 . . . . .	67
DB2 Text Search サーバー情報の更新 . . . . .	69
スタンドアロン DB2 Text Search サーバーの構成 . . . . .	70
TCP/IP 通信のためのサーバー上のサービス・ファ イルの更新 . . . . .	72
DB2 Accessories Suite for DB2 Text Search のイン ストール . . . . .	72
DB2 Accessories Suite for DB2 Text Search のアン インストール . . . . .	74

## 第 6 章 DB2 Text Search のアップグレ ード . . . . . 77

管理者または root インストール用の DB2 Text Search のアップグレード . . . . .	77
非 root インストールの DB2 Text Search のアップ グレード (Linux および UNIX) . . . . .	81
DB2 Text Search を使用しない複数パーティショ ン・インスタンスのアップグレード . . . . .	82
スタンドアロン DB2 Text Search サーバーのアップ グレード . . . . .	83

## 第 7 章 テキスト検索索引の構成と管理 85

DB2 Text Search 用のコマンド行ツール . . . . .	85
テキスト検索コマンドの実行 . . . . .	85

リッチ・テキストおよび専有フォーマットのサポート	86
DB2 Text Search でのリッチ・テキスト文書のサポートを有効にする	86
リッチ・テキストおよび専有フォーマットのサポートの無効化	87
DB2 Text Search インスタンス・サービスの開始	88
DB2 Text Search インスタンス・サービスの停止	88
データベースで DB2 Text Search を使用可能にする	89
DB2 Text Search のデータベースを使用不可にする	90
孤立した DB2 Text Search コレクションの削除	91
DB2 Text Search 用の同義語ディクショナリー	93
DB2 Text Search の同義語ディクショナリーの追加	93
DB2 Text Search の同義語ディクショナリーの除去	94
テキスト検索索引の作成	94
テキスト検索索引の作成	95
テキスト検索索引の保守	102
テキスト検索索引の更新	104
テキスト検索索引イベントの消去	108
テキスト検索索引の変更	108
テキスト検索索引状況の表示	109
DB2 Text Search コレクションのロケーションの変更	109
テキスト検索索引のバックアップとリストア	110
テキスト検索索引のドロップ	111
例: DB2 Text Search 索引更新のスケジューリング	112

## 第 8 章 テキスト検索索引を使用する検索 . . . . . 115

DB2 Text Search の検索関数	115
フルテキスト検索の方法	116
基本検索	117
ファジー検索	117
近接検索	118
特殊文字の検索	119
XML 文書での構造的フルテキスト検索	122
SCORE を使用してテキスト検索索引を検索する	124
DB2 Text Search 引数の構文	125
XML 文書用の検索構文	129
フルテキスト照会のパフォーマンスの強化	134

## 第 9 章 SQL および XML の組み込み検索関数 . . . . . 135

CONTAINS 関数	135
SCORE 関数	138
xmlcolumn-contains 関数	140

## 第 10 章 DB2 Text Search 用の管理コマンド . . . . . 145

db2ts ALTER INDEX	146
db2ts CLEANUP FOR TEXT	151
db2ts CLEAR COMMAND LOCKS	152

db2ts CLEAR EVENTS FOR TEXT	153
db2ts CREATE INDEX	155
db2ts DISABLE DATABASE FOR TEXT	163
db2ts DROP INDEX	164
db2ts ENABLE DATABASE FOR TEXT	166
db2ts HELP	168
db2ts RESET PENDING コマンド	170
db2ts START FOR TEXT	171
db2ts STOP FOR TEXT	173
db2ts UPDATE INDEX	174

## 第 11 章 DB2 Text Search のストアド・プロシージャ . . . . . 179

SYSTS_ADMIN_CMD プロシージャ - テキスト検索管理コマンドの実行	179
SYSTS_ALTER プロシージャ - 索引の更新特性の変更	181
SYSTS_CLEANUP プロシージャ - 無効なテキスト検索索引の削除	187
SYSTS_CLEAR_COMMANDLOCKS プロシージャ - テキスト検索索引のコマンド・ロックの削除	189
SYSTS_CLEAR_EVENTS プロシージャ - 索引のイベント表からの索引付けイベントの削除	191
SYSTS_CONFIGURE プロシージャ - 現行データベースをテキスト検索用に構成する	193
SYSTS_CREATE プロシージャ - 列でのテキスト検索索引の作成	196
SYSTS_DISABLE プロシージャ - テキスト検索の現行データベースを使用不可にする	205
SYSTS_DROP プロシージャ - テキスト検索索引のドロップ	207
SYSTS_ENABLE プロシージャ - 現在のデータベースをテキスト検索用に使用可能にする	210
SYSTS_UPDATE プロシージャ - テキスト検索索引の更新	212
SYSTS_UPGRADE_CATALOG プロシージャ - テキスト検索カタログのアップグレード	215
SYSTS_UPGRADE_INDEX - テキスト検索索引のアップグレード	217

## 第 12 章 テキスト検索の管理ビュー . . . . . 221

SYSIBMTS.TSDEFAULTS ビュー	221
SYSIBMTS.TSLOCKS ビュー	222
SYSIBMTS.TSSERVERS ビュー	223
SYSIBMTS.TSINDEXES ビュー	224
SYSIBMTS.TSCONFIGURATION ビュー	226
SYSIBMTS.TSCOLLECTIONNAMES ビュー	226
SYSIBMTS.TSEVENT ビュー	226
SYSIBMTS.TSSTAGING ビュー	228

## 付録 A. DB2 Text Search および Net Search Extender の比較 . . . . . 229

## 付録 B. DB2 Text Search でサポートされるロケール . . . . . 235

**付録 C. DB2 コマンド . . . . . 237**

db2iupgrade - インスタンスのアップグレード . . . . . 237  
db2icrt - インスタンスの作成 . . . . . 240  
db2idrop - インスタンスの除去 . . . . . 250  
db2iupdt - インスタンスの更新 . . . . . 253

**付録 D. DB2 技術情報の概説 . . . . . 263**

DB2 テクニカル・ライブラリー (ハードコピーまたは PDF 形式) . . . . . 264  
コマンド行プロセッサから SQL 状態ヘルプを表示する . . . . . 266  
異なるバージョンの DB2 インフォメーション・センターへのアクセス . . . . . 267

コンピューターまたはイントラネット・サーバーにインストールされた DB2 インフォメーション・センターの更新 . . . . . 267  
コンピューターまたはイントラネット・サーバーにインストールされた DB2 インフォメーション・センターの手動更新 . . . . . 269  
DB2 チュートリアル . . . . . 271  
DB2 トラブルシューティング情報 . . . . . 271  
ご利用条件 . . . . . 272

**付録 E. 特記事項 . . . . . 275**

**索引 . . . . . 279**





## 第 1 章 DB2 Text Search

DB2® Text Search では、DB2 データベースに保管されたデータに対してテキスト検索照会を実行する SQL ステートメントおよび XQuery ステートメントを実行することで、テキスト列を検索することができます。

DB2 Text Search は、DB2 表に保管されたテキスト列のデータを検索するための拡張機能を提供します。検索システムの照会応答は高速であり、統合されたランク付きの結果セットを提供するので、必要な情報を素早く簡単に見つけることができます。DB2 Text Search の関数を SQL および XQuery ステートメントに取り込むことによって、強力で用途の広いテキスト検索プログラムを作成できます。さらに、検索エンジンは言語学的な分析を使用して、関連する検索照会の結果だけが戻されるようにします。テキスト検索サポートを使用可能にすると、DB2 エンジンに組み込まれた CONTAINS、SCORE、および xmlcolumn-contains 関数を使用して、指定する検索指数に基づいてテキスト検索索引を検索できます。

DB2 Text Search は、検索中のリソース使用量が大きくなるようにするためのデータ・ストリームを採用することによって、高いハイパフォーマンスおよびスケラビリティを実現します。

DB2 Text Search サーバーと DB2 データベース・サーバーを、統合テキスト検索サーバー・セットアップ用に同じシステム上にインストールすることができます。また、DB2 Text Search サーバーと DB2 データベース・サーバーを、スタンドアロン・セットアップ用に異なるシステム上にインストールすることもできます。DB2 Text Search サーバーはそれ自体の Java 仮想マシン (JVM) 内で実行します。DB2 インスタンスを開始した後、DB2 Text Search のサービスは明示的に開始および停止します。

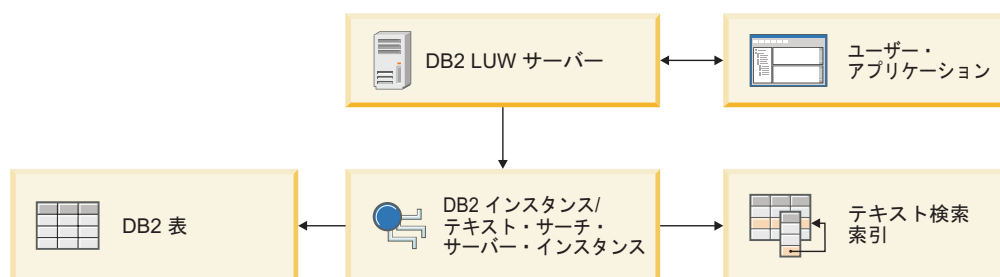


図 1. 統合 DB2 Text Search サーバーのデプロイメント図

DB2 Text Search には、グラフィカル・ユーザー・インターフェースがありません。その代わりに、DB2 Text Search サーバーの構成と管理、コレクション用の同義語ディクショナリーの作成、問題の診断などのタスクのために、コマンド行ツールを使用できます。さらに、さまざまな一般的な管理タスクでストアード・プロシージャ・インターフェースを使用できます。

DB2 Text Search 索引を作成および更新し、索引を使用できる準備ができたなら索引状況を切り替えることで、Net Search Extender から DB2 Text Search にテキスト索

引をマイグレーションすることができます。詳しくは、Net Search Extender から DB2 Text Search へのマイグレーションに関するトピックを参照してください。

DB2 Text Search の主要なフィーチャーは以下のとおりです。

#### **DB2 Database for Linux, UNIX, and Windows との緊密な統合**

- 管理コマンド用のストアード・プロシージャ・インターフェース
- DB2 インストーラーによって実行されるインストールおよび構成
- 不可視の認証
- エラー処理用の SQL コード

#### **文書索引作成**

- 大量のデータに対する高速の索引付け
- pureXML<sup>®</sup> サポート
- 複数の文書形式のサポート
- インクリメンタルで非同期の索引更新

#### **拡張検索テクノロジー**

- SQL、SQL/XML、および XQuery のサポート
- CONTAINS および SCORE SQL 関数
- DB2 オプティマイザーと結合した、組み込み SQL 機能
- xmlcolumn-contains XML 関数
- XML フィルター処理
- サポートされるすべての言語での言語学的処理
- 重み、ワイルドカード、およびオプション用語のサポート
- 同義語ディクショナリーのサポート

## 第 2 章 DB2 Text Search の鍵となるフィーチャーと概念

DB2 Text Search では、DB2 データベースの表列に保管されているテキスト文書を検索するための迅速かつ汎用的な方法が提供されます。文書の検索は、XML 文書での検索用の XQuery または SQL 照会を使用して行うことができます。

テキスト文書は、一意に識別可能でなければなりません。この目的のために、DB2 Text Search では表の主キーを使用します。

DB2 Text Search では、テキスト文書を順次検索するのではなく、より効率的な方法であるテキスト検索索引を使用して検索を行います。テキスト検索索引は、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。

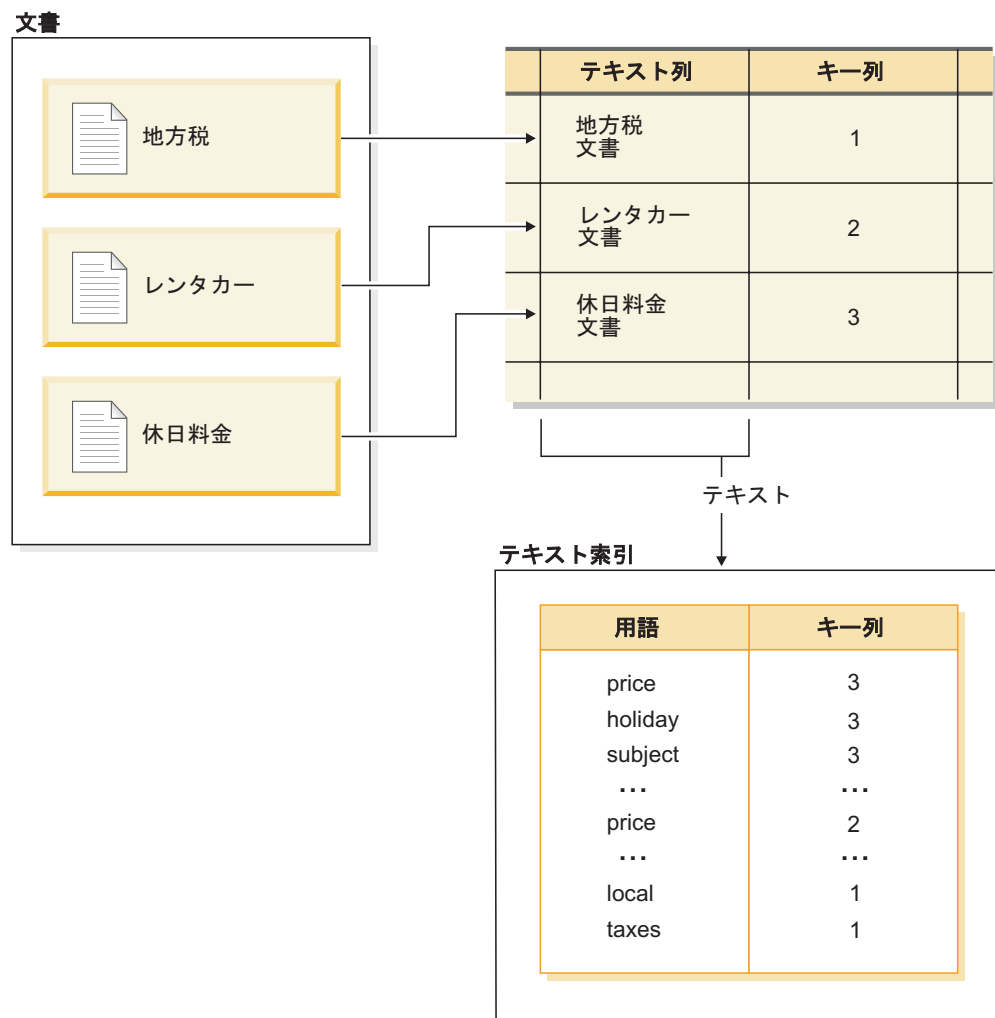


図 2. テキスト検索索引の作成

テキスト検索索引を作成すると、更新頻度などの、索引のプロパティが定義されます。作成直後のテキスト検索索引にはデータは含まれていません。索引を更新す

ると、用語やテキスト文書に関するデータがテキスト検索索引に追加されます。最初の索引更新によって、テキスト列からすべてのテキスト文書が索引に追加されます。それ以降の更新はインクリメンタル更新と呼ばれ、表内のデータとテキスト検索索引内のデータの同期化を行います。DB2 Text Search では、テキスト検索索引とその表との同期化を行う以下の 2 つの方法が提供されます。

- 基本的な同期方法では、新規文書、変更文書、および削除文書に関する情報をステージング表に自動的に保管するトリガーを使用します。
- 拡張された同期方法では、トリガーを使用して変更文書に関する情報をステージング表に保管しますが、新規文書と削除文書に関する情報は整合性処理を介してキャプチャーされ、その情報は補助ステージング表に保管されます。

詳しくは、『テキスト検索索引の作成、更新、およびプロパティの変更』のトピックを参照してください。

DB2 Text Search は、さまざまなソースからデータを収集して、それらに索引付けすることによってその後の検索を高速化できます。DB2 Text Search では、言語学的な分析を使用して、より適切な検索結果が得られるようにします。サポートされる文書フォーマットは以下のとおりです。

- 非構造化プレーン・テキスト。
- 構造化テキスト (HTML または XML 文書のテキストなど)。
- 専有文書フォーマット (PDF または Microsoft Office 文書フォーマットなど)。

専有フォーマットの場合は、フィルタリング・ソフトウェアが必要です。この場合、追加のダウンロードとセットアップ・ステップが必要になる可能性があります。

DB2 Text Search では、パーティション・データベース環境でのフルテキスト検索がサポートされます。さらに、範囲パーティション表のため、あるいは単一パーティション環境やパーティション・データベース環境でマルチディメンション・クラスタリング・フィーチャーを使用する表のために、テキスト検索索引を作成することもできます。テキスト検索索引は、あらゆるパーティション・フィーチャーの組み合わせでサポートされます。パーティション・データベース環境では、複数のデータベース・パーティションにまたがる表のパーティショニングに応じて、テキスト検索索引がパーティション化されます。表のパーティショニングやマルチディメンション・クラスタリングなど、その他のパーティショニング・フィーチャーは、テキスト検索索引のパーティショニングには影響を及ぼしません。

また、DB2 Text Search では、統合セットアップとスタンドアロン・セットアップの両方がサポートされます。パーティション環境の場合は、データベース・サーバーとのリソース競合が回避されるため、スタンドアロン DB2 Text Search サーバーをお勧めします。DB2 Text Search は DB2 pureScale®環境ではサポートされていません。

---

## DB2 Text Search サーバーのデプロイメント・シナリオ

DB2 Text Search では、テキスト検索サーバーの統合インストールおよび DB2 データベース製品から分離したスタンドアロン・テキスト検索サーバーのインストールがサポートされます。スタンドアロン・テキスト検索サーバーは、エンタープライズ・コンテンツ管理 (ECM) テキスト検索サーバーとも呼ばれ、サポートされて

いるホスト・プラットフォーム上にインストールして管理することができます。  
DB2 Text Search は、高可用性災害時リカバリー (HADR) フィーチャーではサポートされません。

DB2 データベース・インスタンスでは TCP/IP を使用して、スタンドアロンの DB2 Text Search サーバーと通信を行います。SSL や GSKit サポートはありません。ただし、**stunnel** プログラムまたは SSH トンネリングを介して、暗号化チャネルを使用することができます。セキュリティー要件に応じて、文書リポジトリとテキスト検索索引ファイルへのアクセスを制限してください。スタンドアロン・テキスト検索サーバーは、テキスト検索索引への無許可アクセスを防ぐため、ファイアウォールの内側にある、セキュアなネットワーク接続を行うコンピューター上にインストールする必要があります。スタンドアロン・テキスト検索サーバーに TCP/IP アクセス制限を設定することで、データベース・サーバーがインストールされているホストによってのみアクセスできるようになります。

以下の図は、統合セットアップとスタンドアロン・セットアップを含む、DB2 Text Search サーバーのデプロイメントの概要を示したものです。統合 DB2 Text Search サーバーをセットアップして構成し、後でスタンドアロン・サーバーに切り替えることができます。ただし、テキスト検索索引を別のテキスト検索サーバーに移動する自動化されたサポートはありません。このため、セットアップによっては、既存のテキスト検索をドロップしてから、新しいテキスト検索サーバーをデータベース・インスタンスに割り当てなければならないことがあります。

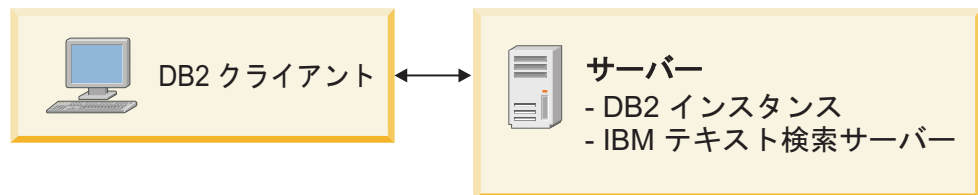


図3. 統合 DB2 Text Search サーバーのセットアップ

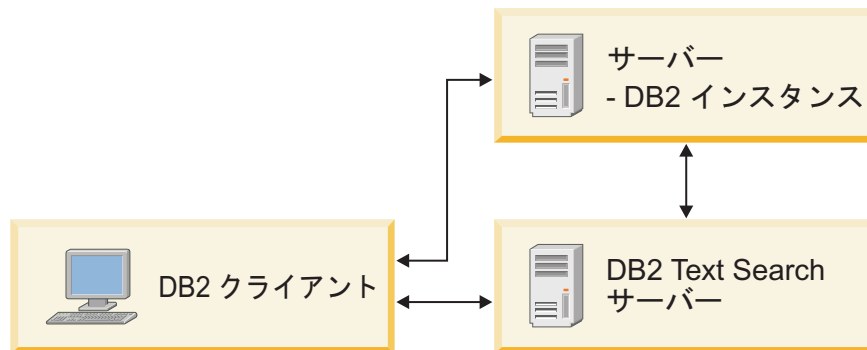


図4. スタンドアロン DB2 Text Search サーバーのセットアップ

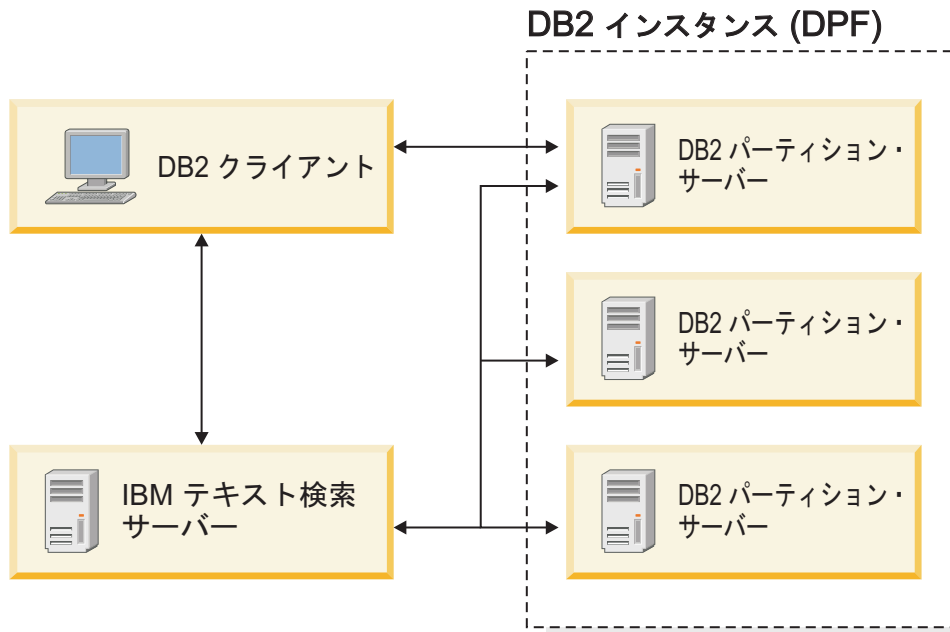


図5. パーティション環境でのスタンドアロン DB2 Text Search サーバー・セットアップ

注: DB2 Text Search のインストール・ディレクトリーは、デプロイメントのタイプによって異なります。

- 統合サーバーの場合:
  - `<TS_HOME>` は、Windows、Linux または UNIX の各オペレーティング・システム上の `../sqllib/db2tss` パスを表します。
- スタンドアロン・セットアップの場合、`<ECMTS_HOME>` はテキスト検索サーバーのインストール場所を表します。
  - デフォルトでは、`<ECMTS_HOME>` は、Linux システム上または UNIX システム上の `/opt/ibm/ECMTextSearch` パスを表します。
  - デフォルトでは、`<ECMTS_HOME>` は、Windows システム上の `C:\Program Files\IBM\ECMTextSearch` パスを表します。

スタンドアロン・テキスト検索サーバーのデプロイメントを、以下の点について考慮する必要があります。

- セキュリティー管理: スタンドアロンの Text Search サーバーにより、データベース・インスタンス所有者以外のテキスト・サーバー・プロセス所有者の定義が可能になります。
- ワークロード管理: スタンドアロンの Text Search サーバーにより、リソースを大量に消費するテキスト検索処理がデータベース・サーバー・タスクから分離されます。

各データベース・インスタンスは、単一テキスト検索サーバーに関連付けられています。複数のパーティション・サーバーを含むパーティション・データベース環境では、スタンドアロン・セットアップにより、リソースを大量に消費する処理が単一のパーティション・サーバーに集中するのを防ぐことができます。

スタンドアロンの Text Search サーバーと統合 Text Search サーバーの違いは初期構成のみです。その中で特に注目すべき点は、スタンドアロンの Text Search サーバーがリッチ・テキスト/専有フォーマット文書の処理用に既に構成されていることです。

---

## テキスト検索索引の作成、更新、およびプロパティの変更

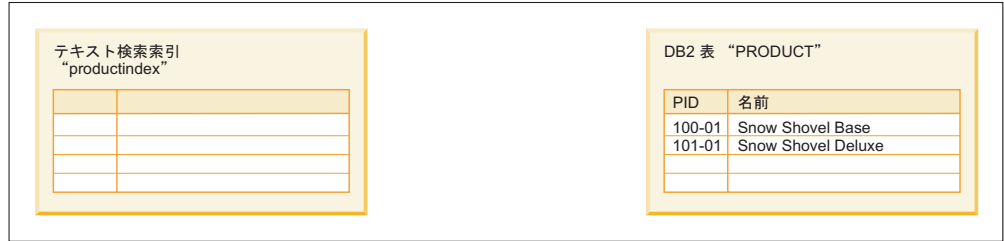
テキスト検索索引の作成は、テキスト索引のプロパティを定義する処理です。テキスト検索索引を作成した後で、その索引に関連付けられている表からデータを追加して、索引を更新する必要があります。 **UPDATE FREQUENCY** または **UPDATE MINIMUM** パラメーターなどの、テキスト検索索引のいくつかのプロパティを後で変更することもできます。

テキスト検索索引を使用すれば、テキスト検索関数を使用してテキスト列のデータを検索できます。テキスト検索索引は、テキスト文書から抽出された重要な用語から構成されます。表の行の主キーは、用語の抽出元を識別するために索引で使用されます。

テキスト検索索引が作成された直後には、データは含まれていません。 **db2ts UPDATE INDEX** コマンドまたは **SYSTS\_UPDATE** 管理 SQL ルーチンを使用して、テキスト検索索引にデータを追加してください。最初の索引更新 (初期更新 と呼ばれる) では、テキスト列のすべてのテキスト文書がテキスト検索索引に追加されます。それ以降の更新 (インクリメンタル更新 と呼ばれる) では、基本表のデータがテキスト検索索引に同期されます。

以下の例では、ユーザーは **SAMPLE** データベースにある **PRODUCT** 表に **MYSHEMA.PRODUCTINDEX** というテキスト検索索引を作成します。「テキスト検索索引の作成および初期更新とインクリメンタル更新の実行」は、ユーザーが初期更新を実行するまで索引が空であることと、ユーザーが表にデータを追加するときに、インクリメンタル更新を実行してテキスト検索索引に新しいデータを追加する必要があることを示しています。

### テキストの索引を作成



### テキストの索引を更新 (初期更新)



### テキストの索引を更新 (インクリメンタル更新)

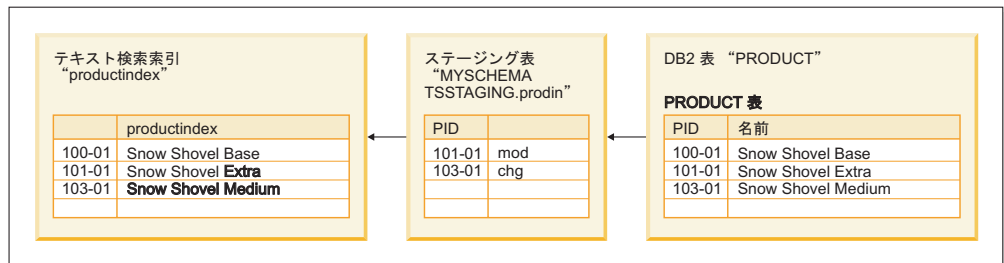


図6. テキスト検索索引の作成および初期更新とインクリメンタル更新の実行

DB2 Text Search では、テキスト索引とその表との同期化を行う以下の 2 つの方法が提供されます。

- 基本的な同期方法では、新規文書、変更文書、および削除文書に関する情報をステージング表に自動的に保管するトリガーを使用します。テキスト索引ごとに 1 つのステージング表が存在します。

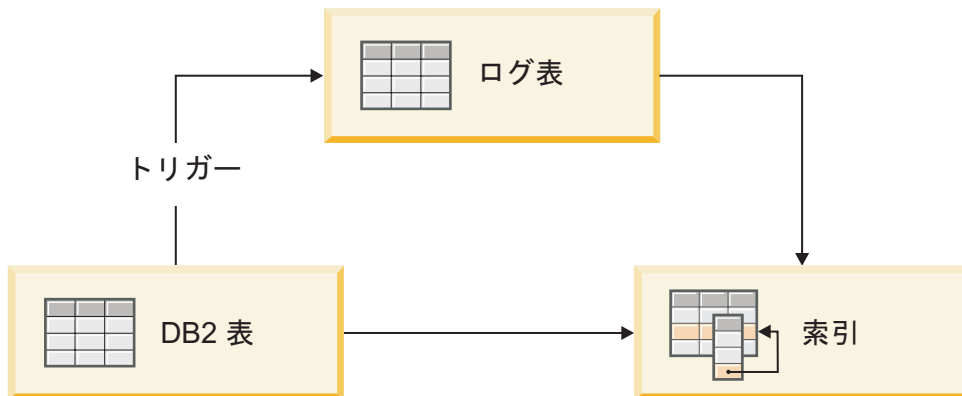


図7. トリガーによるインクリメンタル更新



基本的な方法ではトリガーのみを使用するため、トリガーで認識されない更新は無視されます。例えば、**LOAD** コマンドを指定したデータのロード、または範囲パーティション表の範囲のアタッチやデタッチがこれに該当します。

- 拡張された同期方法では、トリガーを使用して変更文書に関する情報をステージング表に保管しますが、新規文書と削除文書に関する情報は整合性処理を介してキャプチャーされ、その情報はテキスト保守補助ステージング表に保管されます。パーティションをアタッチする場合やデータをロードする場合は、基本表に対して **SET INTEGRITY** コマンドを発行して補助ステージング表のデータを利用可能にする必要があります。パーティションをデタッチする場合は、ステージング表に対しても **SET INTEGRITY** コマンドを発行してデータを処理できるようにする必要があります。または、基本表に対して **RESET PENDING** コマンドを使用して、すべての補助ステージング表のデータをアクセス可能にします。コマンドを実行している間は、読み取り操作と書き込み操作のために基本表にフルアクセスすることができます。パーティションをデタッチする場合は、基本表で **RESET PENDING** コマンドを実行するか、各ステージング表で **SET INTEGRITY** コマンドを実行する必要があります。

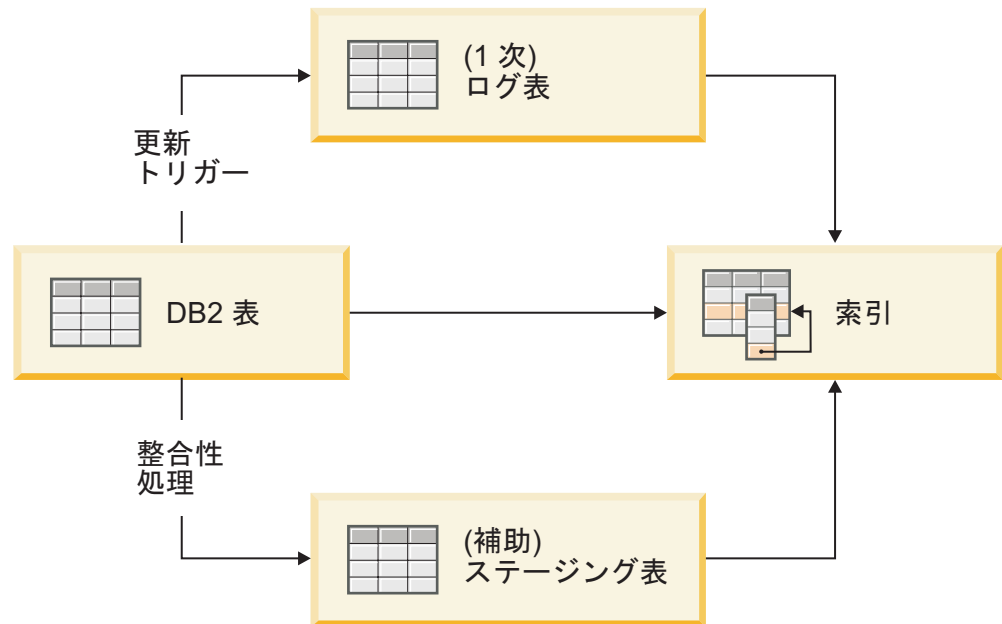


図8. トリガーと整合性処理によるインクリメンタル更新

一部のデータベース操作では、テキスト検索索引が暗黙的または明示的に無効になります。明示的な無効化では、**SYSIBMTS.TSINDEXES** 管理ビューでテキスト検索索引の状況が **INDSTATUS='INVALID'** に設定されます (コマンド **ALTER DATABASE PARTITION GROUP** の場合など)。暗黙的な無効化は、内容変更でステージング・メカニズムがバイパスされた場合 (**LOAD INSERT** が拡張ステージング・インフラストラクチャーなしで使用された場合など) に発生します。暗黙的な無効化では、テキスト検索索引に無効のマークは付けられません。

テキスト索引は、手動オプションまたは自動オプションを使用して更新できます。自動オプションでは、指定された日付および時刻による更新スケジュールを使用します。テキスト検索索引は、**UPDATE INDEX FOR TEXT** コマンドまたは

**SYSPROC.SYSTS\_UPDATE** プロシージャーを実行することで、手動で更新できます。テキスト検索索引は非同期で (つまり、データベースのデータを挿入、更新、または削除するトランザクションとは別に) 更新されます。非同期のテキスト検索索引更新処理により、スループットと並行性が改善されます。これは、複数の更新をまとめて処理し、影響を受けるテキスト索引セグメントのコピーに適用できるためです。その結果、テキスト検索索引は、更新された索引セグメントに元のセグメントが置き換えられる短い間だけ、読み取りアクセスに対してのみロックされます。

テキスト検索索引は、必要に応じて自動的に再編成されます。また、索引を更新するときに、**adminTool** を使用して明示的に索引の再編成をトリガーしたり、**ALLROWS** オプションを使用して索引を再作成したりすることができます。

---

## パーティション・データベース環境での DB2 Text Search

DB2 Text Search では、パーティション・データベース環境でのフルテキスト検索がサポートされます。テキスト検索索引は、それらが作成される基本表と一致するパターンで分散されます。データベース・パーティションごとに、テキスト索引パーティション (コレクションとも呼ばれる) が作成されます。このパターンでは、すべての索引パーティションでテキスト検索索引更新を並行して実行できるため、テキスト検索保守が容易になります。

複数のコレクション・テキスト検索索引更新では、コレクションごとではなく、索引ごとにステージング表が使用され、基本表と似た方法で分散されます。ステージング表では、**DBPARTITIONNUM** スカラー関数を使用して、索引リフレッシュごとに各索引パーティションに適用する必要がある関連する変更を見つけます。各データベース・パーティション・サーバーからのデータは、更新操作の並列化を有効にするために、テキスト検索索引の更新時に対応するテキスト索引パーティションで更新されます。

テキスト検索索引の更新ごとに複数のコレクション更新が行われる可能性があり、テキスト検索サーバーのキャパシティー・プランニングが必要になります。ワークロードを分散する場合、パーティション・データベース環境ではスタンドアロンのリモート・テキスト検索サーバー・セットアップをお勧めします。

DB2 インスタンスとは別にインストールおよび構成される DB2 Text Search サーバー・セットアップは、スタンドアロン・セットアップと呼ばれます。リモート・スタンドアロン・セットアップ (つまり、データベース・サーバーとは別のホストへのセットアップ) を非パーティション、単一パーティションおよび複数パーティションの各 DB2 インスタンスに使用して、リソースを大量に消費するテキスト検索サーバーのワークロードをデータベース・サーバー・ホストから除去することができます。

パーティション・データベース・インスタンスのデフォルトのインスタンス作成時の統合 Text Search サーバーの構成は、最小番号のデータベース・パーティション・サーバーに適用されます。既存のパーティション・データベース環境内の Text Search サーバーの管理および構成を Text Search サーバー・ツールで管理できるように、インストール時に構成する必要はありません。

以下のダイアグラムは、4 つのデータベース・パーティションがある DB2 インスタンスを示しています。これらのパーティションは **Machine1** および **Machine2** とい

う 2 つの専用ホストに配置されており、ホストごとに 2 つのロジカル・パーティションがあります。すべてのデータベース・パーティション・サーバーに対して、単一のテキスト検索サーバーが対応します。

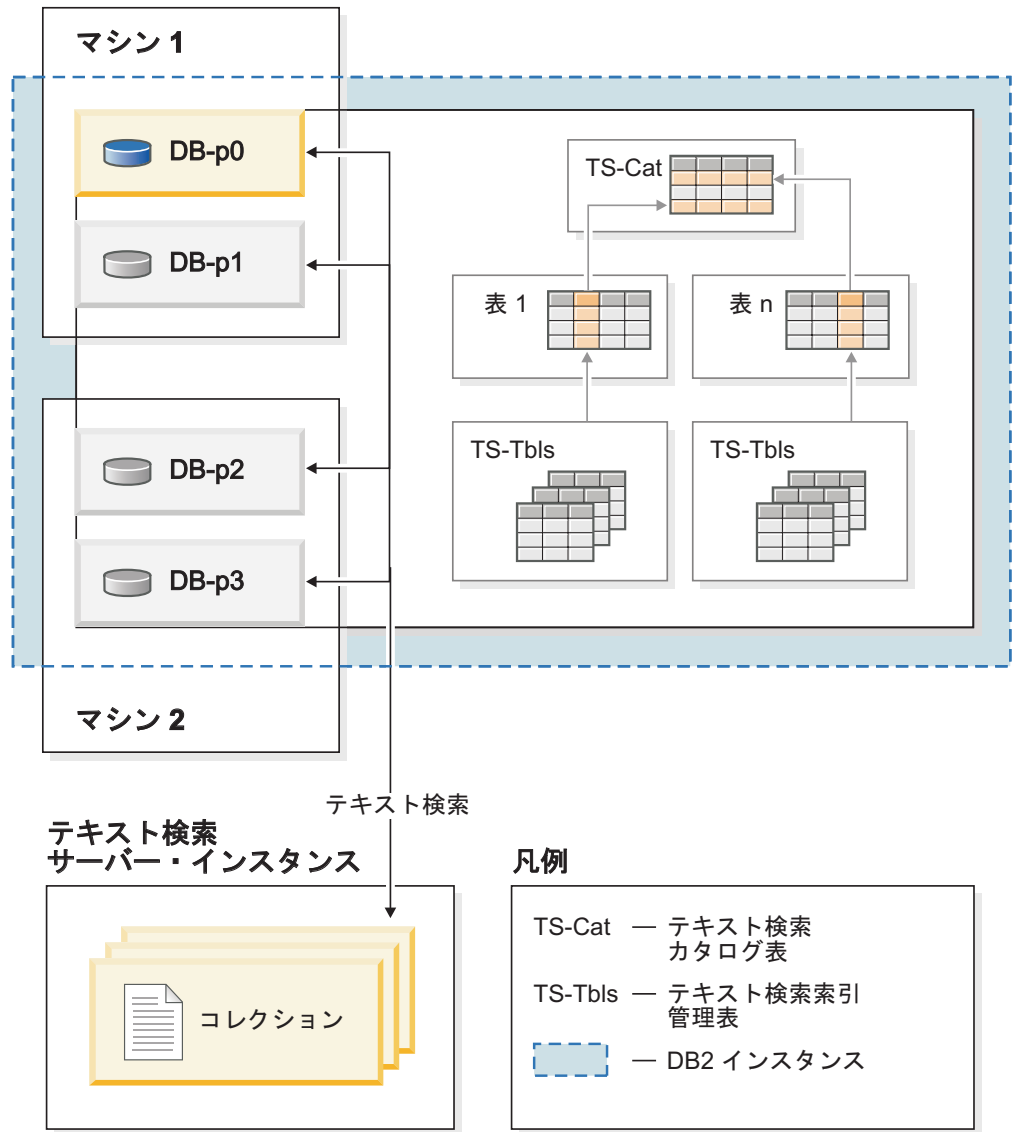


図9. パーティション環境での DB2 Text Search サーバー・セットアップ

スタンドアロン・セットアップは、ワークロードをバランス良く分散し、テキスト検索サーバーと単一データベース・パーティション・サーバーのリソース共有の回避に役立つため、お勧めするセットアップです。

パーティション・データベース環境では、**STATUS** パラメーターと **VERIFY** パラメーターを指定した **db2ts START FOR TEXT** コマンドを、任意のパーティション・サーバー・ホストで発行できます。インスタンス・サービスを開始するには、**db2ts START FOR TEXT** コマンドを統合テキスト検索サーバーのホスト・マシンで実行する必要があります。統合テキスト検索サーバーのホスト・マシンは、最も小さい番号のデータベース・パーティション・サーバーのホストです。カスタム・コレクション・ディレクトリーが使用される場合、これより番号の小さいパーティションが後

で作成されないようにしてください。この制限は、特に Linux と UNIX プラットフォームに関連します。インスタンスを作成するときに DB2 Text Search を構成する場合、構成によって最初に統合テキスト検索サーバーのホストが決定されます。この構成は、常に最も小さい番号のデータベース・パーティション・サーバーのホストのままである必要があります。

パーティション・インスタンス内でデータベース・パーティションを追加および削除することができます。この操作の後に通常、データの再配分を行います。それには **REDISTRIBUTE DATABASE PARTITION GROUP** コマンドを使用し、表のデータを移動してリバランスを行います。テキスト検索索引が影響を受けた表の 1 つによってホストされる場合、そのようなデータの再配分では、テキスト索引パーティションの内容を再度入れ替えて、テキスト索引パーティションを関連データベース・パーティションの新しいセットに合わせて調整する必要があります。テキスト検索索引のインクリメンタル更新は、通常、この目的には適しておらず、代わりに、FOR DATA REDISTRIBUTION オプションを使用してテキスト検索索引を更新する必要があります。これにより、初期更新の場合と同じように、ワークロードが大きい場合にはダウン時間がかなり長くなる可能性があることに注意してください。

パーティション・データベース環境において DB2 Text Search を使用可能にし、管理するときは、以下の点を考慮してください。

- DB2 の資料で説明されているとおりに、DB2 のセットアップが完了していることを確認してください。NFS マウントが root アクセス権限および setuid で構成されている必要があります。
- 始動が失敗したら、DB2 Text Search が正しく構成されているか確認して、**db2ts START** コマンドをもう一度発行する必要があります。
- `db2nodes.cfg` ファイルにパーティション番号を挿入または削除する前に、DB2 Text Search インスタンス・サービスを停止してください。これは、`db2nodes.cfg` 構成ファイルが結果的に変更される可能性があるすべてのコマンドに当てはまります。
- Windows プラットフォームで、パーティション・データベース環境で DB2 Text Search を使用している場合は、`db2nodes.cfg` ファイルで、同じホストに対して IP アドレスとホスト名の両方を使用すべきではありません。

パーティション・データベース環境で検索を行うときには、以下の考慮事項に注意してください。

- 検索中に **RESULT LIMIT** がパーティションごとに評価されます。これは、**RESULT LIMIT** に 3 を指定して、4 個のパーティションを使用すると、最大 12 個までの結果が得られることを意味します。
- **SCORE** 値は、照会で複数のパーティションにアクセスする場合でも、単一パーティションからのすべての文書の **SCORE** 値と比較した場合の、文書の関連性を反映します。

## DB2 Text Search 索引のインクリメンタル更新

DB2 Text Search のデータ同期は、新規文書、変更文書、または削除文書に関する情報をキャプチャーするたびにステージング表を更新するトリガーに基づいています。テキスト索引ごとに 1 つのステージング表が存在します。ステージング表内の情報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** を実行すると言います。

DB2 Text Search **CREATE INDEX** 命令の **INDEX CONFIGURATION** パラメーターの **AUXLOG** オプションにより、テキスト検索索引でテキスト保持ステージング表を使用するかどうかが制御されます。 **AUXLOG** オプションを **ON** に設定した場合、新規文書と削除文書に関する情報は、DB2 Text Search によって保持されている補助ステージング表の整合性処理を介してキャプチャーされ、変更文書に関する情報はトリガーを介してキャプチャーされてステージング表に保管されます。デフォルトでは、この構成オプションは、範囲パーティション表の場合は **ON** に、非パーティション表の場合は **OFF** に設定されます。

テキスト索引のインクリメンタル更新の変更を整合性処理を介してキャプチャーするには、追加の管理タスクの実行が必要になる場合があります。例えば、テキスト検索索引更新を処理する前に、**RESET PENDING** コマンドの発行が必要になる場合があります。テキスト保持ステージング・インフラストラクチャーの効果は、据え置きリフレッシュによるマテリアライズ照会表 (MQT) の効果に似ています。

表を、その表内のすべての行に影響するコマンドのみ (**LOAD REPLACE** コマンドなど) を使用して更新する場合、拡張ステージング・インフラストラクチャーを追加しても効果はありません。代わりに、表の更新後に索引を再作成することをお勧めします。

表にテキスト索引を作成し、初期更新およびインクリメンタル更新を介してデータをロードする場合は、以下の例を参照してください。

1. 表を作成して、データをそれに追加します。

```
db2 "create table test.simple (pk integer not null primary key,
comment varchar(48))"
db2 "insert into test.simple values (1, 'blue and red')"
```

2. テキスト検索索引を作成します。

```
db2ts "create index test.simpleix for text on test.simple(comment)
index configuration(auxlog on) connect to mydb"
```

3. 索引を更新して、データをロードします。

```
db2ts "update index test.simpleix for text connect to mydb"
db2 "load from loaddata4.sql of del insert into test.simple"
```

4. ロード操作の後、基本表はロックされます。例えば、選択操作の結果は、「SQL0668N 理由コード 1 のため、表 TEST.SIMPLE に対する操作は許可されません。SQLSTATE=57016」になります。ステージング表はアクセス可能ですが、新規データに関する情報はまだ含まれていません。

5. 整合性処理を有効にします。

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

以下のメッセージが返されます。

SQL3601W ステートメントにより 1 つ以上の表が自動的に SET INTEGRITY ペンディング状態になりました。SQLSTATE=01586

6. この時点で、ステージング表はロックされ、基本表に対する変更操作はリジェクトされます。例えば、以下のステートメントが失敗した場合、

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

以下のメッセージが返されます。

```
DB21034E The command was processed as an SQL statement because
it was not a valid command line processor command. During SQL processing
it returned:
SQL0668N Operation not allowed for reason code "1" on
table "SYSIBMTS"."SYSTSAUXLOG_IX114555". SQLSTATE=57016
```

7. 表をリセットします。

```
db2ts "reset pending for table test.simple for text connect to mydb"
```

**RESET PENDING** コマンドが正常に発行されると、ステージング表はロックを解除され、基本表に対する変更が再び可能になります。ステージング表のロックを解除するには、基本表に対して **RESET PENDING** コマンドを発行して、依存するすべてのテキスト保持ステージング表のロックを解除するか、または特定のステージング表に対して **SET INTEGRITY** コマンドを使用します。

8. これで、テキスト保持ステージング表に、テキスト検索索引に適用する必要がある変更が格納されました。索引に更新コマンドを発行します。

```
db2ts "update index test.simpleix for text connect to mydb"
```

---

## DB2 Text Search の言語学的処理

DB2 Text Search は、文書および照会の言語学的処理をサポートするディクショナリー・パックを提供します。また、中国語、日本語、韓国語などの言語の場合に N-gram セグメンテーションがサポートされます。中国語、日本語、韓国語などの言語の場合、検索エンジンでは、ディクショナリー・ベース・ワード・セグメンテーションの代わりに N-gram セグメンテーションを選択するオプションが提供されます。

テキスト文書がサポートされる言語の 1 つで記述されている場合、テキストが個別のワードに分割されるトークン化ステージの際に、言語学的処理が実行されます。サポートされない言語については、空白文字または N-gram セグメンテーションを使用して文書が構文解析されます。レンマ化 (ステミングのように、ワードの正規形を検索することを意味しますが、ワードの品詞の分析も行います) は、サポートされない言語については実行されません。

テキスト検索索引を検索するとき、索引付き文書に照会用語または照会用語の言語学的なバリエーションが含まれる場合に、一致が示されます。ワードのバリエーションは、照会の言語によって異なります。

### 中国語、日本語、および韓国語の文書の言語学的処理

検索エンジンでは、優れた検索結果を得られるかどうかはテキスト処理に使用された技法に大きく依存します。テキストが文書から抽出された後に、テキスト処理の最初のステップはテキスト内の個々のワードを識別することです。テキスト内の個々のワードを識別することは、セグメンテーションと呼ばれます。多くの言語で、

空白文字 (ブランク、行末、および特定の句読点) がワード境界を認識するために使用されます。ただし、中国語、日本語、および韓国語では、ワードの分離に空白文字が使用されないため、他の技法を使用する必要があります。

DB2 Text Search では、中国語、日本語、および韓国語に対して、ディクショナリー・ベース・ワード・セグメンテーションとも呼ばれる形態素セグメンテーション・オプション、および N-gram セグメンテーション・オプション (デフォルト設定) の 2 つの処理オプションが提供されます。

形態素セグメンテーションでは、言語固有のディクショナリーを使用して、文書内の文字列の中からそれぞれの単語を識別します。この技法では、ディクショナリーを使用してワード境界を識別するため、正確な検索結果が得られます。

N-gram セグメンテーションは、ワード境界を識別する際の問題を回避し、代わりに、重なり合う文字のペアに索引を付けます。2 つの文字が使用されるので、この技法は bi-gram セグメンテーションとも呼ばれます。N-gram セグメンテーションでは、常に、検索語を含む一致するすべての文書を戻します。ただし、この技法では、照会に一致しない文書が戻されることがあります。

## 例

両方のタイプの言語学的処理がどのように機能するかを例を示すために、文書内の「神奈川県知事選挙」というテキストについて考えます。日本語では、このテキストは 8 文字で成り立っています。この例では、8 つの文字は「神奈川県知事選挙」です。ユーザーが入力する可能性がある照会のサンプルとしては、4 文字の「知事選挙」が考えられます。(文書内のテキストとサンプルの照会の両方に、類似の文字列が共通して含まれています。)

- 形態素セグメンテーションを使用して文書が索引付けられると、検索エンジンは、「神奈川県知事選挙」というテキストを「ABC DEF GH」という文字セットにセグメント化します。

「知事選挙」というサンプル照会は、「EF GH」という文字セットにセグメント化されます。「EF」という文字は、文書テキストのトークンにはありません。文書に「EF」が含まれていなくても、「DEF」は含まれています。

文書テキストに「DEF」は含まれますが、照会には「EF」しか含まれていないため、サンプル照会を使用して文書が見つかることは期待できません。

形態素セグメンテーションを有効にすると、多くの場合、より正確な検索結果が得られますが、結果の数は減る可能性があります。

- N-gram セグメンテーションを使用して文書が索引付けられると、検索エンジンは、「神奈川県知事選挙」というテキストを「AB BC CD DE EF FG GH」という文字セットにセグメント化します。

サンプルの照会の「知事選挙」は、「知事 事選 選挙」という文字セットにセグメント化されます。サンプルの照会の「知事選挙」で検索すると、文書テキストと照会の両方のトークンが同じ順序で出現するため、この文書はこの照会で検出されます。

N-gram セグメンテーションを有効にすると、多くの場合、より多くの検索結果が得られますが、結果の精度は低下する可能性があります。例えば、日本語では、「京都」を検索する場合で、索引付けされた文書に「東京都」というテキストが含まれる場合、「京都」の照会は「東京都」というテキストを含む文書を戻します。これは、日本語で「東京都」と「京都」は 2 つの文字が同じであるためです。

---

## シナリオ: 索引付けおよび検索

DB2 Text Search をインストールして構成した後、検索を実行する前に行う必要のある 4 つのステップがあります。

1. DB2 Text Search のインスタンス・サービスを開始します。
2. DB2 Text Search が使用できるように、データベースを準備します。

データベースを有効にし、構成プロシージャを使用して、Text Search サーバーの関連付けを完了します。データベースは、DB2 Text Search に対して一度だけ有効にする必要があります。構成プロシージャは、以下の場合に必要です。

- 有効化が不完全だった。
- パーティション・データベースの場合。
- スタンドアロン Text Search サーバーのセットアップの場合。

DB2 Text Search で一度有効にしたデータベースに対しては、Net Search Extender を有効にできないことに注意してください。

3. 検索するテキストを含んでいるまたは含む予定の列に、テキスト検索索引を作成します。
4. テキスト検索索引にデータを追加します。これにより、空の新規に作成されたテキスト検索索引にデータが追加されます。

指定された更新頻度に応じて、テキスト検索索引の自動更新をセットアップする場合は、DB2 Text Search 索引更新のスケジューリングに関するトピックを参照してください。

テキスト検索索引にデータが入れると、SQL ステートメントを使用して索引を検索することができるようになり、索引に XML データが含まれる場合には XQuery を使用して検索することができます。

17 ページの図 10 が示しているように、手動または自動のいずれかの方法で、索引が関連付けられたテキスト列に対する変更内容を反映するように既存のテキスト検索索引を更新する必要があります。



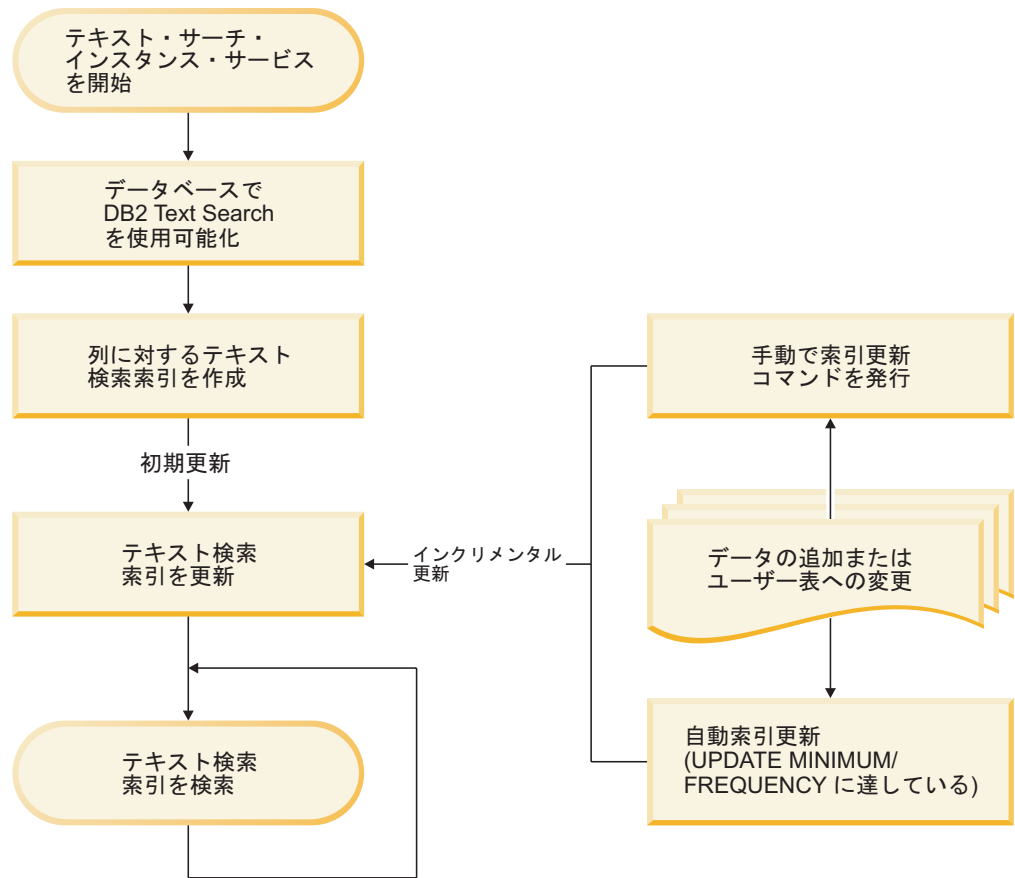


図 10. 統合 Text Search サーバーを使用した非パーティション・インスタンスでの検索のためのテキスト検索索引の設定

## 基本のシナリオ

SAMPLE データベース内の PRODUCT 表にある製品を、DB2 Text Search で検索可能にする方法を考えます。サンプル・データベースが (db2samp1 コマンドを実行して) 作成済みであり、DB2DBDFT 環境変数を SAMPLE に設定していると想定すると、以下のコマンドを発行できます。

```

db2ts START FOR TEXT
db2ts ENABLE DATABASE FOR TEXT
db2ts CREATE INDEX myschema.productindex FOR TEXT ON product(name)
db2ts UPDATE INDEX myschema.productindex FOR TEXT

```

PRODUCT の NAME 列に含まれる製品名および説明は、索引付けられて検索可能になりました。すべての雪かきショベル (snow shovel) の製品 ID を見つけるためには、以下の検索照会を発行します。

```

db2 "SELECT pid FROM product WHERE CONTAINS (name, 'snow shovel') = 1"

```

## DB2 Text Search と Net Search Extender の共存シナリオ

Net Search Extender に対して既にデータベースが有効であり、そのデータベースで Text Search を使用する場合は、索引共存フィーチャーを使用してデータベースを照会することができます。

テキスト検索のためにデータベースを開始します。

```
db2ts start for text
DB20000I The SQL command completed successfully.
```

Net Search Extender 索引が既に存在するデータベースに対して Text Search を有効にします。

```
db2ts enable database for text
CIE00001 Operation completed successfully
```

既存の Net Search Extender 索引を持つ列に、DB2 Text Search 索引を作成および更新します。

```
db2ts "CREATE INDEX db2ts.title_idx FOR TEXT ON books(title)"
CIE00001 Operation completed successfully.
```

```
db2ts "UPDATE INDEX db2ts.title_idx FOR TEXT"
CIE00001 Operation completed successfully.
```

新規の DB2 Text Search 索引をアクティブにして、NSE 索引から新しい索引へ照会処理を切り替えます。

```
db2ts "ALTER INDEX db2ts.title_idx FOR TEXT SET ACTIVE"
CIE00001 Operation completed successfully.
```

DB2 Text Search 索引を使用するために照会を発行します。

```
db2 "select isbn, title from books where contains(title,'top')=1"
```

ISBN	TITLE
123-014014014	Climber's Mountain Tops
111-223334444	Top of the Mountain: Mountain Lore

2 record(s) selected.

両方のタイプのテキスト索引の使用を試行する照会はサポートされていません。例えば、title 列にアクティブな DB2 Text Search 索引があり、bookinfo 列にアクティブな Net Search Extender 索引があるとします。1 つの照会のテキスト索引はすべて同じ索引タイプでなければならないので、検索はエラーを返します。

```
db2 "select isbn, title from books where contains(title, 'top')=1 and
contains(bookinfo, ' MOUNTAIN ')=1"
```

ISBN	TITLE
------	-------

```
SQL20425N Column "BOOKINFO" in table "BOOKS" was specified as an argument to
a text search function, but a text search index does not exist for the column.
SQLSTATE=38H12
```

このエラーを回避するには、bookinfo 列に DB2 Text Search 索引を作成してアクティブにします。

```
db2ts "CREATE INDEX db2ts.bookinfo_idx FOR TEXT ON books( bookinfo )"
CIE00001 Operation completed successfully.
```

```
db2ts ALTER INDEX db2ts.bookinfo_idx FOR TEXT set active
CIE00001 Operation completed successfully.
```

---

## リッチ・テキストおよび専有フォーマットのサポート

DB2 Text Search では、適切に構成された DB2 Text Search インスタンス内のリッチ・テキスト・フォーマットと専有フォーマットの文書の索引付けおよび検索がサポートされています。

DB2 Text Search は、テキスト・データに対するフルテキスト検索用の索引を準備するために、TEXT、XML、および HTML テキスト索引形式をサポートします。また、INSO フォーマットを使用することで、以下のリッチ・テキストまたは専有フォーマットの文書での索引付けおよび検索が可能になります。

- リッチ・テキスト文書は、テキストとともに、太字、イタリック、フォント・タイプ、フォント・サイズ、スペーシングなどの書式設定の指示を含む文書です。
- 専有フォーマットには、pdf、doc、ppt、ods など、さまざまな一般的なオフィス製品が含まれます。

INSO フォーマット・フィーチャーの有効化と構成については、リッチ・テキスト・フォーマット用と専有フォーマット用の DB2 Text Search の設定に関するトピックを参照してください。



---

## 第 3 章 テキスト検索ソリューションの計画

サポートされる文書タイプや言語やユーザー・ロールなどの、特定のキー概念について理解することは、DB2 Text Search の利点を活用するために役立ちます。

---

### 文書の特徴

#### DB2 Text Search でサポートされる文書フォーマット

DB2 Text Search を使用して検索するテキスト文書の形式 (またはタイプ) を指定する必要があります。この情報は、テキスト文書に索引付けするために必要です。

テキスト列データは、プレーン・テキスト、HTML 文書、XML 文書、またはリッチ・テキストや専有フォーマットの文書にすることができます。文書は構文解析されて、索引付けのために関連する部分が抽出されるので、検索可能になります。例えば、HTML 文書のタグやメタデータなど、要素の中には索引付けされないため検索できないものもあります。

#### サポートされるデータ・タイプ

索引付けして検索するテキスト列内のデータ・タイプは、バイナリーまたは文字のどちらかにすることができます。

DB2 Text Search は、以下のデータ・タイプをサポートします。

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

#### サポートされない形式とデータ・タイプの変換

ユーザーは独自の関数を使用して、サポートされない形式またはデータ・タイプをサポートされる形式またはデータ・タイプに変換できます。

ユーザー定義関数 (UDF) を使用してテキスト索引を作成することにより、サポートされない文字をフィルター処理して索引付けし、サポートされない形式を処理可能なサポートされる形式に変換できます。

外部のサポートされないデータ・ストアに保管されている文書を索引付けするときも、このアプローチを使用できます。この場合、DB2 列に文書参照が含まれるとき、UDF を使用して関連した文書参照を持つ文書の内容を戻すことができます。

## サポートされる言語およびコード・ページ

最初にテキスト検索索引を作成するときに、特定の言語を使用してテキスト文書が構文解析されるように指定できます。また、検索中に照会用語が特定の言語で解釈されるように指定することもできます。さらに、バイナリー・データ・タイプの列にテキスト検索索引を作成するときに、コード・ページを指定できます。

### 言語の指定

「ロケール」は、言語およびテリトリー（地域または国）情報の組み合わせであり、5 文字のロケール・コードによって表されます。テキスト検索管理プロシージャにメッセージ・ロケールを定義するには、プロシージャにロケール・コードを渡します。DB2 サーバー上にインストールされたロケールに応じて、これらのロケール・コードを調整することが可能です。

テキスト検索索引の作成時に言語を指定することと、検索照会の発行時に言語を指定することとの間には、以下のような重要な相違があります。

- **db2ts CREATE INDEX** コマンドに指定するロケールによって、索引付けのための文書のトークン化または分析に使用される言語が決まります。索引付けされる列のすべての文書が特定の言語を使用することが分かっている場合、テキスト検索索引を作成するときに適切なロケールを指定してください。ロケールを指定しない場合、データベース・テリトリーを使用して **LANGUAGE** のデフォルト設定が決まります。ロケールを判別するために文書を自動的にスキャンするためには、**SYSIBMTS.TSDEFAULTS** ビューで、**LANGUAGE** 属性を **AUTO** に設定します。**SYSIBMTS.TSDEFAULTS** ビューでは、テキスト検索のためのデータベース・デフォルトが、属性と値の対を使用して記述されます。
- 検索照会で指定するロケールは、照会で言語処理を実行するため、および照会用語の基本型の識別を支援するために使用されます。基本型のロケールが識別された後、そのロケールは検索プロセス自体では使用されません。そのため、英語の基本型の検索語がドイツ語の文書内にあれば、照会に英語を使用して検索結果でドイツ語文書を取得することも可能です。

サポートされるロケールのリストは、 [ここに](#)あります。

### コード・ページの指定

サポートされる DB2 コード・ページの 1 つを使用する文書の場合、その文書に索引付けすることができます。テキスト検索索引の作成時にコード・ページを指定することはオプションですが、それはバイナリー列の文字エンコードを識別するために役立ちます。バイナリー列のコード・ページを指定しない場合、列プロパティからのコード・ページが使用されます。にあります。

## 文書のサイズに関する考慮事項

DB2 Text Search では、索引付けできる文書のサイズ、およびその文書に含まれる文字数についての制限があります。

正常に処理できる文書の最大サイズは、100 MB です。文書がこのサイズ制限を超えている場合、その文書は拒否され、イベント表にその情報（識別するための主キーが含まれる）で項目が作成されます。処理は、その更新操作に含まれる他の文書に対して続行されます。

DB2 Text Search では、テキスト文書ごとに索引付けできる Unicode 文字の数に制限があります。この文字制限によって、テキスト検索索引に含まれる長いテキスト文書が切り捨てられることがあります。

テキスト文書ごとに許可される Unicode 文字数のデフォルト値は、テキスト文書フォーマットによって異なります。

- *max.text.size* の値 (文字数) より大きなテキスト・ファイルは、索引付けされる前にこのサイズに切り捨てられます。デフォルト値は 60,000,000 文字です。
- *max.xml.text.size* の値 (バイト) より大きな XML ファイルは索引付けされません。デフォルト値は 60,000,000 バイトです。この数には、タグ名、属性名、および属性値が含まれますが、XML ディレクティブとコメントは含まれません。
- *max.binary.text.size* の値 (バイト) より大きなバイナリー・ファイルは索引付けされません。デフォルト値は 60,000,000 バイトです。この制限は、文書がテキストに変換された後に適用されます。

テキスト・ファイルのサイズが最大テキスト・ファイル・サイズ (デフォルトでは 60,000,000 文字) を超えた場合、テキスト・ファイルは索引付けされる前にサイズ制限まで切り捨てられます。テキスト文書が構文解析ステージの際に切り捨てられた場合、一部のテキストが正しくまたは完全に処理されていないことを示す警告を受け取ります。

バイナリーまたは XML フォーマットの文書のサイズが最大テキスト・ファイル・サイズ (デフォルトでは 60,000,000 バイト) を超えた場合、文書は索引付けされず、エラーが生成されます。

テキストの処理が不適切または不完全な場合、検索結果は不完全です。可能な場合は、サイズ制限を調整するか、処理する文書を整理します。警告についての詳細が、テキスト検索索引用に作成されたイベント表に書き込まれます。

ファイル・サイズ制限を増やす場合は、ヒープ・サイズをそれに合わせて増やす必要があります。最大ヒープ・サイズは、構成ツールを使用して **startupHeapSize** パラメーターを指定することで調整できます。

---

## DB2 Text Search セキュリティーの概要

DB2 Text Search では、操作を実行するユーザーの許可 ID に基づいて、管理操作を実行します。前のリリースとは異なり、インスタンス所有者のデータベース特権に関する前提条件がなくなったため、fenced ユーザーがインスタンス所有者と同じ 1 次グループに属する必要はありません。

ユーザーの許可 ID で操作を実行することで、監査能力が向上し、テキスト検索管理の制御が改善されます。アクセス制御を単純化するために、以下の 3 つの新しいシステム・ロールを使用できます。

- Text Search 管理者 (SYSTS\_ADM) - データベース・レベルで操作を実行します。
- Text Search マネージャー (SYSTS\_MGR) - 索引レベルで操作を実行します。
- Text Search ユーザー (SYSTS\_USR) - テキスト検索カタログ・データへのアクセス権限を持ちます。

セキュリティー管理者は、ユーザー定義ロールの場合と同様に、これらのロールを付与したり取り消したりすることができますが、それ以外の点では接頭部が SYSTS のロールはシステムによって管理されるため、ドロップすることも作成することもできません。

データベースを作成すると、これらのロールは自動的にそのデータベースの作成者に割り当てられ、制限のないデータベースでは、SYSTS\_USR ロールが PUBLIC に割り当てられます。その他のすべてのロール割り当てはセキュリティー管理者が明示的に行う必要があります。例えば、テキスト検索を使用可能または使用不可にする場合は、SYSTS\_ADM を割り当てます。

制限付きデータベース・セットアップでは、セキュリティー管理者は、スケジューラー・プロシーチャーの実行特権を SYSTS\_MGR ロールに付与し、SYSTS\_USR ロールに対してユーザー特権を付与する必要があります。

**SYSIBMTS** カタログ表の内容の管理またはアクセスを行うための表特権は、DB2 Text Search のデータベースを有効にする際に自動的にこれらのロールに付与されます。同様に、特定のテキスト検索索引のための **SYSIBMTS** 管理表の内容の管理またはアクセスを行う表特権は、テキスト索引の作成時に自動的にこれらのロールに付与されます。例えば、テキスト索引を作成する場合は、他のタイプの索引を作成するために必要な特権に相当する特権が基本表で必要になります。 **SYSIBMTS** 表へのアクセス特権を付与する SYSTS\_MGR ロールも必要です。

特定の索引レベル・コマンドでは、テキスト検索サーバーへの接続が必要になります。関連する接続情報は **SYSIBMTS.TSSERVERS** 管理ビューから取得され、認証トークンが含まれます。トークンは、テキスト検索サーバーの構成時に生成され、適切なテキスト検索サーバーが確実にアドレス指定されるようにするための識別手段として、呼び出し元によって使用されます。間違ったトークンが使用されている場合は、索引の管理や検索要求はリジェクトされます。

以下の表に、必要なロール特権の要約を示します。セキュリティー管理者は、操作を正常に行うために適切なロールをユーザーに付与しておく必要があります。

表 1. ロール特権

	ロール	操作
Text Search 管理者	SYSTS_ADM	有効化、無効化、コマンド・ロックの解除 (すべて)、構成
Text Search マネージャー	SYSTS_MGR	作成、更新、変更、ドロップ、イベントの消去、コマンド・ロックの解除 (索引ごと)、保留のリセット
Text Search ユーザー	SYSTS_USR	テキスト検索 <b>SYSIBMTS</b> カタログへの制限付きアクセス



## ユーザー・ロール

ユーザーのロール、および DB2 Text Search のユーザー権限には、さまざまなものがあります。システム・ロールでは管理操作の実行特権を制御するため、ユーザーの許可 ID には、テキスト検索操作を実行するためのデータベースまたは表へのアクセス特権に加えて、適切なテキスト検索ロールが必要になります。

標準的なユーザーは以下のとおりです。

- Text Search サーバー管理者
- Text Search 管理者
- Text Search 索引マネージャー
- テキスト検索照会を実行するユーザー

### DB2 Text Search サーバー管理者

Text Search サーバー管理者は、DB2 Text Search サーバー・オプションの構成、統合およびスタンドアロン・テキスト・サーバー・デプロイメント用のテキスト検索インスタンス・サービスの開始と停止、およびテキスト検索サーバー操作のモニターを行います。

統合テキスト検索サーバーのセットアップの場合、このロールはデータベース・インスタンス所有者に関連付けられます。

UNIX オペレーティング・システムと Windows オペレーティング・システムでは、インスタンス所有者が以下のように異なります。

- UNIX オペレーティング・システムでは、インスタンス所有者ユーザーは **db2icrt** コマンドで指定されたインスタンスの名前およびユーザー ID です。
- Windows オペレーティング・システムでは、インスタンス所有者は DB2 インスタンス・サービスを実行しているユーザー ID です。

DB2 バージョン 9.7 とは異なり、インスタンス所有者がデータベース特権を保持する必要はありません。スタンドアロン・テキスト検索サーバーのセットアップでは、サーバー管理者はテキスト検索サーバーの実行可能ファイル、構成ファイル、および索引ファイルへの適切なアクセス権を持っている必要があります。

### Text Search 管理者

Text Search 管理者は、データベースを DB2 Text Search で使用可能にしたり、使用不可にしたりします。Text Search 管理者が実行するもう 1 つのメインタスクは、コマンド・ロックのクリアです。

Text Search 管理者には、DBADM 権限に加えて SYSTS\_ADM ロールが必要です。これにより、テキスト検索索引を含むすべてのデータベース・オブジェクトの操作が可能になります。

### Text Search 索引マネージャー

Text Search 索引マネージャーは、テキスト検索索引の定義および保守を行います。

標準的なタスクは以下のとおりです。

- テキスト検索索引の作成およびそれらの特性の定義
- テキスト検索索引の更新

- テキスト検索索引の更新特性の変更
- テキスト検索索引のドロップ
- 定期的なイベント表のクリア

Text Search 索引マネージャーには SYSTS\_MGR ロールがあり、通常は、テキスト検索索引が作成される表に対する CONTROL 特権を持ちます。

### テキスト検索照会を実行するユーザー

検索照会を実行するユーザーは、SQL 照会で DB2 Text Search の CONTAINS および SCORE 関数をユーザー表に対して使用できます。それらのユーザーは、テキスト検索索引で表を参照する XQuery 内で xmlcolumn-contains 関数を使用することもできます。

特定の DB2 Text Search 検索の許可はありません。テキスト検索索引が作成された表に対してユーザーに付与されたアクセス権限に応じて、照会は許可されるかまたは拒否されます。ユーザーが指定の表に対して SELECT ステートメントを発行できる場合、その表に対するテキスト検索も実行できます。

検索照会を実行するユーザーは、例えば以下の機能を照会に含めることができます。

- テキスト検索を特定の文書に制限する (SQL または XQuery を使用)
- 指定の検索指数に関して、ある文書が、マッチング対象の他の文書と比較してどの程度適合するかを示すスコアを戻す (SQL を使用)

## アクセス・ポリシーと通信セキュリティ

### テキスト検索サーバーのファイル・アクセスに関する考慮事項

テキスト・サーバー・プロセスのプロセス所有者には、構成データ、およびカスタム・コレクション・ディレクトリーにあるコレクションを含むすべてのコレクション・データに対する読み取り権限および書き込み権限が必要です。

プロセス所有者は、統合テキスト・サーバーの場合はインスタンス所有者であり、スタンドアロン・テキスト・サーバーの場合は開始コマンドでテキスト・サーバーを開始するユーザーです。

コレクションには、ファイルを直接開いたときに、部分的に読み取れる機密データが含まれる場合があります。無許可アクセスを防ぐには、構成ディレクトリーとコレクション・ディレクトリーへのアクセス権限を確認して更新し、テキスト・サーバーのプロセス所有者だけがファイルにアクセスできるようにします。

### ステージング表のアクセス・ポリシー

テキスト索引に適用する必要がある変更を識別するために、変更 (挿入、更新、削除) された行の主キーがステージング表に挿入されます。

主キーは、機密データを含む基本表のデータ列に基づいている場合があります。デフォルトでは、SYSTS\_ADM ロールと SYSTS\_MGR ロールを持つユーザー、および (いくつかの制限がある) SYSTS\_USR ロールを持つユーザーに、少なくともステージング表の内容に対する読み取り権限があります。基本表のアクセス・ポリシー

および監査ポリシーは、ステージング表には継承されません。特定のステージング表へのアクセスにさらに制限が必要な場合、セキュリティ管理者は、ロールが持つ、特定の表への読み取り権限を取り消し、特定のテキスト索引を管理するユーザーまたはカスタム・ロールにその権限を付与する必要があります。

## スタンドアロンのセットアップ

DB2 データベース・インスタンスでは TCP/IP を使用して、スタンドアロンの DB2 Text Search サーバーと通信を行います。SSL や GSKit サポートは使用できませんが、stunnel プログラムまたは SSH トンネリングを介して、暗号化チャネルを使用することができます。セキュリティ要件に応じて、文書リポジトリとテキスト検索索引ファイルへのアクセスを制限してください。スタンドアロン・テキスト検索サーバーは、テキスト検索索引への無許可アクセスを防ぐため、ファイアウォールの内側にある、セキュアなネットワーク接続を行うコンピューター上にインストールする必要があります。スタンドアロン・テキスト検索サーバーに TCP/IP アクセス制限を設定することで、データベース・サーバーがインストールされているホストによってのみアクセスできるようになります。

---

## DB2 Text Search のキャパシティー・プランニングおよび最適化

DB2 Text Search のパフォーマンスおよびリソース使用量には、いくつかの要因が影響を与えます。DB2 Text Search のシステム・キャパシティーを計画する際には、照会のワークロード、並列して行う索引更新の数、テキスト索引の予期されるサイズと増加率、および索引付けする文書の処理時間を考慮に入れます。

DB2 Text Search では、DB2 データベース内のほとんどのデータ・タイプに対してフルテキスト検索照会を実行できます。これには、XML 文書やリッチ・テキストまたは専有フォーマット・フィーチャーのサポートが含まれます。フルテキスト検索は、データベース・インスタンスと統合されたテキスト検索サーバー・インスタンス、またはデータベース・インスタンスに関連付けられたスタンドアロン・セットアップのテキスト検索サーバー・インスタンスによってサポートされます。データベースとテキスト検索サーバー・インスタンスとの間の通信は、TCP/IP を経由します。フルテキストの索引付けおよび検索のパフォーマンスは、テキスト検索サーバーの構成、使用可能なシステム・リソース、およびテキスト索引固有の設定に依存します。

### テキスト検索サーバーの実装および構成

1 つのテキスト検索サーバーをデータベース・インスタンスとして構成します。テキスト検索サーバーの実動環境用の最小メモリー要件として、4 GB のメモリーが推奨されています。これは、並列して行う索引更新の数に応じて増加します。

テキスト検索索引の更新は、ディスク入出力および CPU またはメモリー所要量の両面でリソースを大量に消費します。Text Search サーバーのリソース使用量を制御するために、複数の構成パラメーターが用意されています。ワークロードを分散する場合、例えば、パーティション・データベース環境では、スタンドアロン・セットアップをお勧めします。

## テキスト検索索引のサイズ

平均すると、テキスト検索索引は元のデータの約 50% から 150% のサイズになります。

テキスト検索索引のサイズに絶対的な限度はありませんが、スループットの要素と完了時間の依存関係を組み合わせたものが、テキスト検索索引の合計サイズの実質的な限度になります。例えば、相当量のデータがテキスト検索索引へ追加されたり、そこから削除されたりする場合、照会パフォーマンスを向上させるためにテキスト検索索引の構造がマージされます。このマージを完了するのにかかる時間は、索引のサイズに依存します。

## スループットに影響する要因

テキスト索引更新の絶対的なスループットは、データ・タイプと索引のフォーマットに依存します。ユーザーが実感する照会パフォーマンスに最も影響を与えるのは、テキスト検索索引のサイズではなく一致した結果の数です。例えば、100 GB のテキスト検索索引に対して単一の検索語を使用する述部が 1 つ含まれている照会のパフォーマンスは、結果数が同じであれば、800 GB のテキスト検索索引に対する検索と同程度になります。

1 文書あたり約 10 KB から 100 KB のテキストがある場合に、テキスト索引の更新に最適な処理が行われます。テキストが 1 MB を超えている場合、または 1 KB 未満の場合、スループットが低下します。

## DB2 Text Search サーバー構成

キュー・サイズ、ヒープ・サイズ、索引付けスレッド数などの要素を調整することによって、DB2 Text Search 構成をチューニングできます。システムのパフォーマンスが最良になるように、これらのパラメーターに対する調整のバランスを取ってください。

DB2 Text Search サーバー構成では、索引付けスレッド数が CPU の数を超えないようにし、並列して行う更新の数が索引付けスレッド数を超えないようにする必要があります。パーティション・データベースで並列して行う更新の数を決定するために、索引の数にテキスト索引のコレクション数が乗算されることに注意してください。

構成を変更する前には、**db2ts STOP FOR TEXT** コマンドを使用して DB2 Text Search インスタンス・サービスを停止します。

configUtility を始動します。

- 統合テキスト検索サーバーでは、<TS\_HOME>/bin ディレクトリーにあります。
- スタンドアロン・テキスト検索サーバーでは、<ECMTS\_HOME>/bin ディレクトリーにあります。

例えば、索引付けスレッドの数を変更するには、次のようにします。

```
configTool configureParams -configPath configPath -numberOfIndexingThreads 3
```

変更を反映させるには、DB2 Text Search プロセスを再開します。

## 最大ヒープ・サイズの構成

文書取り込みスレッドによって文書を受け取る際、その内容は文書キューに配置されます。文書キューに配置された文書は、アクティブな索引付けスレッドがその文書を索引付けするまでそこに配置されたままになります。標準的な操作では、文書を文書キューに配置する速度は、文書の解析や索引付けに必要とされる時間よりも速くなります。そのため、ある時点で文書キューがその容量に達すると、他のスロットが文書キューから解放されるまで、文書取り込みスレッドがブロックされま

す。文書キューが未処理の文書でいっぱいになると、ヒープ・メモリーを消費します。構文解析や索引付けなどの文書処理を行うために、メモリーはさらに消費されます。結合されたヒープ・メモリー使用量を文書処理の最大ヒープ・サイズより小さくする必要があります。デフォルトでは、ヒープ・サイズは 1500 MB になるように構成されています。

また、入出力キューのメモリー・サイズとヒープ・メモリーの比率についても考慮してください。キューのサイズは、キュー内の文書のメモリー使用量によって決まります。長い文書 (例えば、それぞれ 20 MB) を処理するためにキュー・メモリー・サイズを大きく場合は、ヒープ・サイズも大きくすることを検討してください。

*startupHeapSize* 変数では、統合またはスタンドアロン DB2 Text Search サーバーで許可される最大ヒープ・サイズを設定します。デフォルトの開始ヒープ・サイズは 1.5 GB です。この値は、1.5 GB と、オペレーティング・システムおよび JVM バージョンで許されるメモリーの最大容量との間の数値にしなければなりません。以下の例を考慮してください。

- 例えば、Windows システムと 32 ビット JVM の場合、プロセスのヒープ・サイズは最大 2 GB です。このため、*startupHeapSize* パラメーターは 2 GB 未満に設定しなければなりません。例えば、1.8 GB のようにします。
- AIX® システムと 64 ビット JVM の場合は、最大ヒープ・サイズを制限するのはシステムに構成されている仮想メモリーの量だけです。多数の大きい文書 (平均サイズが 20 MB) を連続して処理する必要がある場合は、*startupHeapSize* パラメーターを約 4 GB まで増やします。

最大ヒープ・サイズは、応答ファイルに **IA\_STARTUP\_HEAP\_SIZE** パラメーターを指定して、スタンドアロン DB2 Text Search サーバーをインストールまたはアップグレードする際に設定できます。64 ビット・オペレーティング・システムへのスタンドアロン・テキスト検索サーバーのインストール時またはアップグレード時に、最大ヒープ・サイズを 2 GB より大きい値に設定すると、新規コレクションに対するテキスト文書、XML 文書、およびバイナリー文書のファイル・サイズ制限が増えます。コレクションごとのファイル・サイズ制限は、

<ECMTS\_HOME>%config%collections%collection\_name%parser\_config.xml ファイルに指定されています。新規コレクションのデフォルトのファイル・サイズ制限は、<ECMTS\_HOME>%config%defaults%parser\_config.xml ファイルに指定されています。2 GB を超えるヒープ・メモリー 8.3 MB ごとに、ファイル・サイズ制限の値 (デフォルトでは 60 MB) が 1 MB ずつ (400 MB まで) 増加します。

**重要:** インストール後に構成ツールを使用して最大ヒープ・サイズを変更する場合は、`parser_config.xml` ファイル内のファイル・サイズ制限を手動で調整する必要があります。ファイル・サイズ制限は、インストール時およびアップグレード時に、応答ファイルに `IA_STARTUP_HEAP_SIZE` パラメーターを指定した場合にのみ、自動的に調整されます。

最大ヒープを変更するには、次のコマンドを発行します。

```
configTool configureParams -configPath <full-path-to-configuration-folder>
-startupHeapSize <value>
```

ここで、`<value>` はヒープ・サイズで、`<full-path-to-configuration-folder>` は、DB2 Text Search サーバーの `config.xml` ファイルの絶対パスです。

32 ビットのオペレーティング・システムでの標準的な構成は次のとおりです。

- 最大ヒープ・サイズ: 1.8 GB
- キュー・サイズ: それぞれ 90 MB
- ファイル・サイズ制限: 60 MB

64 ビットのオペレーティング・システムでの標準的な構成は次のとおりです。

- 最大ヒープ・サイズ: 3 GB
- キュー・サイズ: それぞれ 150 MB
- ファイル・サイズ制限: 200 MB

## DB2 Text Search 索引付けスレッド

文書を解析および索引付けするために、複数の索引付けスレッドが並行して処理を行うことができます。こうすると、通常はテキスト検索索引の更新のための合計経過時間が削減されます。

索引付けスレッドは、キューから文書を取り出し、索引付けプロセスを管理します。その際、索引前処理スレッドを使用することにより、索引付け用に文書の内容が準備され、その結果がテキスト索引コレクションに書き込まれます。

索引前処理スレッドは、テキストを抽出し、言語を特定し、文書のトークン化および分析を行います。

通常は、索引付けスレッドの数と前処理スレッドの数は、等しくなるように構成されます。ただし、例えば、大きい文書を処理するようなシナリオでは、前処理スレッドの数を増やすと、パフォーマンス上の利点が得られます。

### 索引付けスレッドの使用法

複数の索引付けスレッドが同じコレクションで処理する場合、スレッド間で処理を同期するのに必要な調整が生じて、効果が削減されます。また、単一スレッド化された索引付けスレッドは、解析する間には効率よく処理しますが、マージやディスクへの書き込みの際には、パフォーマンスが低化する可能性があります。例えば、4つの異なるテキスト索引に対して4つの索引付けスレッドで処理する方が、1つのテキスト索引に対して4つの索引付けスレッドで処理するよりもスループットは向上します。

## 索引付けスレッドの数

索引付けスレッドの数は 2 つ以上で、かつ使用可能な CPU の数を超えないようにする必要があります。スレッドの共有を防止するには、並列して行う索引更新の最大数が、索引付けスレッドの数を超えないようにする必要があります。索引付けスレッドの数が多すぎたり、並行する索引更新の数が多すぎたりすると、プロセス・コンテキスト・スイッチのメモリーの使用量のために、システム全体のパフォーマンスが低化します。

例えば、頻繁に更新されるテキスト索引が 40 個あり、システムに CPU が 8 個ある場合、索引付けスレッドの数を 8 個より多くしてはなりません。また、テキスト索引の更新スケジュールをずらして、索引スレッドの競合を最小にします。

索引付けスレッド数に対するデフォルトの設定値は 4 です。これと同じデフォルト設定値が、索引前処理スレッドにも適用されます。

索引付けスレッド数を構成するには、次のコマンドを発行します。

```
configTool configureParams -configPath <full-path-to-configuration-folder>  
-numberOfIndexerThreads <value>
```

ここで、<value> はスレッド数で、<full-path-to-configuration-folder> は、DB2 Text Search サーバーの config.xml ファイルの絶対パスです。

前処理スレッド数を構成するには、次のコマンドを発行します。

```
configTool configureParams -configPath <full-path-to-configuration-folder>  
-numberOfPreprocessingThreads <value>
```

ここで、<value> はスレッド数で、<full-path-to-configuration-folder> は、DB2 Text Search サーバーの config.xml ファイルの絶対パスです。

## DB2 Text Search のキュー・メモリー・サイズ

DB2 Text Search のキュー・メモリー・サイズは、最適な索引更新処理が行われるように適切に設定されていなければなりません。キュー・メモリーの割り当ての制御は、データベースおよびテキスト・サーバーの両方に対して行うことができます。

データベースのキュー・メモリーによって、更新処理のためにどの時点においてもテキスト・サーバーに送信できる文書の数が決まります。データベースのキュー・メモリー・サイズを制御するには、SYSIBMTS.TSDEFAULTS 管理ビューを更新して、**DocumentResultQueueSize** パラメーターに値を設定します。デフォルト値は 10,000 です。この値は、1 つのコレクションの更新操作に予約されるデータベース・メモリー量を制限するために使用されます。複数パーティション・セットアップの場合、並列実行用に構成された単一のテキスト索引更新では、更新が必要なコレクションごとにメモリー・スペースが予約されます。

キュー・メモリー制御の 2 番目のメカニズムは、テキスト・サーバーに対して適用されます。2 つの構成値によって、キュー・メモリーの使用量が決まります。

- *inputQueueMemorySize*:

索引付けサーバー上の入力キューのメモリー・サイズを指定します。入力キュー

は、前処理のために待機している文書を格納します。メモリー・サイズを大きくすると高速になりますが、消費するリソースも大きくなります。デフォルトのサイズは 15 MB です。

- *outputQueueMemorySize*:

索引付けサーバー上の出力キューのメモリー・サイズを指定します。出力キューは、前処理後に索引付けのために待機している文書を格納します。メモリー・サイズを大きくすると高速になりますが、消費するリソースも大きくなります。デフォルトのサイズは 15 MB です。

入出力キューのメモリー・サイズとヒープ・メモリーの比率を考慮してください。キューのサイズは、キュー内の文書のメモリー使用量によって決まります。長い文書 (例えば、それぞれ 20 MB) を処理する場合、キュー・メモリー・サイズを大きくすること、およびヒープ・サイズを大きくすることを考慮してください。

例えば、*inputQueueMemory* サイズを変更するには、次のコマンドを発行します。

```
configTool configureParams -configPath <full-path-to-configuration-folder>
-inputQueueMemorySize <value>
```

ここで、<value> はメモリー・サイズで、<full-path-to-configuration-folder> は、DB2 Text Search の config.xml ファイルの絶対パスです。

## DB2 Text Search 索引の計画および最適化

データ・ソースの特性は、パフォーマンスに重大な影響を与えます。

テキスト索引の更新を完了するのに必要な時間は、主に以下の要因に依存します。

- 索引を付けられる文書の数
- 文書のサイズ
- 索引のタイプ
- 索引更新の並列性
- テキスト検索サーバーの構成

各文書の処理時間は、大体の固定時間と可変時間の合計です。固定時間は、プレーン・テキスト、XML、INSO などの文書タイプの影響を受けます。メモリーの使用や再利用の時間に多少の変動があるため、固定時間は概算です。可変時間は、主に文書サイズや言語学的処理の変動によって決まります。

INSO 文書の索引の場合、異なる MIME タイプの処理も処理時間に影響します。

文書のサイズが小さいと、一定の時間内に処理できる文書数は増えます。しかし、文書あたりの固定コストがあるため、全体のスループットは文書が大きいよりも小さい方が悪くなります。

### DB2 Text Search 索引ソースの特性

索引付けまたは検索のパフォーマンスを向上させるには、次の手法を使用します。

- 主キー列には、VARCHAR タイプの代わりに INTEGER などの数値データ・タイプを使用します。複数の VARCHAR 列から成る複合主キーは避けて、照会結果のトラフィックを最小化します。



- システムに、索引の更新操作に使用できる十分な量の実メモリを確保します。索引の更新には、データベース・バッファ・プールに必要なメモリに加えてさらにメモリが必要になります。メモリが不足していると、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ため、検索パフォーマンスが大幅に低下します。
- テキスト検索サーバーの索引更新で、多数の小さい文書进行处理する必要がある場合は、並列して行う索引更新の数を減らし、その代わりにキュー・サイズを増やして、テキスト・サーバーへの最大文書フローを大きくすることを検討してください。詳しくは、キャパシティー・プランニングのトピックを参照してください。
- イベント表に書き込まれるエラー・メッセージや警告メッセージの数が多いと、索引更新のパフォーマンスが低下することがあります。このため、索引付けする内容がアクセス可能であり、そのフォーマットが正しいことを確認してください。

## 非同期索引の更新

パフォーマンスの向上のために、テキスト検索索引とそれに関連付けられたユーザー表は、その表のテキスト文書を更新または削除する、またはその表にテキスト文書を挿入する DB2 トランザクションの範囲内では、同期されません。その代わりに、テキスト検索索引は非同期で更新されます。

テキスト検索索引の非同期の更新を容易にするために、テキスト検索索引ごとにステージング表 (ログ表とも呼ばれる) が作成されます。テキスト表のトリガーは、テキスト検索索引が関連付けられているテキスト列に対する変更 (挿入、削除、更新など) を収集して、それらをステージング表に書き込みます。補助ステージング表を使用して、整合性処理によって認識される変更をキャプチャーする場合があります。テキスト検索索引の更新は、さらに後のステージで、手動更新または自動更新のどちらかで適用されます。更新は、索引のごく小さい部分のコピーに対して行われます。更新中にも索引に対する検索を実行できますが、同期が完了するまでは更新済みのテキスト検索にはアクセスできません。

`updateautocommit` オプションを使用する場合は、コミット・サイクルのたびに相当量のプロセッサ使用が必要なため、索引更新を完了するのにかかる合計時間が増加することに注意してください。この `updateautocommit` オプションの設定は、特に、初期更新の場合や、全行または大部分の行が対象となる更新の場合に有効です。この設定は、合計経過時間が長い場合に、システムやサーバーの障害が原因でロールバックが発生して、完了済みの作業が失われることを防止します。

## DB2 Text Search 索引の最適化

DB2 Text Search 索引の最適化を行うと、テキスト検索索引が圧縮され、索引付けおよび検索の速度が向上します。最適化によって、削除済みの文書がテキスト検索索引から削除され、ディスク上の索引セグメント・ファイルがマージされます。

同じ索引に対して最適化と索引付けを並列して行うことはできません。最適化と索引付けのセッションをスケジュール設定する際には、このことを考慮に入れてください。ただし、最適化と検索は、並列して行うことができます。索引の最適化中は、ディスク・スペース消費量が高くなる可能性があります。特に同じ索引に対して検索を並行して行っている場合に高くなります。

索引の最適化は、文書セット全体に索引付けを行った後、またはインクリメンタル索引更新の後に実行することができます。索引の最適化には長時間かかる場合があります。これは索引のサイズによって異なります。インクリメンタル更新によって頻繁に文書が追加される場合は、最適化の実行頻度を少なくして、最適化プロセスによる余分なプロセッサ使用を最小化してください。

索引を最適化するには、以下のようにします。

1. `ECMTS_HOME/bin` ディレクトリーから、**optimizeIndex** コマンドを指定して管理ツールを開始します。例:

```
adminTool.bat optimizeIndex -configPath
"C:%Program Files%IBM%ECMTextSearch%config"
-collectionName MyCollection
```

2. **optimizeIndexStatus** コマンドを指定して管理ツールを実行すると、最後に実行した最適化プロセスの状況を確認できます。

## ディスク使用量

### テキスト索引サイズ

テキスト検索索引が使用するディスク・スペースの量は、各文書のテキストの状態に大きく依存します。しかしながら、テキスト検索索引に必要なディスク・スペースと元のデータに必要なディスク・スペースの間には、ほぼ比例する関係があります。通常、ディスク上の索引のサイズは元のテキスト・サイズの 50 - 150% になります。例えば、整数の主キーを含む表での、100,000 個の 20 KB の文書のためのテキスト検索索引には、約 1100 MB のディスク・スペース (100,000 x 20 KB x 55%) が必要になることが予想されます。ソース文書に比較したテキスト検索索引のサイズは、以下の要因に依存します。

- 文書の平均サイズ
- 文書キーのサイズ (主キー列)
- ソート可能フィールドの数
- 固有用語の数と分布

索引の更新中、追加のワークスペースが必要です。中間スペース所要量は、最大セグメント・サイズに達しない限り、最終のテキスト検索索引サイズの約 2、3 倍になります。必要なフリー・スペースは、最大セグメント・サイズの 2、3 倍です。古いセグメントが検索で使用された場合、セグメントをマージした後もディスク・スペースが予約されます。

### ログ・ファイル

`db2diag.log` ファイルに加えて、DB2 Text Search はトレースおよび DB2 Text Search サーバーからのメッセージを含む構成ツール・ログ・ファイルを生成します。

統合 Text Search サーバーの場合、デフォルトのログ・ファイル・ロケーションは `db2tss/log` ディレクトリーです。DB2 データベースとテキスト検索ログのロケーションを同じにするには、そのロケーションを `<instanceHome>/sql1lib/db2dump/`

tslog (UNIX プラットフォームの場合) または  
<instanceProfilePath>%<instance\_name>%db2tss%tslog (Windows プラットフォームの場合) に設定してください。

スタンドアロン・セットアップの場合、DB2 Text Search サーバー・ログのデフォルト・ロケーションは <ECMTS\_HOME>/log です。デフォルト・ロケーションは、インストール時に応答ファイルに **IA\_LOG\_PATH** パラメーターを設定することで変更可能です。

いずれの場合も、対象のロケーションにログ・ファイル用の十分な空きディスク・スペースがあることを確認してください。最小 100 MB の空きディスク・スペースが必要です。ログ・ファイル用の十分なスペースがないと、テキスト検索サービスはロギングを停止し、ディスク・フル・エラーをスローします。

## 管理表

**CREATE INDEX FOR TEXT** コマンドを実行する際、テキスト検索索引の管理表に対して表スペースを指定していない場合、管理表は基本表がある表スペースに作成されます。適切なロケーションを決定する場合、以下の情報を考慮してください。

- テキスト索引のステージング表

ステージング表では、基本表におけるテキスト索引のインクリメンタル更新で更新された行への参照が保留されます。この表は更新ごとに自動的にクリーンアップされます。

Size =  
number of rows for index updates \* (length of primary key of base table + 18)

- テキスト索引のイベント表

イベント表では、テキスト索引処理についての状況情報が扱われます。この情報には、索引更新中のエラーや警告などが含まれます。最悪の場合、致命的ではないエラーで各文書が拒否されると、イベントの数は、文書の数に更新処理の begin および end メッセージを加えた数になります。イベント表は自動的に消去されないため、**CLEAR EVENTS FOR INDEX** 命令が実行されるまでサイズが増加します。

Event table size =  
number of events \* (length of primary key of base table + 1050)

## DB2 Text Search 索引のロケーション

このリリースでは、デフォルトの索引のロケーションが変更されたことに注意してください。

統合 Text Search サーバーの場合、構成およびコレクションのメタデータは、UNIX の場合は *instanceHome/sqlllib/db2tss/config* に、Windows の場合は *instanceProfilePath%instance\_name%db2tss%config* に保管されます。

各テキスト検索索引のための構成およびコレクション・メタデータは、ほとんどスペースを必要としません。ただし、カスタム・パスが指定されていないかぎり、テキスト検索索引のロケーションは *db2tss/config* のサブディレクトリ内になります。多くの場合、このロケーションのサイズは制限されているため、統合 Text Search サーバーで複数の索引または大容量の索引を作成する予定の場合は、Text

Search サーバーの **defaultDataDirectory** パラメーターを十分なディスク・スペースがあるカスタム・ロケーションに設定することを強くお勧めします。

コレクション・データのロケーションは、コレクションの作成時に決定され、collection.xml ファイルに保管されます。スタンドアロン DB2 Text Search サーバーの場合、コレクション用の構成ファイルのロケーションは **defaultDataDirectory** パラメーターによって決定されます。デフォルトでは、コレクション構成ディレクトリは `<ECMETS_HOME>%config%collections` ですが、コレクション・データは **defaultDataDirectory%collection\_name%data%text** コレクション構成ディレクトリ内の下のサブディレクトリ内にあります。

いずれにしても、複数の大容量の索引を作成する予定の場合、特に並行する索引の更新がスケジュールされるのであれば、それらの索引を別個のまたはストライピングされたディスク装置に保管することを考慮してください。

## DB2 Text Search 索引更新に関する索引固有のパラメーター

コレクションに固有の次のパラメーターを構成して、パフォーマンスを向上させることができます。

- **MaxMergeDocs**
- **MaxMergeMB**
- **MergeFactor**
- **BufferSize**

特定のコレクションの索引付けパラメーターは、`ECMETS_HOME%config%collections%collection_name%collection.xml` ファイルを編集することで変更可能です。今後作成されるコレクションに対するデフォルト設定を変更するには、`ECMETS_HOME%config%defaults%collection.xml` ファイル内の上記のパラメーターの値を設定します。

- **MaxMergeDocs** パラメーターは、索引内の他のセグメントとマージできる最大セグメント (単位は文書数) を定義します。索引付けの全体的なスループットと、セグメントのマージ時間にはトレードオフの関係があります。

**MaxMergeDocs** パラメーターに小さい値 (例えば 100,000 文書) を指定すると、セグメントはサイズに関して制限を受けることになります。この場合、セグメントのマージはより高速になり、索引付けのフローはよりスムーズになり、タイムアウトは発生しません。ただし、内容が非常に多い場合は、セグメント数が非常に多くなるため、徐々に索引付けのスループットは低下します。

**MaxMergeDocs** パラメーターに大きい値 (例えば 100,000,000 や 500,000,000 文書) を指定すると、(索引が非常に大きくなるまでは) セグメント数が少なくなるため、索引付けの全体的なスループットは向上します。ただし、セグメントのマージ時間は長くなるため、索引付けでタイムアウトが発生する可能性があります。

一般的には、**MaxMergeDocs** の値は、小さい文書のコレクションに対しては大きい値に、大きい文書のコレクションに対しては小さい値にする必要があります。

- **MaxMergeMB** パラメーターは、索引内の他のセグメントとマージできる最大セグメント (単位はファイルの物理的なサイズ) を定義します。

索引付けの全体的なスループットと、セグメントのマージ時間にはトレードオフの関係があります。**MaxMergeMB** パラメーターに小さい値、例えば 500 MB を指定すると、セグメントはサイズに関して制限を受けることになります。この場合、セグメントのマージはより高速になり、索引付けのフローはよりスムーズになります。ただし、内容が非常に多い場合は、セグメント数が非常に多くなるため、徐々に索引付けのスループットは低下し、検索のパフォーマンスも低下します。

**MaxMergeMB** パラメーターに大きい値、例えば 50,000 MB や 100,000 MB を指定すると、(索引が非常に大きくなるまでは) セグメント数が少なくなるため、索引付けの全体的なスループットは向上します。ただし、セグメントのマージ時間は長くなるため、索引付けでタイムアウトが発生する可能性があります。

- **MergeFactor** パラメーターは、一度にマージするセグメントの数を定義します。また、索引に蓄積できるセグメントの総数も制御します。高い頻度で少量のマージ (例えば一度に 2 個) を行うか、低い頻度で大量のマージ (例えば一度に 10 個) を行うかというトレードオフの関係があります。**MergeFactor** パラメーターに小さい値を指定すると、タイムアウトを防止できます。このマージ係数を変更しても、通常、パフォーマンスに影響はありません。
- **BufferSize** パラメーターは、追加された文書をバッファリングする際に使用できる RAM の容量を指定します。これを超えると、文書は新規セグメントとしてフラッシュされます。高い頻度で少量ずつディスクにフラッシュするか、低い頻度で大量にディスクにフラッシュするかというトレードオフの関係があります。場合によっては、**BufferSize** パラメーターの値を増やすと、パフォーマンスを向上させることができます。例えば、小さい文書の単一のコレクションを索引付けする際にこのバッファ・サイズを大きくすると、特に索引の最初の 100,000 文書についてパフォーマンスが向上します。

## DB2 Text Search システムのチューニング

テキスト索引の更新処理のパフォーマンスおよびテキスト検索照会のパフォーマンスは、さまざまなシステム特性によって影響を受けます。

次の点を考慮してください。

- Windows 用の TCP/IP ポートの考慮事項
- ファイル記述子

### DB2 Text Search と Windows 用の TCP/IP ポートの考慮事項

32 ビット Windows オペレーティング・システムでは、TCP/IP ポートの数や、ポートを再び使用するまでの待ち時間によって、負荷の高い照会を処理する機能に影響が出ます。

### Windows (32 ビット) におけるポートの割り当て

データベース・サーバーと同じホスト上の個別のプロセスとして、統合された DB2 Text Search を実行します。データベース・サーバーとテキスト・サーバーは、TCP/IP 接続で通信します。

TCP/IP 接続で使用可能なポートの数は、ポートの数や、接続を閉じた後にポートを再び使用するまでの待ち時間によって左右されます。これらのパラメーターのデフ

オルト構成値では、負荷の高い照会を処理するのに十分な使用可能なポートを供給することができない可能性があります。TCP/IP ポートが少なすぎると、CIE00756 Connection failed エラーを受け取る可能性があります。

CIE00756 Connection failed エラーが発生した場合、以下のコマンドを実行してサーバーのポートの使用状況を表示してください。

```
netstat -n  
netstat -n | c:\windows\system32\find /I <port_number>
```

出力に多数の TCP/IP 接続およびローカル・アドレス 127.0.0.1:port\_number が TIME\_WAIT 状態にあることが示される場合、サーバーで TCP/IP ポートの数が足りなくなっている可能性があります。

DB2 Text Search のポート番号は次のコマンドを発行することにより判別できます。

```
configTool printAdminHTTPPort -configPath %INSTPROF%\%DB2INSTANCE%\db2tss\config
```

ここで、INSTPROF には、統合 DB2 Text Search サーバー・セットアップに適用される DB2INSTPROF レジストリー変数の値を設定します。

## ポート設定

ポート設定は、

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\TCPIP\Parameters にある以下のレジストリー項目によって制御されます。

- **TcpTimedWaitDelay**

TCP/IP が閉じた接続を解放し、そのリソースを再使用できるようになるまで経過する時間 (秒) を決定する DWORD 値です。値は、30 から 300 の範囲で指定します。TcpTimedWaitDelay の値を小さく設定して、ソケットが TIME\_WAIT の状態にとどまる時間を少なくします。

- **MaxUserPort**

アプリケーションが使用可能なユーザー・ポートを要求するときに、TCP/IP が割り当てることができる最も大きなポート番号を決定する DWORD 値です。MaxUserPort 値を大きく設定して、ポートへ接続できるソケットの総数を増やします。

多くの接続要求を出すシステムでは、TcpTimedWaitDelay に 30 秒、MaxUserPort に 32678 を設定するとパフォーマンスが向上する可能性があります。

これらのレジストリー項目を追加または変更した後、Windows マシンをリブートし、変更を反映します。

## DB2 Text Search ファイル記述子

DB2 Text Search の索引更新および索引照会では、ファイル記述子などのシステム・リソースを消費して、複数の索引更新要求および索引検索要求を処理します。

標準的なシステムでは、プロセスごとのオープン・ファイル記述子の数が 1024 などの比較的小さな数に制限されることがあるため、テキスト検索サーバーがファイル記述子を使い尽くす場合があります。これが生じると、検索および更新要求は失敗します。

このエラーを解決するには、次のようにします。

- サーバー・ログに「オープン・ファイルが多すぎます」というようなメッセージ・ストリングの例外がないか調べます。
- UNIX システムの場合は、`ulimit -a` でシステム制限を確認します。

ファイル記述子を増やすには、次のステップに従ってください。

1. テキスト検索サーバーをシャットダウンします。
2. オペレーティング・システムのマニュアルに従って、プロセスごとのファイル記述子の数を増やします。ファイル記述子の数は、全ログイン・セッションの全要求に対応するのに十分な数に増やす必要があります。
3. テキスト検索サーバーを再始動します。

---

## DB2 Text Search 照会の計画

テキスト検索照会を計画する際に、考慮すべき要素がいくつかあります。

### DB2 Text Search の引数

ワイルドカード文字とそれらの展開制限、引数の大/小文字の区別、および引数オプションは、それぞれ別のタイプのテキスト検索引数であり、すべて照会パフォーマンスに影響を与える可能性があります。

#### ワイルドカード文字

検索語の先頭にワイルドカード文字を使用すると、照会の処理速度が低下します。可能であれば、`*search_term` や `?search_term` などの検索を実行することは避けてください。

#### ワイルドカードの展開制限

照会用語にワイルドカードが含まれていると、その照会用語は、一致文書を取得するために展開されます。テキスト索引コレクションには、ワイルドカード展開制限が許可するより多くの一致する用語が含まれている場合があります。この場合、既に展開された用語に一致する文書のサブセットのみが戻されます。この制限は、アスタリスク (\*) のワイルドカード文字に適用されます。

デフォルトでは、1024 個の用語を戻すことができます。この制限を変更するには、`queryExpansionLimit` パラメーターとパラメーターの値を `ECMTS_HOME\config\config.xml` ファイルに指定します。例えば、制限を 4096 個に設定するには、次の行をファイルに追加します。

```
<queryExpansionLimit>4096</queryExpansionLimit>
```

## 大/小文字の区別

テキスト検索指数では、二重引用符を使用して完全一致の用語や句を指定した場合でも、大/小文字の区別はありません。例えば、用語 "Hamlet" を検索すると、Shakespeare の演劇 Hamlet と、小さな村を意味する用語 hamlet の両方が戻されません。

## 検索指数のオプション

検索指数のオプションとは、検索指数のプロパティです。例えば、bank という語に対する以下の検索照会では、QUERYLANGUAGE 検索指数のオプションが異なっています。

```
...CONTAINS(column, 'bank', 'QUERYLANGUAGE=en_US')  
および CONTAINS(column, 'bank', 'QUERYLANGUAGE=de_DE')...
```

## DB2 Text Search における複数の述部

1 つの照会に複数の述部が含まれている場合は、それらの述部の編成方法に応じて次の制限があることに注意してください。

### 演算子 UNION と OR

複数の述部を結合する場合、OR ではなく UNION を使用すると照会パフォーマンスが高くなる場合があります。

### JOIN の使用

テキスト検索関数は、外部結合の述部に使用できますが、LEFT OUTER JOIN および FULL OUTER JOIN の場合には制限があります。これらの場合、テキスト検索述部は、このテキスト索引の検索結果が基本表の主キーと結合できる場合のみ適用できます。例えば、次のタイプの照会がサポートされます。

```
select place.placenum, location.description from place  
LEFT OUTER JOIN location on (location.mgrid = place.ownerid)  
where  
(location.description is null and contains(place.description, 'Paris')=1 )
```

CONTAINS 関数および SCORE 関数を、LEFT OUTER JOIN または FULL OUTER JOIN の述部として使用することはサポートされていません。

## DB2 Text Search のロケールおよび言語

ロケールの指定は、テキスト検索照会のパフォーマンスにも影響があります。

### ロケールの指定

マルチリンガル環境でテキスト検索索引を使用して検索を実行する場合は、常に検索照会に QUERYLANGUAGE オプションを使用して、検索語の解釈に使用するロケール(言語と地域情報の組み合わせ)を指定することをお勧めします。例えば、bald などの検索語がある場合、検索照会で QUERYLANGUAGE=en\_US を設定して、それを英語のワードとして扱うように指定できます。同様に、それをドイツ語のワードとして扱うためには、QUERYLANGUAGE を de\_DE に設定します。ただし、戻される結果は、照会で指定される QUERYLANGUAGE とは関係なく、索引付けに使用される LANGUAGE に大きく依存することに注意してください。



検索照会で QUERYLANGUAGE が指定されない場合には、以下の論理が使用されます。

- 検索語のロケールは、基礎となるテキスト索引に対して索引の作成時に設定されたロケールと同じであると解釈されます。
- 索引の作成時に索引に設定されたロケールが AUTO の場合、これはデフォルトの英語 (en\_US) となり、検索語は英語のワードとして扱われます。

制約事項:

- 検索照会で指定されたロケールが無効である場合 (QUERYLANGUAGE=Mongolian など)、その照会は無効と見なされて、例外がスローされます。
- 検索照会で QUERYLANGUAGE=AUTO を設定することはサポートされないオプションであり、照会の結果は定義されていません。

QUERYLANGUAGE で指定されるロケールは、検索照会から生じるエラー・メッセージのロケールには影響を与えないことに注意してください。使用されるエラー・メッセージ・ロケールは、テキスト検索インスタンスのサービスを開始したかどうかによって異なります。それらを開始していない場合、メッセージは en\_US を使用して記述されます。それらを開始した場合、メッセージが記述されるロケールは **START FOR TEXT** コマンドを発行した環境のロケールと同じになります。

## DB2 Text Search の SCORE 関数

文書のスコアは動的で、照会を行うたびに個別に計算されます。

文書の更新、およびテキスト索引に対する文書の追加や削除によって、照会用語に対する文書のスコアは変化する可能性があります。

交通 (transportation) と汚染 (pollution) について論じている一連の文書があるとしめます。pollution と transportation の両方の用語の言及を含んでおり、用語 transportation よりも用語 pollution のスコアが高い文書のみを検索するには、次のコマンドを使用します。

```
SELECT document_id
FROM document_library
WHERE SCORE(document_content, 'pollution') >
SCORE(document_content, 'transportation')
and CONTAINS(document_content, 'transportation pollution') = 1
```

パフォーマンスを上げるには、次のようにしてランキング調整 (^) 修飾子を使用するように照会の形式を設定して、検索関数の実行を 1 回だけにすることができます。

```
SELECT document_id
FROM document_library
WHERE SCORE(document_content, 'pollution^10 transportation') > 0
ORDER BY SCORE(document_content, 'pollution^10 transportation') DESC
```

pollution のスコアが低い場合、最初の照会では何も結果が戻されません。2 番目の照会では pollution の重要度が高く設定されますが、すべての文書で pollution のスコアが低い場合に文書が戻されます。

## DB2 Text Search の RESULTLIMIT 関数

1 つの照会内に **RESULTLIMIT** を複数回指定している場合は、同じ検索引数を使用して、予測可能な結果が得られるようにする必要があります。

## 説明

同じ照会で **RESULTLIMIT** を指定する複数のテキスト検索を使用する場合、同じ検索引数を使用してください。異なるテキスト検索引数を使用すると、予想する結果が戻されないことがあります。

例えば、次の照会では、**RESULTLIMIT** で指定された 10 の文書が戻されるかどうかは予測不能です。

```
SELECT EMPNO
FROM EMP_RESUME WHERE RESUME_FORMAT = 'ascii'
AND CONTAINS(RESUME, '"ruby on rails"', 'RESULTLIMIT=10') = 1
AND CONTAINS(RESUME, '"java script"', 'RESULTLIMIT=10') = 1
```

代わりに、以下のように **RESULTLIMIT** を使用してください。

```
SELECT EMPNO
FROM EMP_RESUME WHERE RESUME_FORMAT = 'ascii'
AND CONTAINS(RESUME, '"java script" "ruby on rails"', 'RESULTLIMIT=10') = 1
```

この方式が有効なのは、両方の **CONTAINS** 関数が同じ表列で機能するときだけであることに注意してください。それらが同じ列で機能していない場合には、照会のパフォーマンスを改善するために **FETCH FIRST n ROWS** を使用してみてください。

## DB2 Text Search のパーサー構成

XML 検索に使用できるいくつかの設定を構成できます。

すべてのパーサー構成パラメーターは、`parser_config.xml` ファイルの、パーサーを定義する XML 要素 `com.ibm.es.nuvo.parser.xml.XMLParser` 内にあります。各パラメーターは、**Parameter** 要素によって次の形式で指定されます。

```
<Parameter Name="parameter">setting</Parameter>
```

### ParserName: text

#### ParserClass: com.ibm.es.nuvo.parser.text.TextParser

コンテンツ・タイプがテキストのときに呼び出されるクラス。

#### required.text.confidence

使用されていません。

#### fail.back.parser

テキストのパーサーが失敗し、コンテンツ・タイプが不明として指定され、コンテンツの検出でコンテンツをバイナリーと特定した場合にアクティブにされるパーサー。

#### fail.back.encoding

エンコードが不明または **NULL** として指定されている場合に使用されるエンコード。

#### detection.encoding.buffer.size

エンコードを判別するためにコンテンツ検出メカニズムに渡されるバッファー・サイズ (バイト単位)。デフォルトは 2000 バイトです。

### ParserName: xml

**titleTagNameList**

タイトル・フィールドとして処理されるタグのコンマ区切りのリスト。

**maxTextUnicodeChars**

使用されていません。

**handleExternalFiles**

使用されていません。

**handleSkippedEntities**

使用されていません。

## DB2 Text Search の XML 名前空間

XML 名前空間の検索には回避策が必要です。

名前空間バインディングを含む XML 文書にエラーを生じさせないで索引を付けることができますが、名前空間情報は各タグから除去されます。その結果、名前空間バインディングのある XML 文書でのテキスト検索によって、好ましくない結果が生じることがあります。

ただし、DB2 XQuery を使用する照会では、この制約に対する回避策があります。DB2 Text Search エンジン は名前空間を認識しませんが、ネームスペースのための DB2 XQuery サポートを使用して、テキスト検索から戻される不要な文書に対する名前空間のフィルター操作を行うことができます。

デフォルトのデータベース環境変数が SAMPLE に設定されて、prod\_desc\_idx という名前のテキスト検索索引が PRODUCT 表に作成される、以下の例を検討してください。

```
db2ts "ENABLE DATABASE FOR TEXT"
db2ts "CREATE INDEX prod_desc_idx FOR TEXT ON product(description)"
```

ここで、名前空間 <http://posample.org/wheelshovel> のある新しい行が PRODUCT 表に追加されます。この表には、名前空間 <http://posample.org> のある 2 つの XML 文書が既に存在します。

```
INSERT INTO PRODUCT VALUES ('100-104-01', 'Wheeled Snow Shovel',
99.99, NULL, NULL, NULL, XMLPARSE(DOCUMENT '<product xmlns=
"http://posample.org/wheelshovel" pid="100-104-01">
<description><name>Wheeled Snow Shovel</name><details>
Wheeled Snow Shovel, lever assisted, ergonomic foam grips,
gravel wheel, clears away snow 3 times faster</details>
<price>99.99</price></description></product>'))
```

その後、以下のようにテキスト検索索引が更新されます。

```
db2ts "UPDATE INDEX prod_desc_idx FOR TEXT"
```

デフォルト要素を <http://posample.org> として指定する以下の XQuery 式は、ワード ergonomic を含む一致する XPath の /product/description/details を持つすべての文書を戻します。

```
xquery declare default element namespace "http://posample.org";
db2-fn:xmlcolumn-contains('PRODUCT.DESCRPTION', '@xmlns:
'/product/description/details [. contains ("ergonomic")]''')
```

3 つの文書が戻されます。その 2 つは名前空間 `http://posample.org` があるために予期されるものであり、1 つは名前空間 `http://posample.org/wheelshovel` があるために予期されないものです。

次の XQuery 式は、DB2 Text Search エンジンから戻される文書をフィルターに掛けるための XML 検索および名前空間の DB2 XQuery サポートを使用して、名前空間が `http://posample.org` の文書だけが戻されるように、パス式 `/product/..` を使用します。

```
xquery declare default element namespace "http://posample.org";
db2-fn:xmlcolumn-contains('PRODUCT.DESCRPTION', '@xmlns:
'/product/description/details [. contains ("ergonomic")]''')/product/..
```

**注:** SQL 照会は、DB2 XQuery を使用してネーム・スペースのフィルター処理を強制できます。前に示した例の場合、SQL 照会を使用する対応する式は次のようになります。

```
xquery declare default element namespace "http://posample.org";
db2-fn:sqlquery("select description from product where
contains(description, '@xmlns:'/product/description/details
[. contains ("ergonomic")]''') = 1")
```

回避策は、以下のようになります。

```
xquery declare default element namespace "http://posample.org";
db2-fn:sqlquery("select description from product where
contains(description, '@xmlns:'/product/description/details
[. contains ("ergonomic")]''') = 1")/product/..
```

同様に、(直前に示した照会のように一致する文書を戻すだけでなく) 文書内の特定の要素にアクセスするには、以下の照会を使用できます。

```
xquery declare default element namespace "http://posample.org";
db2-fn:xmlcolumn-contains('PRODUCT.DESCRPTION', '@xmlns:
'/product/description/details [. contains ("ergonomic")]''')
/product/description[price > 20]/name
```

**注:** この回避策には限界があり、例えば 1 つの文書内に複数の製品要素がある場合などに、予想どおりに機能しないことがあります。

---

## 第 4 章 DB2 Text Search のインストールおよび構成

DB2 Text Search は、オプションでインストール可能なコンポーネントであり、そのインストールと構成はすべての DB2 データベース・サーバー製品に完全に統合されています。

DB2 インストーラーで DB2 Text Search が自動的にインストールおよび構成されるようにすることができます。行う必要のあるステップは、プラットフォームによって異なります。46 ページの図 11 は Windows オペレーティング・システム上でのインストールおよび構成の手順を説明し、47 ページの図 12 は Linux および UNIX オペレーティング・システム上での手順を説明しています。

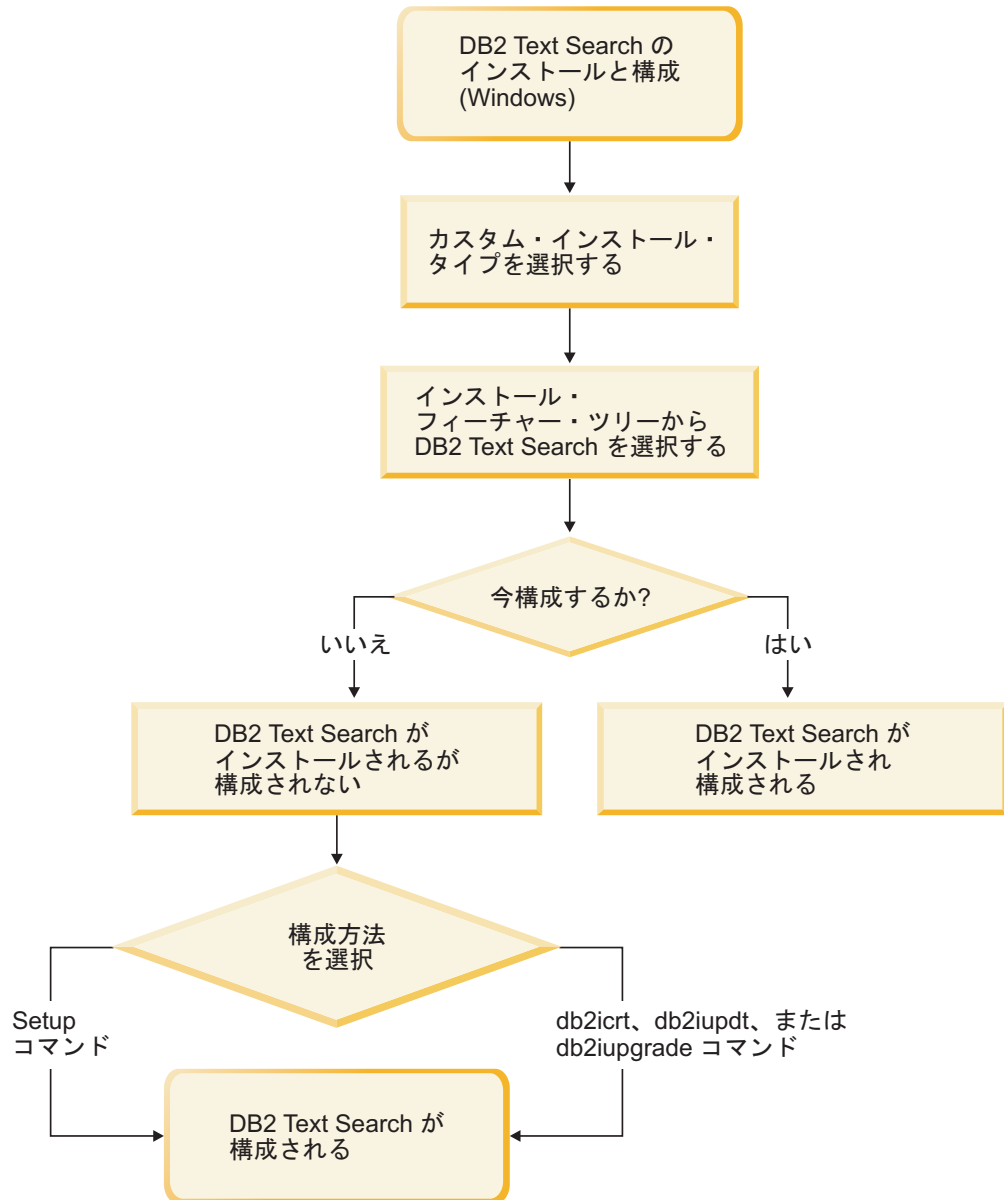


図 11. Windows プラットフォームでのインストールおよび構成

Windows では、インストール・タイプを選択して、構成するかどうかを決め、構成方法を選択します。

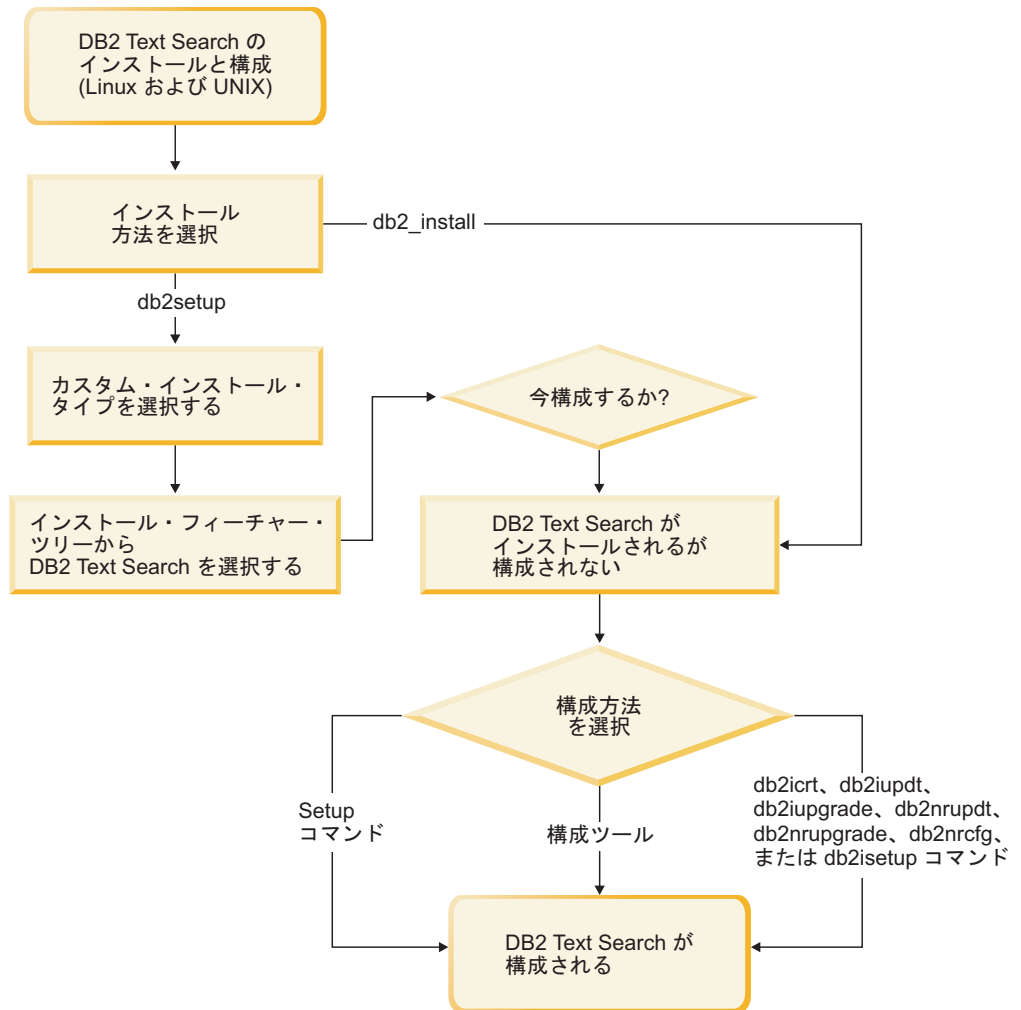


図 12. Linux および UNIX プラットフォームでのインストールおよび構成

Linux および UNIX では、インストール方法とタイプを選択して、構成するかどうかを決め、構成方法を選択します。 **db2setup** を非 root ユーザーとして実行する場合、後で (SYSADM 権限を持つ) システム管理者が **DB2RFE** コマンドを実行して、サービス・ファイルで必要なポート番号を予約するようにします。

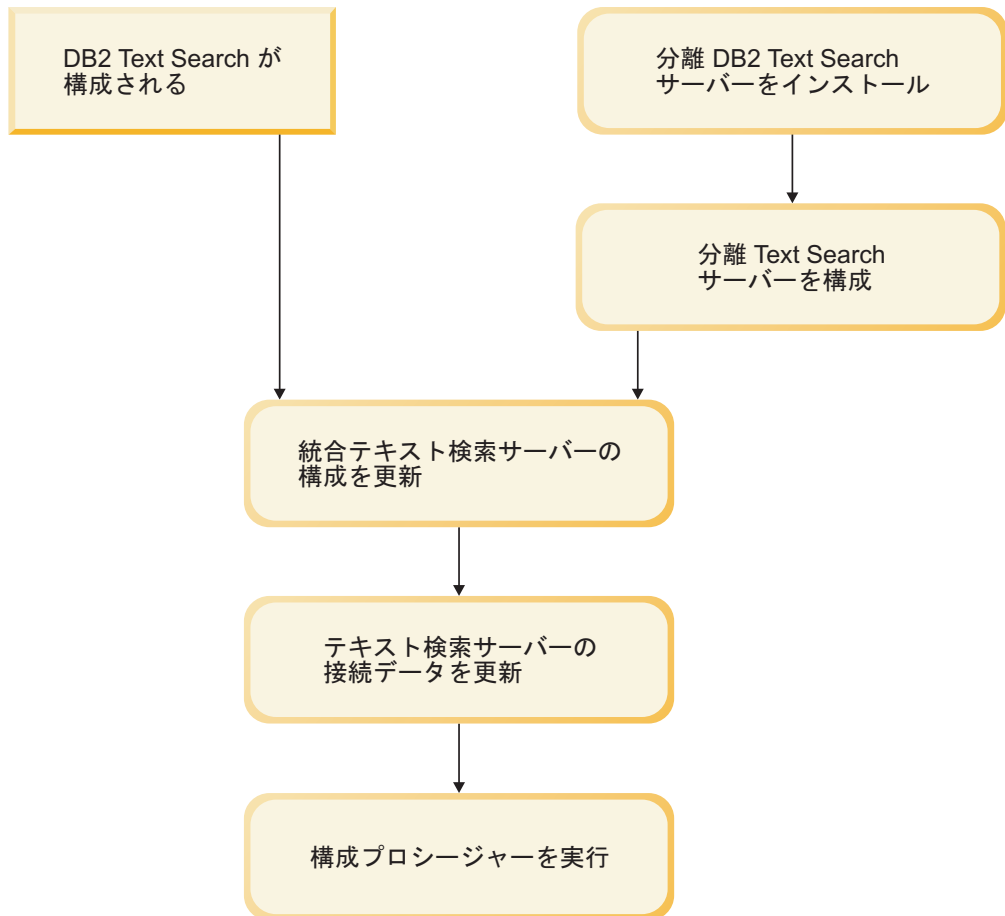


図 13. スタンドアロン DB2 Text Search サーバーの構成

スタンドアロン DB2 Text Search サーバーの場合は、統合テキスト検索サーバーの構成を更新します。次に、サーバー接続データを更新して、**CONFIGURE** プロシージャを実行します。

DB2 Text Search には、次の制約事項があります。

- DB2 セットアップ・ウィザードを使用してデータベース・パーティション・インスタンスを作成する場合、調整メンバーまたはインスタンス所有パーティションで行う必要があります。
- DB2 Text Search は DB2 pureScale環境ではサポートされていません。

## DB2 Text Search のハードウェアおよびソフトウェアの要件

### ソフトウェア・プラットフォーム

DB2 Text Search は、以下のオペレーティング・システムのプラットフォームでサポートされます。

- AIX バージョン 6.1
- HP-UX 11i (Itanium ベースの HP Integrity シリーズ・プラットフォーム)
- Red Hat Enterprise Linux Server 5 (x86 および x64 プラットフォーム)



- Red Hat Enterprise Linux Server 6 (x86 および x64 プラットフォーム)
- Solaris 10 (UltraSPARC および x64 プラットフォーム)
- SUSE Linux Enterprise Server 10 (x86 および x64 プラットフォーム)
- SUSE Linux Enterprise Server 11 (x86 および x64 プラットフォーム)
- Windows Server 2003 (x86 および x64 プラットフォーム)
- Windows Server 2008 (x86 および x64 プラットフォーム)

**重要:** Linux オペレーティング・システム上に libstdc++.so.5 共有ライブラリーをインストールする必要があります。

スタンドアロン DB2 Text Search サーバーは、HP-UX 11i、および Solaris 10 x64 のオペレーティング・システムを除き、前述のプラットフォームで使用できます。クロスプラットフォームの使用がサポートされます。これらのプラットフォーム上の DB2 データベース・インスタンスを、サポートされるプラットフォームでスタンドアロン DB2 Text Search サーバーを使用するように構成できます。

## ハードウェア要件

DB2 Text Search の最小ハードウェア要件は以下のとおりです。

表 2. DB2 Text Search のハードウェア要件

DB2 Text Search サーバー	プロセッサ	RAM / メモリー	ディスク
統合セットアップ  (DB2 データベース・サーバーの要件に加えて)	2 デュアルコア 2.66 GHz	4 GB	一時ワークスペースを含め、各テキスト検索索引には、索引付けするすべての文書の約 4 倍のサイズが必要です。例えば、1 KB の文字サイズで 100 万行ある列のテキスト索引には、約 4 GB のディスク・スペースが必要になります。
スタンドアロンのセットアップ			

実際のディスク・スペース、メモリー、およびプロセッサの使用量は、コレクションの数、コレクションごとの文書の数、同時に索引付けされるコレクションの数、必要とされる索引付けスループット、および照会の負荷など、さまざまな要因によって異なります。詳しくは、DB2 Text Search のキャパシティー・プランニングのトピックを参照してください。

Linux や UNIX などのオペレーティング・システムにおける各種のユーザー・プロセスのリソース限界に関する情報を入手するようにお勧めします。インフォメーション・センターの該当するトピックを確認してください。これらの全般的なリソース制限の要件は、DB2 Text Search サーバーの統合およびスタンドアロンのどちらのセットアップにも適用されます。

---

## デフォルト構成の DB2 Text Search のインストール

### DB2 セットアップ・ウィザードによる DB2 Text Search のインストールおよび構成

DB2 Text Search のインストールは、DB2 データベース製品のカスタム・インストールの一部として、DB2 セットアップ・ウィザードを使用して行うことができます。

#### このタスクについて

DB2 データベース製品のカスタム・インストールを実行し、フィーチャー・ツリーから DB2 Text Search を選択します。DB2 Text Search が自動で構成されるようにすることも、後で手動で構成することもできます。DB2 セットアップ・ウィザードを使用してパーティション・インスタンスを作成している場合は、調整メンバーまたはインスタンス所有パーティションで行う必要があります。

#### 手順

**setup** または **db2setup** を使用して DB2 Text Search のカスタム・インストールを実行するには、以下のようにします。

1. お使いのプラットフォームに応じた指示にしたがって、DB2 サーバーをインストールします。
  - 「DB2 サーバー機能 インストール」の『DB2 セットアップ・ウィザードによる DB2 サーバーのインストール (Windows)』
  - 「DB2 サーバー機能 インストール」の『DB2 セットアップ・ウィザードによる DB2 サーバーのインストール (Linux および UNIX)』

フィーチャー・ツリーから DB2 Text Search コンポーネントを選択できます。インストール中に、DB2 Text Search をデフォルト・インスタンス用に構成するオプションがあります。DB2 Text Search を構成しない場合は、ステップ 2 はスキップします。

2. 自分で DB2 Text Search を構成する場合に、これらのフィールドに有効な値がなければ、有効なサービス名およびポート番号を入力してください。DB2 Text Search は、インストール直後に構成する必要はありません。後から構成することができます。後から構成を実行する方法の説明は、65 ページの『第 5 章 DB2 Text Search の構成』を参照してください。

### 応答ファイルによる DB2 Text Search のインストールおよび構成

DB2 Text Search のインストールおよび構成は、DB2 データベース製品のカスタム・サイレント・インストールの一部として行うことができます。このタイプのインストールでは、応答ファイルで **setup** または **db2setup** コマンドを使用します。

#### このタスクについて

DB2 Text Search をインストールするには、DB2 データベース製品のカスタム・インストールを実行します。DB2 Text Search をインストールして構成するには、応答ファイルにいくつかのキーワードを追加する必要があります。

## 手順

カスタム・インストールを実行するには、以下のようにします。

1. 以下に示す行を DB2 データベース製品をインストールするために使用する応答ファイルに追加します。

```
COMP = TEXT_SEARCH
```

2. インストール中に DB2 Text Search を構成するには、以下の行を応答ファイルに追加します。

- root インストールの場合のみ:

```
db2inst_name.TEXT_SEARCH_HTTP_SERVICE_NAME = db2j_db2inst_name
```

ここで、*db2inst\_name* は、DB2 インスタンスの名前、*db2j\_db2inst\_name* はサービス名です。

- root インストールおよび非 root インストールの場合:

```
db2inst_name.TEXT_SEARCH_HTTP_PORT_NUMBER = port-number
```

非 root インストールの場合に **TEXT\_SEARCH\_HTTP\_SERVICE\_NAME** キーワードの値を入力すると、エラーが戻されます。

使用されていない任意の有効なサービス名およびポート番号を指定できます。値を入力しないと、応答ファイル・キーワード

*db2inst\_name.CONFIGURE\_TEXT\_SEARCH* が YES に設定されている場合、構成にはデフォルト値が使用されます。

3. ご使用のプラットフォームに応じた指示にしたがって、DB2 データベース製品をインストールします。
  - 「DB2 サーバー機能 インストール」の『応答ファイルによる DB2 製品のインストール (Windows)』
  - 「DB2 サーバー機能 インストール」の『応答ファイルによる DB2 製品のインストール (Linux および UNIX)』

## 次のタスク

DB2 Text Search は、インストール直後に構成する必要はありません。後から構成することができます。後から構成を実行する方法の説明は、65 ページの『第 5 章 DB2 Text Search の構成』を参照してください。

## db2\_install を使用した DB2 Text Search のインストール (Linux および UNIX)

**db2\_install** コマンドの発行時に、DB2 Text Search もインストールされます。

### このタスクについて

**重要:** コマンド **db2\_install** は推奨されておらず、将来のリリースで削除される予定です。代わりに応答ファイルを指定して **db2setup** コマンドを使用します。

DB2 Text Search をインストールするには、「DB2 サーバー機能 インストール」の『**db2\_install** を使用した DB2 製品のインストール』で説明しているステップに

従ってください。DB2 Text Search は、DB2 データベース製品のインストールの一部として自動的にインストールされます。

これが非 root インストールである場合、DB2 インスタンスが作成され、DB2 Text Search がインストールされます。これが root インストールである場合は、DB2 インスタンスを作成し、使用可能な方法の 1 つで DB2 Text Search を構成する必要があります。

DB2 Text Search は、インストール直後に構成する必要はありません。構成を実行する方法の説明は、65 ページの『第 5 章 DB2 Text Search の構成』を参照してください。

---

## 初期構成なしの DB2 Text Search のインストール

### DB2 セットアップ・ウィザードを使用した DB2 データベース・サーバーのインストール (Windows)

このタスクでは、Windows 上で DB2 セットアップ・ウィザードを開始する方法を説明します。DB2 セットアップ・ウィザードを使用して、インストールを定義し、DB2 データベース製品をご使用のシステムにインストールします。

#### 始める前に

DB2 セットアップ・ウィザードを開始する前に、以下の事柄を行います。

- パーティション・データベース環境のセットアップを予定している場合は、『パーティション・データベース環境のセットアップ』を参照してください。
- ご使用のシステムがインストール、メモリー、およびディスクの各要件に合うことを確認します。
- LDAP を使用して、DB2 サーバーを Windows オペレーティング・システムの Active Directory に登録する予定であれば、インストールの前にディレクトリー・スキーマを拡張します。そうでない場合は、手動でノードを登録し、データベースをカタログする必要があります。詳しくは、『LDAP ディレクトリー・サービス用の Active Directory スキーマの拡張 (Windows)』のトピックを参照してください。
- インストールを実行するために推奨されるユーザー権限を持つ、ローカル管理者ユーザー・アカウントを持っている必要があります。LocalSystem を DAS および DB2 インスタンス・ユーザーとして使用できる、データベース・パーティション・フィーチャーを使用していない DB2 データベース・サーバーでは、システム特権を持つ非管理者ユーザーがインストールを実行できます。

**注:** 非管理者ユーザー・アカウントが製品のインストールを実行する場合、DB2 データベース製品のインストールを試行する前に VS2010 ランタイム・ライブラリーがインストールされている必要があります。DB2 データベース製品をインストールする前にオペレーティング・システムには VS2010 ランタイム・ライブラリーが必要です。VS2010 ランタイム・ライブラリーは、Microsoft ランタイム・ライブラリーのダウンロード Web サイトから入手できます。次の 2 つの選択が存在します。vcredist\_x86.exe (32 ビット・システム用) または vcredist\_x64.exe (64 ビット・システム用)

- 必須ではありませんが、リポートなしでインストール・プログラムがコンピューター上の任意のファイルを更新できるようにするために、すべてのプログラムを閉じることをお勧めします。
- DB2 製品を仮想ドライブまたはマップされていないネットワーク・ドライブ (例えば、Windows エクスプローラで `\\hostname\sharename` と表示されるもの) からインストールすることはサポートされていません。DB2 製品のインストールを試行する前に、ネットワーク・ドライブを Windows ドライブ名 (例えば、Z:) にマップする必要があります。

#### 制約事項

- どのユーザー・アカウントでも、DB2 セットアップ・ウィザードの複数のインスタンスを実行することはできません。
- DB2 コピー名とインスタンス名は、数値で始めることはできません。DB2 コピー名は、文字 A から Z、a から z および 0 から 9 で構成される 64 英文字に制限されています。
- DB2 コピー名とインスタンス名は、すべての DB2 コピーの間で固有でなければなりません。
- XML フィーチャーは、データベース・パーティションが 1 個のみであるデータベースでのみ使用できます。
- 以下のいずれかが既にインストールされている場合は、同じパスに他の DB2 データベース製品をインストールすることはできません。
  - IBM® Data Server Runtime Client
  - IBM Data Server Driver Package
  - DB2 インフォメーション・センター
- DB2 セットアップ・ウィザード・フィールドでは英語以外の文字を受け入れません。
- Windows Vista か Windows 2008、またはそれ以降で拡張セキュリティーを有効にする場合、ローカル DB2 コマンドとアプリケーションを実行するために、ユーザーは DB2ADMNS または DB2USERS グループに属している必要があります。これは、ローカル管理者にデフォルトで付与されている特権を制限する特別なセキュリティー・フィーチャー (ユーザー・アクセス制御) のためです。ユーザーがこれらのグループの 1 つに属していない場合、ローカル DB2 構成またはアプリケーション・データに対する読み取りアクセス権限が与えられません。

#### 手順

次のようにして、DB2 セットアップ・ウィザードを開始します。

1. DB2 インストール用に定義したローカル管理者アカウントで、システムにログインします。
2. DB2 データベース製品 DVD を所有している場合は、これをドライブに挿入します。自動実行フィーチャーを有効にしている場合、DB2 セットアップ・ランチパッドが自動的に開始されます。自動実行機能が作動しない場合は、Windows エクスプローラを使用し、DB2 データベース製品 DVD をブラウズして **setup** アイコンをダブルクリックし、DB2 セットアップ・ランチパッドを開始します。

3. DB2 データベース製品をパスポート・アドバンテージからダウンロードした場合は、実行可能ファイルを実行して DB2 データベース製品インストール・ファイルを解凍します。Windows エクスプローラを使用し、DB2 インストール・ファイルをブラウザして **setup** アイコンをダブルクリックし、DB2 セットアップ・ランチパッドを開始します。
4. DB2 セットアップ・ランチパッドから、インストールの前提条件およびリリース情報を表示することができます。あるいは、インストールに直接進むこともできます。後で追加されたインストール前提条件およびリリース情報を参照することもできます。
5. 「製品のインストール」をクリックすると、「製品のインストール」ウィンドウに、インストールに使用できる製品が表示されます。

既存の DB2 データベース製品がコンピューターにインストールされていない場合は、「新規インストール」をクリックして、インストールを起動します。DB2 セットアップ・ウィザードのプロンプトに従ってインストールを進めます。

既存の DB2 データベース製品が 1 つ以上コンピューターにインストールされている場合は、次のようにできます。

- 新しい DB2 コピーを作成するには、「新規インストール」をクリックします。
  - 既存の DB2 コピーの更新、既存の DB2 コピーへの機能追加、既存の DB2 バージョン 9.5 またはバージョン 9.7 のコピーのアップグレード、またはアドオン製品のインストールを実行するには、「既存の処理」をクリックします。
6. DB2 セットアップ・ウィザードは、システム言語を判別してから、その言語用のセットアップ・プログラムを立ち上げます。残りのステップについて説明しているオンライン・ヘルプを利用できます。オンライン・ヘルプを呼び出すには、「ヘルプ」をクリックするか、または **F1** を押します。「キャンセル」をクリックすれば、いつでもインストールを終了できます。
  7. DB2 セットアップ・ウィザードを使用する際のサンプル・パネルからインストール・プロセスに進みます。関連リンクを参照してください。

## タスクの結果

DB2 データベース製品がインストールされるデフォルトの場所は `Program_Files\IBM\sqllib` ディレクトリで、`Program_Files` は Program Files ディレクトリ内の場所を表します。

インストール先のシステムでこのディレクトリが既に使用中の場合、DB2 データベース製品のインストール・パスに `_xx` が追加されます。 `xx` は 01 で始まる数字で、インストール済みの DB2 コピーの数に応じて増加します。

独自の DB2 データベース製品のインストール・パスを指定することもできます。

## 次のタスク

- インストールを検証します。
- 必要なインストール後の作業を実行します。

インストール時に検出されるエラーの詳細については、My Documents¥DB2LOG¥ ディレクトリーにあるインストール・ログ・ファイルを確認してください。ログ・ファイルは DB2-ProductAbbrrev-DateTime.log という形式になります (例えば DB2-ESE-Tue Apr 04 17\_04\_45 2012.log)。

これが Vista 64 ビット上の新しい DB2 製品インストールであり、32 ビットの OLE DB プロバイダーを使用する予定の場合は、IBMDADB2 DLL を手動で登録する必要があります。この DLL を登録するには、次のコマンドを実行します。

```
c:¥windows¥SysWOW64¥regsvr32 /s c:¥Program_Files¥IBM¥SQLLIB¥bin¥ibmdadb2.d11
```

*Program\_Files* は Program Files ディレクトリーの場所を表します。

ローカル・コンピューターか、ネットワーク上の別のコンピューターにある DB2 資料に DB2 データベース製品からアクセスできるようにする場合は、DB2 インフォメーション・センターをインストールする必要があります。DB2 インフォメーション・センターには、DB2 データベース・システムと DB2 関連製品の資料が収録されています。デフォルトでは、DB2 インフォメーション・センターがローカルにインストール済みでなければ、Web を介して DB2 情報にアクセスできます。

DB2 セットアップ・ウィザードを実行して、IBM Data Studio をインストールすることができます。

#### **DB2 Express® Edition および DB2 Workgroup Server Edition のメモリ限度**

DB2 Express Edition をインストールしている場合、このインスタンスで許可される最大メモリは 4 GB です。

DB2 Workgroup Server Edition をインストールしている場合、このインスタンスで許可される最大メモリは 64 GB です。

インスタンスに割り振られるメモリ量は、INSTANCE\_MEMORY データベース・マネージャー構成パラメーターによって決まります。

#### **バージョン 9.5 または 9.7 からアップグレードする際の重要な注意事項:**

- セルフチューニング・メモリ・マネージャーを使用する場合、ライセンス限度を超えてインスタンス全体のメモリ限度が増やされることはありません。

## **DB2 セットアップ・ウィザードによる DB2 サーバーのインストール (Linux および UNIX)**

このタスクでは、Linux および UNIX オペレーティング・システムで DB2 セットアップ・ウィザードを開始する方法を説明します。DB2 セットアップ・ウィザードを使用して、インストール設定を定義し、ご使用のシステムに DB2 データベース製品をインストールします。

### **始める前に**

DB2 セットアップ・ウィザードを開始する前に、以下の事柄を行います。

- パーティション・データベース環境のセットアップを予定している場合は、「DB2 サーバー機能 インストール」の『パーティション・データベース環境のセットアップ』を参照してください。

- ご使用のシステムがインストール、メモリー、およびディスクの各要件に合うことを確認します。
- サポートされるブラウザがインストールされていることを確認します。
- DB2 データベース・サーバーは、root 権限と非 root 権限のどちらを使用してもインストールできます。非ルート・インストールについては、「DB2 サーバー機能 インストール」の『非ルート・インストールの概要 (Linux および UNIX)』を参照してください。
- DB2 データベース製品イメージが使用可能でなければなりません。DB2 インストール・イメージは、物理的な DB2 データベース製品の DVD を購入するか、またはパスポート・アドバンテージからインストール・イメージをダウンロードすることによって入手することができます。
- 英語版以外の DB2 データベース製品をインストールする場合は、該当する National Language Packages が必要になります。
- DB2 セットアップ・ウィザードは、グラフィック・インストーラーです。ご使用のマシンで DB2 セットアップ・ウィザードを実行するには、グラフィカル・ユーザー・インターフェースを表示できる X windows ソフトウェアが必要です。X windows サーバーが実行中であることを確認します。ディスプレイを正しくエクスポートしたことを確認してください。例えば、export DISPLAY=9.26.163.144:0 のようにします。
- セキュリティー・ソフトウェアを使用している環境の場合、DB2 セットアップ・ウィザードを開始する前に、必要な DB2 ユーザーを手動で作成しなければなりません。

#### 制約事項

- どのユーザー・アカウントでも、DB2 セットアップ・ウィザードの複数のインスタンスを実行することはできません。
- XML フィーチャーは、コード・セット UTF-8 で定義され、データベース・パーティションが 1 個のみであるデータベースでのみ使用できます。
- DB2 セットアップ・ウィザード・フィールドでは英語以外の文字を受け入れません。
- Itanium ベースの HP Integrity Series システム上の HP-UX 11i V2 の場合、DB2 インスタンス所有者のセットアップ・ウィザードで作成されたユーザー、fenced ユーザー、または DAS には DB2 セットアップ・ウィザードで指定されたパスワードを使ってアクセスすることはできません。セットアップ・ウィザードが終了した後、それらのユーザーのパスワードを再設定する必要があります。これは、セットアップ・ウィザードを使ったインスタンスまたは DAS の作成には影響しません。したがって、インスタンスまたは DAS を再作成する必要はありません。

#### 手順

次のようにして、DB2 セットアップ・ウィザードを開始します。

1. 物理的な DB2 データベース製品 DVD を入手している場合は、次のコマンドを入力することによって、DB2 データベース製品 DVD がマウントされているディレクトリーに移動します。

```
cd /dvdrom
```



ここで、`/dvdrom` は、DB2 データベース製品 DVD のマウント・ポイントを表しています。

2. DB2 データベース製品イメージをダウンロードした場合は、製品ファイルを解凍して `untar` しなければなりません。
  - a. 以下のようにして、製品ファイルを解凍します。

```
gzip -d product.tar.gz
```

ここで、`product` はダウンロードした製品の名前です。

- b. 以下のようにして、製品ファイルを `untar` します。

#### Linux オペレーティング・システムの場合

```
tar -xvf product.tar
```

#### AIX、HP-UX、および Solaris オペレーティング・システムの場合

```
gntar -xvf product.tar
```

ここで、`product` はダウンロードした製品の名前です。

- c. 以下のようにディレクトリーを変更します。

```
cd ./product
```

ここで、`product` はダウンロードした製品の名前です。

**注:** National Language Package をダウンロードした場合、同じディレクトリーに `untar` します。それぞれのサブディレクトリー (例えば、`./nlpack`) が同じディレクトリーに作成されるので、インストーラーは、プロンプト画面を表示しなくてもインストール・イメージを自動的に検出できます。

3. データベース製品イメージのあるディレクトリーから `./db2setup` コマンドを入力して、DB2 セットアップ・ウィザードを開始します。
4. 「IBM DB2 セットアップ・ランチパッド」 がオープンします。このウィンドウから、インストールの前提条件およびリリース・ノートを表示することができます。あるいは、インストールに直接進むこともできます。追加された最新のインストール前提条件およびリリース情報を参照することをお勧めします。
5. 「製品のインストール」をクリックすると、「製品のインストール」ウィンドウに、インストールに使用できる製品が表示されます。

「新規インストール」をクリックすることにより、インストールを起動します。DB2 セットアップ・ウィザードのプロンプトに従ってインストールを進めます。

6. DB2 セットアップ・ウィザードを使用する際のサンプル・パネルからインストール・プロセスに進みます。関連リンクを参照してください。

インストールを開始した後、DB2 セットアップ・ウィザードのインストール・パネルに従って、選択を行ってください。残りのステップについて説明しているインストール操作のヘルプを利用できます。インストール操作のヘルプを呼び出すには、「ヘルプ (Help)」をクリックするか、または F1 を押します。「キャンセル」をクリックすれば、いつでもインストールを終了できます。

## タスクの結果

非 root インストールの場合、DB2 データベース製品は必ず `$HOME/sqllib` ディレクトリにインストールされます。ここで、`$HOME` は非 root ユーザーのホーム・ディレクトリを表します。

root インストールの場合には、DB2 データベース製品はデフォルトでは以下のいずれかのディレクトリにインストールされます。

### AIX、HP-UX、および Solaris

`/opt/IBM/db2/V10.1`

### Linux

`/opt/ibm/db2/V10.1`

インストール先のシステムでこのディレクトリが既に使用中の場合、DB2 データベース製品のインストール・パスに `_xx` が追加されます。`_xx` は 01 で始まる数字で、インストール済みの DB2 コピーの数に応じて増加します。

独自の DB2 データベース製品のインストール・パスを指定することもできます。

DB2 インストール・パスには、以下の規則があります。

- 英小文字 (a から z)、英大文字 (A から Z)、および下線文字 ( `_` ) を使用できます。
- 128 文字を超えることはできません。
- スペースは使用できません。
- 英語以外の文字は使用できません。

インストール・ログ・ファイルは、以下で構成されています。

- DB2 セットアップ・ログ・ファイル。このファイルは、エラーを含むすべての DB2 インストール情報をキャプチャーします。
  - root インストールの場合、DB2 セットアップ・ログ・ファイル名は `db2setup.log` です。
  - 非 root インストールの場合、DB2 セットアップ・ログ・ファイル名は `db2setup_username.log` となり、`username` はインストールを実行した非 root ユーザー ID です。
- DB2 エラー・ログ・ファイル。このファイルは、Java によって戻されるエラー出力 (例外やトラップ情報など) をキャプチャーします。
  - root インストールの場合、DB2 エラー・ログ・ファイル名は `db2setup.err` です。
  - 非 root インストールの場合、DB2 エラー・ログ・ファイル名は `db2setup_username.err` となり、`username` はインストールを実行した非 root ユーザー ID です。

デフォルトでは、`/tmp` ディレクトリにこうしたログ・ファイルがあります。これらのログ・ファイルの場所を指定できます。

`db2setup.his` ファイルはなくなりました。代わりに、DB2 インストーラーは DB2 セットアップ・ログ・ファイルのコピーを `DB2_DIR/install/logs/` ディレクトリに保管し、名前を `db2install.history` に変更します。この名前が既存の場合は、

DB2 インストーラーは名前を `db2install.history.xxxx` (`xxxx` はこのマシンにインストールした数に応じて 0000 から 9999 になる) に変更します。

ヒストリー・ファイルのリストはインストール・コピーごとに異なります。インストール・コピーが除去されると、このインストール・パスの下のヒストリー・ファイルもまた除去されます。このコピー・アクションはインストールの終了直前に行われるので、完了前にプログラムが停止したり異常終了したりすると、ヒストリー・ファイルは作成されません。

## 次のタスク

- インストールを検証します。
- 必要なインストール後の作業を実行します。

DB2 セットアップ・ウィザードを実行して、IBM Data Studio をインストールすることができます。

また National Language Packages は、DB2 データベース製品のインストール後に、National Language Packages があるディレクトリーから `./db2setup` コマンドを実行するとインストールできます。

Linux x86 では、ローカル・コンピューターか、ネットワーク上の別のコンピューターにある DB2 資料に DB2 データベース製品からアクセスできるようにする場合は、DB2 インフォメーション・センターをインストールする必要があります。DB2 インフォメーション・センター には、DB2 データベース・システムと DB2 関連製品の資料が収録されています。

### DB2 Express Edition および DB2 Workgroup Server Edition のメモリー限度

DB2 Express Edition をインストールしている場合、このインスタンスで許可される最大メモリーは 4 GB です。

DB2 Workgroup Server Edition をインストールしている場合、このインスタンスで許可される最大メモリーは 64 GB です。

インスタンスに割り振られるメモリー量は、`INSTANCE_MEMORY` データベース・マネージャー構成パラメーターによって決まります。

#### バージョン 9.5 または 9.7 からアップグレードする際の重要な注意事項:

- バージョン 9.5 または 9.7 DB2 データベース製品のメモリー構成が許容限度を超過すると、DB2 データベース製品は現行バージョンへのアップグレード後に開始しない可能性があります。
- セルフチューニング・メモリー・マネージャーを使用する場合、ライセンス限度を超えてインスタンス全体のメモリー限度が増やされることはありません。

## DB2 の応答ファイル・インストールの概要 (Windows)

Windows では、DB2 製品の応答ファイル・インストールを 1 台のマシンでも複数台のマシンでも実行できます。応答ファイル・インストールは、サイレント・インストールまたは無人インストールとも呼ばれます。

## 始める前に

インストールを開始する前に、下記のことを確認してください。

- ご使用のシステムが、DB2 製品をインストールするために必要なメモリー、ハードウェア、およびソフトウェア要件をすべて満たしていること。
- インストールの実行に必要なすべてのユーザー・アカウントを持っていること。
- すべての DB2 プロセスが停止していること。

## 手順

- 1 台のマシンで DB2 製品の応答ファイル・インストールを実行するには、以下のようになります。
  1. 以下のいずれかの方法で、応答ファイルを作成してカスタマイズします。
    - サンプル応答ファイルを変更する。サンプル応答ファイルは、(db2¥Windows¥samples) にあります。
    - DB2 セットアップ・ウィザードを使用して、応答ファイルを生成する。
    - 応答ファイル生成プログラムを使用する。
  2. カスタマイズした応答ファイルを指定して **setup -u** コマンドを実行します。例えば、インストール時に作成した応答ファイルを使用して、次のようになります。

```
setup -u my.rsp
```
- 複数のマシンで DB2 製品の応答ファイル・インストールを実行するには、下記のようにします。
  1. ディレクトリーへの共有アクセスをセットアップします。
  2. サンプル応答ファイルを使用して応答ファイルを作成します。
  3. 応答ファイルを使用して DB2 製品をインストールします。

## DB2 の応答ファイル・インストールの概要 (Linux および UNIX)

ここでは、Linux またはUNIX において応答ファイル・インストールを実行する方法について説明します。初回インストール後も、追加のコンポーネントや製品をインストールするために応答ファイルを使用できます。応答ファイル・インストールは、サイレント・インストールまたは無人インストールとも呼ばれます。

## 始める前に

インストールを開始する前に、下記のことを確認してください。

- ご使用のシステムが、DB2 データベース製品をインストールするために必要なメモリー、ハードウェア、およびソフトウェア要件をすべて満たしていること。
- すべての DB2 プロセスが停止されていること。コンピューター上の既存の DB2 インストール済み環境の上に DB2 データベース製品をインストールする場合は、その既存の DB2 インストール済み環境に関係するすべての DB2 インスタンスおよび DB2 DAS で、すべての DB2 アプリケーション、DB2 データベース・マネージャー、および DB2 プロセスを停止させる必要があります。

## 制約事項

応答ファイルによる方法で DB2 を Linux または UNIX オペレーティング・システムにインストールするときは、以下に示す制限に注意しなければなりません。

- インスタンスまたはグローバル・プロファイルのレジストリー・キーワードを BLANK ("BLANK" という語) に設定すると、現在設定されているキーワードのリストからそのキーワードが削除されてしまいます。
- インストールの前に、ディスク・スペースが十分あることを確認してください。そうしない場合、インストールに失敗するなら、手動でのクリーンアップが必要です。
- 複数インストールを実行する場合、または複数の DVD から DB2 データベース製品をインストールする場合は、DVD ドライブではなくネットワーク・ファイル・システムからインストールすることをお勧めします。ネットワーク・ファイル・システムからインストールすると、インストールの実行にかかる時間がかなり短縮されます。
- 複数のクライアントをインストールする場合は、コード・サーバー上にマウント済みのファイル・システムをセットアップして、パフォーマンスを向上させる必要があります。

## 手順

応答ファイル・インストールを実行するには、以下のようにします。

1. DB2 データベース製品 DVD をマウントするか、またはインストール・イメージが保管されているファイル・システムにアクセスします。
2. サンプル応答ファイルを使用して応答ファイルを作成します。

応答ファイルのファイル・タイプは `.rsp` です。例えば、`ese.rsp` などです。

3. 応答ファイルを使用して DB2 をインストールします。

---

## スタンドアロン・テキスト検索サーバーのインストールおよび構成

### スタンドアロン・サーバーのインストール・スペース所要量

スタンドアロン・テキスト検索サーバーのインストールには、1 GB 以上のディスク・スペースが必要です。

さらに、各コレクションの構成データ用に少量のディスク・スペースと、索引データ用に大量のスペースが必要です。詳しくは、DB2 Text Search のディスクの使用量のトピックを参照してください。

索引データ・ファイルのロケーションは、デフォルトのデータ・ディレクトリーを使用して構成することも、テキスト検索索引の作成時にコレクション・ディレクトリー・パラメーターとして指定することもできます。

### スタンドアロン DB2 Text Search サーバーのインストール

スタンドアロン DB2 Text Search サーバーはサイレント・インストールを行うことができます。サイレント・インストール・オプションでは、応答ファイルから入力値が取得されます。1 回のスタンドアロン・セットアップで 1 つ以上のサーバーをインストールできます。スタンドアロン・テキスト検索サーバーは、ECM Text Search サーバーとも呼ばれます。

## 手順

スタンドアロン・テキスト検索サーバーをインストールするには、以下のようになります。

1. 空のインストール・ディレクトリーを作成します。
  - 例えば、Linux または UNIX システムの場合、以下のディレクトリーを作成します。

```
/opt/ibm/ECMTextSearch
```

- 例えば、Windows システムの場合、以下のディレクトリーを作成します。

```
C:%Program Files%IBM%ECMTextSearch
```

このディレクトリーを `<ECMTS_HOME>` と呼びます。

2. IBM DB2 Accessories Suite for Linux, UNIX, and Windows の Web サイトから、使用するプラットフォーム用の DB2 Accessories Suite をダウンロードします。インストール・ファイルを一時ディレクトリーに解凍します。
3. 必要な権限や許可を持つユーザーとしてログインします。
  - Linux および UNIX のシステムでは、インストール・ディレクトリーに対する読み取り、書き込み、および実行の権限が必要です。
  - Windows システムでは、管理者権限が必要です。
4. ライセンスを確認し、`ecmts_response.txt` ファイルを編集して、設定をカスタマイズします。
5. セットアップ・ファイル `ecmts15_install_<platform>.exe` を使用して、スタンドアロン Text Search サーバーをインストールします。以下のコマンドを発行して、インストールを開始します。

```
./<ecmts_setup_file_name> -i silent -f ecmts_response.txt
```

例えば、Windows システムの場合、以下のコマンドを発行します。

```
ecmts15_install_win32.exe -i silent -f ecmts_response.txt
```

6. インストールが正常に実行されたことを確認します。

`<ECMTS_HOME>` ディレクトリーに作成されたインストール・ログ・ファイルおよびフォルダーを確認します。少なくとも、`bin`、`lib`、`config`、および `resource` のフォルダーが確認できるはずです。

7. `<ECMTS_HOME>` ディレクトリーから開始スクリプトを実行して、サーバーを開始します。
  - Linux および UNIX プラットフォームの場合:

```
bin/startup.sh
```
  - Windows プラットフォームの場合:

```
bin%startup
```
8. テキスト検索サーバーのプロパティを構成してカスタマイズします。詳しくは、スタンドアロン DB2 Text Search サーバーの構成についてのトピックを参照してください。

## スタンドアロン・サーバーを Windows サービスとしてインストールおよび構成する

スタンドアロン・テキスト検索サーバーのプロセスは、Microsoft Windows サービスとしてインストールおよび構成できます。

### このタスクについて

スタンドアロン・サーバー・サービスは、ローカル・システム・アカウントで実行され、開始タイプは自動的に設定されます。サービスの名前を指定し、インストール後にサービスを自動的に開始するかどうかを指定します。

### 手順

スタンドアロン・サーバーを Windows サービスとしてインストールおよび実行するには、以下のようにします。

1. `ecmts_response.txt` 応答ファイルを開き、以下のパラメーターを設定します。

- `IA_INSTALL_AS_WINDOWS_SERVICE`  
このパラメーターの値を `YES` に設定します。
- `IA_WINDOWS_SERVICE_NAME`  
スタンドアロン DB2 Text Search の Windows サービスの固有の名前を指定します。これはオプション・パラメーターです。

このパラメーターの値が指定されていないか、`AUTO` に設定されている場合、デフォルト名の `ECM Text Search Server` が Windows サービスに割り当てられます。サービスが既に存在していて、その名前が指定されていない場合は、例えば、`ECM Text Search Server1` のように名前に数字の接尾部が追加されます。サービスの名前を指定した場合に、同じ名前のサービスが既に存在する場合は、エラーが返されます。

- `IA_START_SERVER`  
インストール後に DB2 Text Search の Windows サービスを自動的に開始するには、`IA_START_SERVER` パラメーターを `YES` に設定するようにします。これはオプション・パラメーターです。デフォルト設定は `YES` です。
2. スタンドアロン・テキスト検索サーバーをインストールします。セットアップおよび応答ファイルを含むディレクトリーから、ご使用のオペレーティング・システムに適切なセットアップ・ファイルを実行します。
3. Microsoft の「サービス」ウィンドウを使用して、テキスト検索の Windows サービスを開始および停止できます。「サービス」ウィンドウにアクセスするには、Windows のコントロール・パネルを開き、「管理ツール」>「サービス」をクリックします。

## スタンドアロン DB2 Text Search サーバーのアンインストール

スタンドアロン DB2 Text Search は、`Uninstall_ECMTTextSearch` コマンドを使用してアンインストールできます。

### 始める前に

アンインストール処理を開始する前に、DB2 Text Search サービスをすべて停止し、スタンドアロン DB2 Text Search サーバーをシャットダウンします。

## 手順

スタンドアロン DB2 Text Search サーバーをアンインストールするには、以下のようになります。

1. *ECMTS\_HOME* ディレクトリーにナビゲートします。
2. 以下のいずれかのプラットフォーム固有のコマンドを発行して、アンインストールを開始します。

- Linux および UNIX オペレーティング・システムの場合:

```
INSTALL_DIR/Uninstall_ECMTTextSearch/Uninstall_ECMTTextSearch -i silent
```

- Windows オペレーティング・システムの場合:

```
ECMTS_HOME\Uninstall_ECMTTextSearch\Uninstall_ECMTTextSearch.exe -i silent
```

アンインストール・プログラムは、*ECMTS\_HOME* ディレクトリーからすべてのデータを削除するわけではありません。例えば、*uninstall.log* ファイルは、アンインストール・プログラムを実行した後も残ります。次のディレクトリーの一部またはすべてが、アンインストール・プログラムによって削除されない可能性があります。手動で削除する必要があります。

- *ECMTS\_HOME*\config
- *ECMTS\_HOME*\license
- *ECMTS\_HOME*\log
- *ECMTS\_HOME*\resource
- *ECMTS\_HOME*\temp
- *ECMTS\_HOME*\Uninstall\_ECMTTextSearch

**ヒント:** 場合によっては、将来使用するために、*ECMTS\_HOME*\config ディレクトリーに保管されているコレクションや構成データをバックアップする必要があります。

## タスクの結果

DB2 Text Search サーバーがアンインストールされ、テキスト検索索引の管理やフルテキスト照会の実行には使用できなくなります。ただし、テキスト索引コレクションや構成データはそのまま残ります。



---

## 第 5 章 DB2 Text Search の構成

DB2 Text Search の構成オプションは、初期構成を行うか、または再構成を行うか、そしてどのプラットフォームを使用するかによって異なります。

### 始める前に

DB2 Text Search を再構成する前に、88 ページの『DB2 Text Search インスタンス・サービスの停止』に概説しているように、テキスト検索インスタンス・サービスを停止します。

パーティション・インスタンスの場合、構成ツールの使用時は調整メンバーまたはインスタンス所有パーティションで行う必要があります。これは、統合テキスト検索サーバーが初期構成されたインスタンス・ホストであり、最も小さい数字が付いたパーティション・サーバー・ホストです。

### 手順

- DB2 Text Search が構成されているかどうかを判別します。

以下のコマンドを発行して構成ツールを実行します。

```
configTool printAll -configPath absolute-path-to-configuration-folder
```

DB2 Text Search が構成されていない場合、**printAll** オプションの出力で、認証トークンは空ストリングです。

- DB2 Text Search の初回の構成を行います。

Linux および UNIX オペレーティング・システム上では、以下のいずれかの方法を使用して、DB2 Text Search を構成します。

- 50 ページの『応答ファイルによる DB2 Text Search のインストールおよび構成』の説明に従って、サイレント・インストールを再実行する。
- 50 ページの『DB2 セットアップ・ウィザードによる DB2 Text Search のインストールおよび構成』の説明に従って、GUI インストールを再実行する。
- 構成ツールを使用する。67 ページの『統合 DB2 Text Search サーバーの初期構成』を参照してください。構成ツールを使用して手動構成を実行する場合は、ほとんどのパラメーターを手動で構成する必要があるのに対し、インストーラーを使用する場合は、構成する必要があるのは 2つのパラメーターだけであることを注意してください。
- 以下のいずれかのコマンドを使用して、インスタンス・タイプおよび操作に応じて DB2 Text Search を構成します。
  - root インストールでは、構成時に DB2 Text Search を選択することによって、GUI で **db2isetup** コマンドを発行して、既存の DB2 インスタンスを構成できます。また、**db2iupdt** コマンドに **-j** オプションを指定して発行して、統合 DB2 Text Search サーバーを構成することもできます。**-j** オプションを指定した **db2icrt** コマンドを使用してインスタンスを作成する場合、デフォルトで DB2 Text Search も構成されることに注意してください。

- 非 root インストールでは、**db2isetup** コマンドを発行してインスタンスを GUI で構成するか、または **db2nrupdt** コマンドか **db2nrupgrade** コマンドに **-j** オプションを指定して発行します。

Windows オペレーティング・システム上では、以下のいずれかの方法を使用して、DB2 Text Search を構成します。

- 50 ページの『応答ファイルによる DB2 Text Search のインストールおよび構成』の説明に従って、サイレント・インストールを再実行する。
- 50 ページの『DB2 セットアップ・ウィザードによる DB2 Text Search のインストールおよび構成』の説明に従って、GUI インストールを再実行する。
- **db2iupdt** コマンドに **-j** オプションを指定して発行する。 **-j** オプションを指定して **db2icrt** コマンドを使用し、インスタンスを作成する場合、デフォルトで DB2 Text Search も構成されることに注意してください。

- Java Developer Kit が IBM 製であるかどうか判別します。

DB2 Text Search は、`db2 get dbm cfg` コマンドの **JDK\_PATH** で指示される場所にある Java Developer Kit を内部で使用します。この Java Developer Kit は IBM 製でなければなりません。Java Developer Kit が IBM 製であるかどうかを検証するには、以下のコマンドを実行します。

```
JDK_PATH/jre/bin/java -version
```

このコマンドは、Java バージョン情報を表示し、Java Developer Kit が IBM 製である場合は、ストリングの一部に **IBM** と表示されます。

- DB2 Text Search を再構成します。

DB2 Text Search を構成した後、GUI インストーラーを使用して再構成することはできません。構成を更新するには、手動で行う必要があります。

Linux および UNIX オペレーティング・システム上では、以下のいずれかの方法を使用して、DB2 Text Search を再構成します。

- 50 ページの『応答ファイルによる DB2 Text Search のインストールおよび構成』の説明に従って、サイレント・インストールを再実行する。
- 構成ツールを使用する。 67 ページの『統合 DB2 Text Search サーバーの初期構成』を参照してください。
- 以下のいずれかのコマンドを使用して、インスタンス・タイプおよび操作に応じて DB2 Text Search を再構成します。
  - root インストールでは、構成する DB2 Text Search インスタンスを選択することによって、GUI で **db2isetup** コマンドを発行して、既存の DB2 インスタンスを構成できます。また、**db2iupdt** コマンドに **-j** オプションを指定して発行して、統合 DB2 Text Search サーバーを構成することもできます。
  - 非 root インストールでは、**db2isetup** コマンドを発行してインスタンスを GUI で構成するか、または **db2nrupdt** コマンドか **db2nrupgrade** コマンドに **-j** オプションを指定して発行します。

Windows オペレーティング・システム上では、以下のいずれかの方法を使用して、DB2 Text Search を再構成します。

- 50 ページの『応答ファイルによる DB2 Text Search のインストールおよび構成』の説明に従って、サイレント・インストールを再実行する。
- 構成ツールを使用する。『統合 DB2 Text Search サーバーの初期構成』を参照してください。
- **db2iupdt**、または **db2iupgrade** コマンドを、以下のように必要に応じて **-j** オプションを指定して実行する。
  - **-j "TEXT\_SEARCH"** は、デフォルトのサービス名および生成されたポート値で DB2 Text Search を構成しようとします。
  - **-j "TEXT\_SEARCH,[servicename]"** は、自動的に生成されるポート番号でサービス名を予約するか、またはサービス名が既にサービス・ファイルで予約済みである場合には、サービス名に割り当てられているのと同じポート番号でサービス名を予約します。
  - **-j "TEXT\_SEARCH,[port number]"** はポートをデフォルトのサービス名で予約します。
  - **-j "TEXT\_SEARCH,[servicename],[port#]"** は指定されたサービス名とポート番号を予約します。

注: Windows オペレーティング・システムでは、DB2 コマンド・ウィンドウ内の **PATH** は *current-default-copy-install-path%db2tss%bin* を指すため、現行の DB2 コピーにないインスタンスを構成するには、まず、そのコピー用の適切な DB2 コマンド・ウィンドウに切り替えてください。

---

## 統合 DB2 Text Search サーバーの初期構成

構成ツールは、DB2 Text Search の初期構成を実行するか、または現行の構成を変更するために使用できるコマンド行ツールです。

### 始める前に

構成設定をカスタマイズするには、ほとんどの場合、DB2 Text Search インスタンス・サービスを停止する必要があります。

### このタスクについて

インストール後の初期構成で最も便利な方法は、DB2 インストーラーを使用することです。構成の更新だけでなく手動による初期構成でも、構成ツールを使用する必要があります。

### 手順

DB2 Text Search サーバーの初期構成を実行するには、以下の手順を使用します。詳しくは、構成ツールのトピックを参照してください。

1. **configTool** コマンドに **configureParams** オプションを指定して実行し、構成パスを設定します。
  - 以下の構成オプションを確認し、必要に応じてデフォルトを変更します。
    - defaultDataDirectory**: テキスト索引コレクションのロケーション。各コレクションは独自のサブディレクトリーに保管されます。

**-logPath:** Text Search サーバーのログ・ファイルおよびトレース・ファイルのロケーション。

**-tempDirPath:** 一時ディレクトリーへのパス。

**-installPath:** Windows では DB2PATH\db2tss、Linux および UNIX では DB2DIR/db2tss ディレクトリーである、DB2 Text Search インストール・ディレクトリーへのパス。ここで DB2DIR は DB2 コピーのロケーションです。

**-startupHeapSize:** テキスト検索サーバーの最大ヒープ・サイズ。

例えば、**defaultDataDirectory** および **installPath** のオプションを構成するには、以下のコマンドを発行します。

```
configTool configureParams -configPath <absolute-path-to-config-folder>
-defaultDataDirectory dataPath -installPath ipath
```

- Windows オペレーティング・システムでは、以下のようにコマンドを指定します。指定する必要があるのは **configPath** のみで、他のすべてのパラメーターにはデフォルトのパスおよび値が割り当てられます。

```
configTool
-configPath absolute-path-to-config-folder
```

2. DB2 Text Search は、認証トークンを使用して、テキスト検索索引管理およびテキスト検索要求を認証します。以下のように **configTool** コマンドに **generateToken** パラメーターを指定して発行し、認証トークンを生成します。

```
configTool generateToken
-configPath absolute-path-to-config-folder
-seed myInstanceName
```

3. 以下のように **configTool** コマンドに **configureHTTPListener** パラメーターを指定して発行、HTTP ポートを指定します。

```
configTool configureHTTPListener
-configPath absolute-path-to-config-folder
-adminHTTPPort port-number
```

**注:** ポートの値は 1024 から 65535 の間でなければなりません。

管理 HTTP ポートは、TCP/IP を使用するテキスト検索処理間の通信を可能にします。root 権限がある場合は、DB2 データベース製品のインストール中またはインスタンス作成中に、サービス名およびポートを指定できます。これらは、サービス・ファイルの更新に使用されます。

4. サービス・ファイルを更新します。

72 ページの『TCP/IP 通信のためのサーバー上のサービス・ファイルの更新』を参照してください。

構成ツールを使用して構成する場合、ツールはサービス・ファイルを更新しません。したがって、サービス・ファイルは手動で更新する必要があります。

**注:** サービス・ファイルは、root ユーザーのみが更新できます。非 root ユーザーは、システム管理者にまず **db2rfe** コマンドを実行してもらう必要があります。

## DB2 Text Search サーバー情報の更新

DB2 Text Search サーバー情報は、Text Search サーバーに接続してテキスト検索索引の管理および検索を実行するためにデータベースで使用します。このため、システムが正しく機能できるようにするには、有効な設定が必要であり、テキスト検索カタログの `SYSIBMTS.TSSERVERS` 管理ビューに設定を定義する必要があります。

### 始める前に

テキスト検索サーバー情報の更新には、指定されたデータベースの `SYSTS_ADM` ロールと `DBADM` 特権が必要です。

### このタスクについて

サーバー情報は、主に接続情報で構成されます。例えば、サーバー・ホスト名、サーバー・トークン値とサーバー・ポート番号、サーバーのロケールなどのサーバーの特性、リッチ・テキスト・サポートに対してテキスト検索のセットアップが有効かどうか、および DB2 インスタンスによって利用される検索サーバーが統合されているか (DB2 インスタンスの一部として DB2 によって構成されているか)、それとも別個のスタンドアロン・インストールであるテキスト検索サーバーかなどの情報が含まれています。

次のシナリオでは、最初に更新が必要です。

- テキスト検索のためにデータベースを有効にしたが、有効化が不完全であるという警告メッセージが表示された場合
- スタンドアロン・テキスト検索サーバーの初期構成
- パーティション・データベース
- DB2 Text Search のアップグレード
- 管理のためにクライアント・マシンからストアード・プロシージャーを使用する
- また、テキスト検索サーバーの接続情報の更新の後

データベースを有効にする際、構成にアクセスするのに必要な権限を持っている場合は、`SYSIBMTS.TSSERVERS` 管理ビューが統合サーバーの初期接続情報を使用して更新されます。関連するテキスト検索サーバーのデータを使用して、`SYSIBMTS.TSSERVERS` のテキスト・サーバーの情報を確認して更新し、`SYSTS_CONFIGURE` プロシージャーを実行して、更新された情報を適用してください。インスタンスに複数のデータベースがある場合は、同じテキスト検索サーバーの情報を使用して各データベースを構成します。

再構成が必要な場合、テキスト検索の管理操作がアクティブでないことを確認し、テキスト検索サーバーをシャットダウンしてから変更を適用します。

テキスト検索のインストールとテキスト検索の DB2 インスタンスの構成に関連するいくつかの特性を更新する必要があります。これには、以下のものが含まれます。

- DB2 インスタンスによって利用される検索サーバーが統合されているか (DB2 インスタンスの一部として DB2 によって構成されているか)、それとも別個のスタンドアロン・インストールであるテキスト検索サーバーかの指示。

- テキスト検索のセットアップがリッチ・テキスト・サポートに関して有効になっているかの指示。

## 手順

DB2 Text Search サーバーの情報を更新するには、以下のようになります。

1. **configTool** コマンドに `printAll` オプションを指定して発行し、ホスト名、トークン、ポート番号など、必要なテキスト検索サーバーのプロパティ値を取得します。詳しくは、**configTool** に関するトピックを参照してください。
2. **SYSIBMTS.TSSERVERS** 管理ビューの項目を確認して、必要な更新を行います。

- ビューが空の場合、**INSERT** ステートメントを使用します。例:

```
INSERT INTO SYSIBMTS.TSSERVERS (HOST, PORT, TOKEN, SERVERSTATUS)
VALUES ('localhost', 55000, '9kfsjg48=', 0);
```

- ビューに既に行が含まれている場合、**UPDATE** ステートメントを使用します。例:

```
UPDATE SYSIBMTS.TSSERVERS SET (HOST, PORT, TOKEN)
= ('tsmach1.ibm.com', 55002, 'k3j4fjk9u=')
```

複数のデータベース・パーティションを使用している場合、または管理操作をクライアントから実行する場合は、必ず、ローカル・ホストの代わりに実際のホスト名または IP アドレスを使用してください。これは、スタンドアロン・テキスト検索サーバーのローカル・インストールだけでなく、統合サーバーにも適用されます。

3. **SYSTS\_CONFIGURE** プロシージャを実行します。詳しくは、**SYSTS\_CONFIGURE** プロシージャに関するトピックを参照してください。
4. **SYSIBMTS.TSSERVERS** 管理ビューの値が、構成ツールによって戻されるものであることを確認します。
5. テキスト検索サービスを開始して、テキスト検索サーバーと通信できることを確認します。

---

## スタンドアロン DB2 Text Search サーバーの構成

スタンドアロン DB2 Text Search サーバーをインストールした後に、構成ツールを使用して、デフォルト・プロパティをいくつかカスタマイズします。関連するシステム・レベル・プロパティとシステムのセキュリティー・プロパティを構成することができます。

プロパティを構成する前に、スタンドアロン DB2 Text Search サーバーがシャットダウンされていることと、テキスト検索サービスが停止していることを確認してください。スタンドアロン・テキスト検索サーバーの構成が終了し、関連した DB2 インスタンス内で使用可能なデータベースの必要な構成更新が完了するまでは、テキスト検索サーバーを再始動しないでください。

テキスト検索サーバーのプロパティは、テキスト検索サーバーが停止している場合でも、構成ツールを使用して表示することができます。

## システム構成

必ず、構成ツールを使用して、少なくとも以下のプロパティを確認して構成するようにしてください。

- **configureHTTPListener**: DB2 Text Search サーバー・ポートとホスト名を構成します。
- **generateToken**: 認証トークンおよび暗号鍵を生成します。
- **defaultDataDirectory**: コレクションのパラメーターを構成します。

**要確認**: *configPath* の値にブランクが含まれている場合は、その値を引用符で囲む必要があります。

詳細および追加のオプション構成については、DB2 Text Search の構成ツールに関するトピックを参照してください。

## セキュリティ構成

DB2 データベース・サーバーからスタンドアロン DB2 Text Search サーバーへの API 要求ごとに、認証トークンによる認証が行われます。初期トークンは、スタンドアロン・テキスト検索サーバーのインストール時に生成されます。

1. 構成ツールを使用して、シード値を明示的に指定し、認証トークンを生成します。トークン・ストリングの最大長は 32 バイトです。
2. DB2 インスタンスで構成ツールを実行して、対応するトークン値を設定します。
3. トークンを含む接続情報を、使用可能なデータベースごとに SYSIBMTS.TSSERVER 管理ビューに保管します。

現在の認証トークンと暗号鍵の値は、DB2 Text Search 構成ツールを使用して表示することができます。ただし、スタンドアロン DB2 Text Search サーバーで使用されるシード値を判別することはできません。 **configTool** ユーティリティーを使用して、トークンを明示的に生成し、DB2 インスタンスのマスター構成を更新して、トークンに構成された値と一致させてください。

テキスト検索サーバーのプロパティを構成するには、以下の適切なプラットフォーム固有のコマンドを入力して構成ツールを実行します。

- Linux および UNIX プラットフォームの場合:  

```
configTool.sh configuration_command -configPath value  
[-locale value] -command_specific_arguments
```
- Windows プラットフォームの場合:  

```
configTool.bat configuration_command -configPath value  
[-locale value] -command_specific_arguments
```

例えば、Linux サーバーの現在の認証トークンを印刷するには、次のコマンドを使用します。

```
configTool.sh printToken -configPath /opt/ibm/ECMTextSearch/config
```

注: Linux および UNIX プラットフォーム上のスタンドアロン DB2 Text Search サーバーの場合は、.sh 接頭部を含む構成ツール・コマンド全体を指定する必要があります。接頭部のないスクリプト名は、統合 DB2 Text Search サーバーでのみサポートされます。

---

## TCP/IP 通信のためのサーバー上のサービス・ファイルの更新

このタスクは、DB2 インスタンスのための TCP/IP 通信の構成 のメインタスクの一部です。

### このタスクについて

TCP/IP サービス・ファイルは、サーバー・アプリケーションがクライアント要求を listen するときのポートを指定します。DBM 構成ファイルの **svcname** フィールドにサービス名を指定した場合、そのサービス名からポート番号/プロトコルへのマッピングによってサービス・ファイルを更新する必要があります。DBM 構成ファイルの **svcname** フィールドにポート番号を指定した場合、サービス・ファイルの更新は不要 です。

services ファイルを更新し、サーバーが着信クライアント要求を listen するポートを指定します。services ファイルのデフォルトの位置は、オペレーティング・システムによって異なります。

#### Linux および UNIX オペレーティング・システム

/etc/services

#### Windows オペレーティング・システム

%SystemRoot%\system32\drivers\etc\services

### 手順

テキスト・エディターを使用して、サービス・ファイルに接続 (Connection) の項目を追加します。例:

```
db2c_db2inst1 3700/tcp # DB2 connection service port
```

各要素の意味は以下のとおりです。

#### db2c\_db2inst1

接続サービス名を表します。

**3700** 接続ポート番号を表します。

**tcp** 使用している通信プロトコルを表します。

---

## DB2 Accessories Suite for DB2 Text Search のインストール

DB2 Accessories Suite は、リッチ・テキストおよび専有フォーマットの文書に対して、DB2 Text Search を使用して索引付けおよび検索を実行できるようにします。新しいインストールを開始することも、既存のインストールの上にインストールを実行することもできます。



## 始める前に

DB2 Accessories Suite を Linux および UNIX 上にインストールするには、DB2 サーバーにシステム管理者としてログオンする必要があります。Windows では、Local Administrator 権限を持つユーザーとしてログオンする必要があります。

DB2 Accessories Suite をダウンロードします。ダウンロード・リンクについては、<https://www.ibm.com/services/forms/preLogin.do?source=swg-dm-db2accsuite> を参照してください。フィーチャーが正しく機能するように、最新バージョンの DB2 Accessories Suite リリースまたはフィックスパックをインストールします。

インストーラー・ファイル、ライセンス・ファイル、およびリリース情報ファイルが同じディレクトリーにあることを確認します。

## 手順

DB2 Accessories Suite をインストールするには、次のようにします。

1. DB2 Text Search インスタンス・サービスを停止します。サービスを停止するには、**db2ts STOP FOR TEXT** コマンドを発行します。
2. DB2 Text Search インストール・ディレクトリーへの書き込み権限を持つ必要な権限があるユーザーとして、DB2 データベース・サーバーにログオンします。例えば、Linux プラットフォームでは、ディレクトリーは<DB2PATH>/db2tss ディレクトリーの下にあり、<DB2PATH> は DB2 データベース・サーバーのインストール・ディレクトリーを表します。
3. 2 つのインストール・モードがあります。1 つはコンソール・インストールで、もう 1 つはサイレント・インストールです。
  - コンソール・インストールを実行するには、以下のようにします。
    - a. Accessories Suite フィルター・インストーラーを実行します。
      - Linux および UNIX プラットフォームのコマンド行からインストーラー **installAccSuiteV10.bin** を実行します。
      - Windows プラットフォームでは 2 つのアプローチがあります。
        - コマンド・ウィンドウからインストーラー **installAccSuiteV10.exe** を実行します。
        - インストーラーのバイナリー・ファイルをダブルクリックします。
      - b. ライセンスに同意した後に、インストール・パスの入力を求めるプロンプトが出されたら、最新の DB2 コピー内の db2tss サブディレクトリーのロケーションを入力します。
      - c. db2tss ディレクトリーは既に存在していなければなりません。このディレクトリーがない場合、DB2 Text Search のインストールおよび構成が正しく行われていません。
      - d. 要約を参照し、インストールについて確認します。
  - サイレント・インストールを実行するには、以下のようにします。
    - a. **LICENSE\_ACCEPTED** パラメーターを **true** として設定し、db2tss ディレクトリーを含む、インストールの正しい絶対パス **USER\_INSTALL\_DIR** を割り当てることによって、応答ファイルを変更します。

- b. Accessories Suite フィルター・インストーラーをサイレント・モデルで実行します。
  - Linux および UNIX プラットフォーム上のコマンド行から  
**installAccSuiteV10.bin -i silent -f installer.properties** コマンドを実行します。
  - Windows プラットフォームのコマンド・ウィンドウから  
**installAccSuiteV10.exe -i silent -f installer.properties** コマンドを実行します。

## タスクの結果

DB2 Accessories Suite のインストールが正常に完了しました。

## 次のタスク

これで、DB2 Text Search のリッチ・テキスト文書サポートを有効化できるようになります。詳しくは、86 ページの『DB2 Text Search でのリッチ・テキスト文書のサポートを有効にする』を参照してください。

---

## DB2 Accessories Suite for DB2 Text Search のアンインストール

**Uninstall\_DB2AS** コマンドを使用して、スタンドアロン DB2<sup>®</sup> Text Search をアンインストールできます。

### 始める前に

Linux および UNIX のプラットフォームで DB2 Accessories Suite をアンインストールするには、DB2 データベース・サーバーにシステム管理者としてログインしている必要があります。Windows プラットフォームでは、Local Administrator 権限を持つユーザーとしてログインしている必要があります。

### 手順

DB2 Accessories Suite をアンインストールするには、以下のようになります。

1. DB2 Text Search インスタンス・サービスを停止します。サービスを停止するには、db2ts“STOP FOR TEXT”を実行します。
2. オペレーティング・システムで必要な特権を持つユーザーとして DB2 データベース・サーバーにログオンします。
3. リッチ・テキスト・フィーチャーで以前有効にしたすべてのテキスト検索インスタンスのリッチ・テキスト文書のサポートを無効にします。詳しくは、リッチ・テキスト文書をサポートするための DB2 Text Search の無効化についてのトピックを参照してください。
4. DB2 Accessories Suite インストーラーをアンインストールします。インストーラーをアンインストールするには、以下のようになります。
  - Linux および UNIX オペレーティング・システムの場合:  

```
<DB2DIR>/db2tss/Uninstall_DB2ASV10/Uninstall_DB2AS.bin
```

ここで、<DB2DIR> は最新の DB2 コピーのロケーションです。
  - Windows オペレーティング・システムの場合:

<DB2PATH>\db2tss\Uninstall\_DB2ASV10\Uninstall\_DB2AS.exe

ここで、<DB2PATH> は最新の DB2 コピーをインストールしたロケーションです。

## タスクの結果

DB2 Accessories Suite のアンインストールが完了しました。



---

## 第 6 章 DB2 Text Search のアップグレード

---

### 管理者または root インストール用の DB2 Text Search のアップグレード

最新の機能を入手するには、DB2 Text Search インスタンスをアップグレードしてください。テキスト検索インスタンスをアップグレードする場合、DB2 サーバー、インスタンス、およびすべてのデータベースをアップグレードすることが必要です。

#### このタスクについて

以下の手順では、DB2 Text Search バージョン 9.5 またはバージョン 9.7 の root インストール環境 (Linux または UNIX プラットフォームの場合) または管理者インストール環境 (Windows プラットフォームの場合) をアップグレードするためのプロセスを説明します。

#### 手順

1. インスタンス所有者または SYSADM 権限を持つユーザーとしてログインします。
2. DB2 データベース・インスタンスおよび DB2 Text Search インスタンス・サービスを停止します。
3. DB2 Text Search 構成ディレクトリーをバックアップします。

- Linux および UNIX オペレーティング・システムの場合、次のロケーションにあります。

```
$INSTHOME/sql1lib/db2tss/config
```

ここで、*INSTHOME* は、インスタンスのホーム・パスを表します。

- Windows システムの場合、次のロケーションにあります。

```
<INSTPROF>\<INSTNAME>\db2tss\config
```

ここで、*<INSTPROF>* はインスタンス・プロファイル・ディレクトリーを表し、*<INSTNAME>* はアップグレードするインスタンスの名前を示します。

4. DB2 Text Search でリッチ・テキスト文書サポートを有効にしている、DB2 バージョン 9.5 フィックスパック 3 以降からアップグレードする場合、リッチ・テキスト文書サポートを無効にします。リッチ・テキスト文書サポートの無効化の方法の詳細については、DB2 Text Search でのリッチ・テキスト文書サポートの無効化に関するトピックを参照してください。
5. DB2 サーバーに root (Linux および UNIX オペレーティング・システムの場合) またはローカル管理者権限を持つユーザー (Windows オペレーティング・システムの場合) としてログインします。複数パーティション・インスタンスをアップグレードする場合、インスタンスのアップグレードはインスタンス所有パーティションから実行しなければなりません。

6. カスタム・インストールを使用して DB2 Enterprise Server Edition for Linux, UNIX, and Windows バージョン 10.1 の新しいコピーをインストールし、DB2 Text Search が選択されていることを確認します。DB2 Text Search は、カスタム・インストールを選択した場合だけ選択可能なオプションのコンポーネントです。Work-With-Existing モードを選択して、以前の DB2 のコピーの上に新しい DB2 バージョン 10.1 コピーをインストールし、アップグレードするコンポーネントとして DB2 Text Search を選択することもできます。このアプローチでインストールした後は、DB2 インスタンスをアップグレードする必要はありません。
7. **configTool upgradeInstance** コマンドを発行して、DB2 インスタンス用に DB2 Text Search サーバーをアップグレードします。

- Linux および UNIX オペレーティング・システムの場合:

```
$DB2DIR/db2tss/bin/configTool upgradeInstance
-installedConfigPath $DB2DIR/cfg/db2tss/config
-configPath $INSTHOME/sql1lib/db2tss/config
```

ここで、*INSTNAME* はインスタンスのホーム・ディレクトリーで、*DB2DIR* は新しくインストールされた DB2 バージョン 10.1 コピーの場所です。

- Windows オペレーティング・システムの場合:

```
<DB2PATH>%db2tss%bin%configTool upgradeInstance
-installedConfigPath "<DB2PATH>%CFG%DB2TSS%CONFIG"
-configPath "<INSTPROFDIR>%<INSTANCENAME>%DB2TSS%CONFIG"
```

ここで、*<DB2PATH>* は新しくインストールされた DB2 バージョン 10.1 コピーの場所で、*<INSTPROFDIR>* はインスタンスのプロファイル・ディレクトリーです。

**注:** Windows システムでは、DB2 インスタンスが DB2 Text Search 用に以前に構成されておらず、アップグレードする DB2 バージョンがバージョン 9.7 フィックスパック 1 以降の場合、このステップはスキップできます。

**configTool upgradeInstance** コマンドは、テキスト検索構成とデータ・ファイルおよびディレクトリーを置換、変更、およびマージします。

#### config ディレクトリー

以下のファイルがまだ存在しない場合は、コマンドにより、*<ECMTS\_HOME>%config* ディレクトリーにそれらのファイルがコピーされます。

- constructors.xml
- ecmts\_logging.properties
- ecmts\_config\_logging.properties
- mimetypes.xml
- monitoredEventsConfig.xml

以下のファイルがコピーされ、既存のファイルがあれば上書きされます。

- build\_info.properties
- constructors.xsd
- ecmts\_config\_logging.properties

以下のファイルはマージされます。新しい設定については値が追加され、既存の設定については値が維持されます。

- config.xml
- jetty.xml

以下のファイルは変更されません。

- authentication.xml
- key.txt
- collections サブディレクトリーのすべてのファイル

### log ディレクトリー

コマンドにより、既存の log ディレクトリーの内容は変更されません。ただし、新しいログ・ファイルが生成されると、その新しいファイルは既存のログ・ファイルに置き換わります。

**configTool upgradeInstance** コマンドでは、統合テキスト検索サーバーのテキスト検索フィルターをアップグレードしません。

8. **db2iupgrade** コマンドを発行することにより、現在の DB2 インスタンスをアップグレードします。

- Linux および UNIX オペレーティング・システムの場合、コマンドは `$DB2DIR/instance` ディレクトリーにあり、`DB2DIR` は新しくインストールした DB2 データベース・サーバーのバージョン 10.1 コピーの場所です。

```
db2iupgrade -j "TEXT_SEARCH [[,service-name] | [,port-number]]" DB2INST
```

- Windows オペレーティング・システムの場合、プロパティー・ファイルは `<DB2PATH>%bin` ディレクトリーにあります。`<DB2PATH>` は新しくインストールされた DB2 バージョン 10.1 コピーの場所です。

```
db2iupgrade DB2INST /j "TEXT_SEARCH [[,service-name] | [,port-number]]"
```

詳しくは、**db2iupgrade** コマンドに関するトピックを参照してください。

**注:** DB2 バージョン 10.1 の新しいコピーを、アップグレード・オプション付きでインストールし、その際に DB2 Text Search をアップグレード対象のフィーチャーとして選択した場合は、このステップはスキップして構いません。

9. 次のスクリプトを実行して、以前のリリースで使用されていた DB2 Text Search の構成可能なプロパティーすべての値をバックアップします。

- Linux および UNIX オペレーティング・システムの場合:

```
$DB2DIR/db2tss/bin/bkuptscfg.sh $INSTNAME
```

ここで、`DB2DIR` は新しくインストールされた DB2 バージョン 10.1 コピーの場所を表し、`INSTNAME` はアップグレードするインスタンスの名前を表します。

- Windows オペレーティング・システムの場合:

```
<DB2PATH>%db2tss%bin%bkuptscfg.bat <INSTANCENAME> <DB2PATH>
```

ここで、`<DB2PATH>` は新しくインストールされた DB2 バージョン 10.1 コピーの場所を表し、`<INSTANCENAME>` はアップグレードするインスタンスの名前を表します。

バックアップされる構成可能なプロパティは、1 つのプロパティ・ファイルにリダイレクトされます。

- Linux および UNIX オペレーティング・システムの場合、プロパティ・ファイルは `$INSTHOME/sql1lib/db2tss/config/db2tssrvupg.cfg` ディレクトリにあり、`INSTHOME` はインスタンスのホーム・ディレクトリを表します。
- Windows オペレーティング・システムの場合、プロパティ・ファイルは `<INSTPROFDIR>%<INSTANCENAME>%db2tss%config%db2tssrvupg.cfg` ディレクトリにあります。ここで `<INSTPROFDIR>` は、インスタンス・プロファイルのディレクトリを表し、`db2set DB2INSTPROF` コマンドを発行することによって取得できます。また、`<INSTANCENAME>` はアップグレードするインスタンスの名前を表します。

注: DB2 インスタンスが以前の DB2 リリースのコピーで DB2 Text Search 用に構成されていない場合、このステップはスキップできます。

10. `DB2INSTANCE` 環境変数を現在アップグレードしているインスタンスに設定します。
11. `db2 upgrade database` コマンドを発行することにより、データベースをアップグレードします。
12. アップグレードした各データベースで、手順 7 でバックアップしたプロパティ値を比較することによって、テキスト検索 `SYSIBMTS.TSSERVERS` カタログ表のテキスト検索サーバーのプロパティ情報が適切であるか確認します。カタログ表のトークンまたはポート番号の値が空、または適切でない場合、テキスト・サーバーの情報を手動で更新する必要があります。更新方法の詳細については、DB2 Text Search サーバーの情報の更新に関するトピックを参照してください。
13. DB2 Text Search の構成可能なプロパティすべての値を確認します。バックアップした値と比較して、正しい値になっていることを確認してください。次のコマンドを発行して、構成値をチェックします。

```
configTool printAll -configPath <configuration-directory>
```
14. DB2 Text Search でリッチ・テキスト文書サポートを無効にした場合は、DB2 V10 Accessories Suite をインストールする必要があります。詳しくは、DB2 Accessories Suite のインストールに関するトピックを参照してください。
15. 次に、リッチ・テキスト文書サポートを有効にします。詳しくは、DB2 Text Search でリッチ・テキストおよび専有フォーマットのサポートを有効にすることに関するトピックを参照してください。
16. DB2 Text Search インスタンス・サービスを始動することにより、アップグレードが成功したことを確認します。リッチ・テキスト文書サポートを無効にした場合、テキスト検索照会を発行しアップグレード前の結果と比較することにより、リッチ・テキスト文書サポートが有効になっていることを確認します。



## 非 root インストールの DB2 Text Search のアップグレード (Linux および UNIX)

最新の機能を入手するには、DB2 Text Search インスタンスをアップグレードしてください。テキスト検索インスタンスをアップグレードする場合、DB2 サーバー、インスタンス、およびすべてのデータベースをアップグレードすることが必要です。

### このタスクについて

以下の手順では、Linux または UNIX プラットフォーム上で DB2 Text Search バージョン 9.5 またはバージョン 9.7 の非 root インストール環境をアップグレードするためのプロセスを説明します。

### 手順

DB2 Text Search の非 root インストールをアップグレードするには、次のようになります。

1. インスタンス所有者または SYSADM 権限を持つユーザーとしてログインします。次に DB2 インスタンスおよび DB2 Text Search インスタンス・サービスを停止します。
2. 古い DB2 コピーを *<backup-dir>* ディレクトリーにバックアップします。
3. DB2 Text Search でリッチ・テキスト文書サポートを有効にしている、DB2 バージョン 9.5 フィックスパック 3 以降からアップグレードする場合、リッチ・テキスト文書サポートを無効にします。リッチ・テキスト文書サポートの無効化の方法の詳細については、DB2 Text Search でのリッチ・テキスト文書サポートの無効化に関するトピックを参照してください。
4. 非 root ユーザーとして DB2 サーバーにログオンします。データベース・インスタンスのタイプが、非 root インストール環境としてアップグレード可能かどうかを確認します。
5. アップグレードする DB2 Text Search コンポーネントを選択したアップグレード・オプションを指定して、DB2 V10.1 の新しいコピーをインストールします。**-f nobackup** パラメーターを指定して、DB2 データベース製品のインストールが失敗した場合、フィーチャー・ツリーから DB2 Text Search コンポーネントを選択して、DB2 データベース製品を手動でインストールし、次のコマンドを発行して非 root インスタンスをアップグレードする必要があります。

```
db2nrupgrade -b <backup-dir> -j "TEXT_SEARCH"
```

ここで、*<backup-dir>* は古い DB2 バージョンの構成ファイルが保管されているディレクトリーを指定します。**db2nrupgrade** コマンドの詳細については、**db2nrupgrade** に関するトピックを参照してください。

6. 次のスクリプトを実行して、データベースのアップグレードの前に以前のリリースで使用していた DB2 Text Search のすべての構成可能なプロパティの値をバックアップします。

```
$INSTHOME/sql1lib/db2tss/bin/bkuptscfg.sh
```

バックアップされた構成可能なプロパティは、*\$INSTHOME/sql1lib/db2tss/config/db2tssrvupg.cfg* プロパティ・ファイルにリダイレクトされます。

7. **UPGRADE DATABASE** コマンドを発行することにより、既存のデータベースをアップグレードします。
8. アップグレードした各データベースで、手順 6 でバックアップされたプロパティ値を比較することによって、テキスト検索カタログ表の **SYSIBMTS.SYSTSSERVERS** カタログ表のテキスト検索サーバーのプロパティ情報が適切であるか確認します。カタログ表のトークンまたはポート番号の値が空、または適切でない場合、テキスト・サーバーの情報を手動で更新する必要があります。更新方法については、DB2 Text Search サーバーの情報の更新に関するトピックを参照してください。
9. root ベースのフィーチャーを有効にします。 root アクセス権を持つシステム管理者に **db2rfe** コマンドを発行してもらうことを依頼できるかもしれません。
10. **configTool upgradeInstance** コマンドを発行することにより、インスタンスの DB2 Text Search サーバーをアップグレードします。
  - Linux および UNIX オペレーティング・システムの場合:
 

```
configTool upgradeInstance
  -installedConfigPath $DB2DIR/cfg/db2tss/config
  -configPath $INSTHOME/sql/lib/db2tss/config
```

ここで、*INSTHOME* はインスタンス・ホーム・ディレクトリーで、*DB2DIR* は DB2 バージョン 10.1 コピーのロケーションです。
11. DB2 Text Search の構成可能なプロパティすべての値を確認します。バックアップした値と比較して、正しい値になっていることを確認してください。次のコマンドを発行して、構成値をチェックします。
 

```
configTool printAll -configPath configuration-directory
```
12. DB2 Text Search でリッチ・テキスト文書サポートを無効にした場合は、DB2 V10 Accessories Suite をインストールする必要があります。詳しくは、DB2 Accessories Suite のインストールに関するトピックを参照してください。
13. 次に、リッチ・テキスト文書サポートを有効にします。詳しくは、DB2 Text Search でリッチ・テキストおよび専有フォーマットのサポートを有効にすることに関するトピックを参照してください。
14. DB2 Text Search インスタンス・サービスを始動することにより、アップグレードが成功したことを確認します。リッチ・テキスト文書サポートを無効にした場合、テキスト検索照会を発行しアップグレード前の結果と比較することにより、リッチ・テキスト文書サポートが有効になっていることを確認します。

---

## DB2 Text Search を使用しない複数パーティション・インスタンスのアップグレード

最新の機能を入手するには、DB2 Text Search インスタンスをアップグレードしてください。テキスト検索インスタンスをアップグレードする場合、DB2 サーバー、インスタンス、およびすべてのデータベースをアップグレードすることが必要です。

## このタスクについて

DB2 バージョン 10.1 からは、テキスト検索で、パーティション・データベース環境での索引がサポートされています。以下の手順では、root インストール環境の DB2 バージョン 9.5 またはバージョン 9.7 の複数パーティション・インスタンスをアップグレードする処理について説明します。DB2 Text Search はインスタンス上にインストールしないでください。

### 手順

1. インスタンス所有者または SYSADM 権限を持つユーザーとしてログインします。
2. アップグレード後の DB2 Text Search バージョンの新しいコピーをインストールし、カスタム・インストールを実行します。DB2 Text Search は、カスタム・インストールを選択した場合だけ選択可能なオプションのコンポーネントです。
3. 次のように **db2iupgrade** コマンドを発行することにより、インスタンスをアップグレードします。  

```
db2iupgrade /j "text_search [[,service-name]][[,port-number]]"
```
4. **DB2 UPGRADE DATABASE** コマンドを発行することにより、既存のデータベースをアップグレードします。
5. アップグレードされた各データベースについて、テキスト・サーバー情報を手動で更新します。詳しくは、DB2 Text Search サーバー情報の更新に関するトピックを参照してください。

---

## スタンドアロン DB2 Text Search サーバーのアップグレード

スタンドアロン DB2 Text Search サーバーを既にインストールしてある場合、サポートされる最新のフィーチャーおよび機能を手入手するために、既存のインストール済み環境に修正をインストールする必要があります。

### 始める前に

修正をインストールする前に、添付されているすべてのリリース・ノートをお読みになり、当てはまる前提条件またはマイグレーション手順を判別してください。

### 手順

1. スタンドアロン DB2 Text Search サーバーの現在のインストール済み環境の `<ECMTS_HOME>` ディレクトリーにナビゲートします。
2. シャットダウン・スクリプトを実行して、テキスト検索サーバーを停止します。
  - Linux および UNIX プラットフォームの場合:  

```
<ECMTS_HOME>/bin/shutdown.sh
```
  - Windows プラットフォームの場合:  

```
<ECMTS_HOME>%bin%shutdown
```
3. 修正インストールの実行可能ファイルを `<ECMTS_HOME>` ディレクトリーにコピーします。インストール済み環境の `ecmts_response.txt` ファイルがこのディレクトリー内に存在するようにします。

4. 以下のいずれかのコマンドを入力して、修正のインストールを開始します。応答ファイル内の情報は変更しないでください。

```
<ecmts_setup_file_name> -i silent -f ecmts_response.txt
```

**注:** 更新をインストールしても、認証トークンは変更されません。

---

## 第 7 章 テキスト検索索引の構成と管理

---

### DB2 Text Search 用のコマンド行ツール

DB2 Text Search には、使用を容易にするための 5 つのコマンド行ツールが含まれています。

#### 構成ツール

DB2 Text Search の初期構成とその後の構成との両方を実行します。

#### 管理ツール

DB2 Text Search サーバーに関連したさまざまな管理用タスクを実行します。

#### 同義語ツール

テキスト検索索引に同義語ディクショナリーを追加し、テキスト検索索引から同義語ディクショナリーを除去します。

#### ストップワード・ツール

頻繁に出現する用語 (ストップワードと呼ばれる) を、テキスト検索照会から除去します。

#### ログ・フォーマッター・ツール

システム・メッセージおよびトレース・メッセージを表示および保管します。

---

### テキスト検索コマンドの実行

**db2ts** コマンド・シェルを実行することによって、または DB2 Text Search のスタート・プロシージャである管理 SQL ルーチンの 1 つを呼び出すことによって、コマンドを実行できます。

#### このタスクについて

**db2ts** コマンド・シェルを使用するには、コマンド・ストリングをパラメーターとして渡します。**db2ts** コマンド・シェルは DB2 コマンド・シェルと似ており、リモート・データベースが使用されているときはコマンドに接続情報を含める必要があります。ただし、DB2 コマンド・シェルと異なるのは、**db2ts** にはセッションがありません。代わりに、**db2ts** では各コマンドが別個の単位になっているので、それぞれコマンドごとに接続を確立する必要があります。**DB2DBDFT** 環境変数で指定されたデフォルト・データベースでローカルにコマンドを実行する場合は、データベース接続を指定する必要はありません。オペレーティング・システム・レベルで **DB2DBDFT** 環境変数を設定してください。**db2set** コマンドでも設定を行う場合は、必ず同じ値を使用するようにしてください。

管理 SQL ルーチンを使用すると、DB2 Text Search をインストールしていない DB2 クライアントから管理呼び出しを実行することができます。コマンド・ストリ

ングのパラメーターとして汎用 SYSTS\_ADMIN\_CMD 管理 SQL ルーチン呼び出すこともできますし、そのコマンドの特定の管理 SQL ルーチン呼び出すこともできます。

**注:** `db2ts` コマンドで生成されたエラー・メッセージはクライアントのロケールで書き込まれますが、管理ルーチンで生成されたメッセージは `message-locale` 引数で指定されたロケール、またはロケールを指定していない場合は `en_US` ロケールで書き込まれます。

`START FOR TEXT` や `STOP FOR TEXT` などの一部のコマンドは特定のデータベースに関連付けられていないため、`db2ts` コマンド・シェルを使用するだけで実行できます。

---

## リッチ・テキストおよび専有フォーマットのサポート

### DB2 Text Search でのリッチ・テキスト文書のサポートを有効にする

リッチ・テキスト・サポートは、適切に構成された DB2 Text Search サーバー上で有効にすることができます。

#### 始める前に

DB2 Text Search サーバーでのリッチ・テキスト文書のサポートを有効にするには、インスタンス所有者として、`richtextTool` ユーティリティーを `enable` オプション付きで実行する必要があります。

リッチ・テキスト文書のサポートを有効にする前に、各 DB2 Text Search サーバーでリッチ・テキスト文書のサポート用の準備を行う必要があります。詳しくは、72 ページの『DB2 Accessories Suite for DB2 Text Search のインストール』を参照してください。

#### 制約事項

`richtextTool enable` を実行するには、インスタンス所有者としてログオンする必要があります。

#### 手順

1. インスタンス所有者としてログオンします。
2. DB2 Text Search インスタンス・サービスを停止します。サービスを停止するには、`db2ts STOP FOR TEXT` を実行します。
3. DB2 コマンド・ウィンドウから `richtextTool` ユーティリティーを実行して、サポートを有効にします。

- Linux および UNIX オペレーティング・システムの場合:

```
$INSTHOME/sqllib/db2tss/bin/richtextTool enable DB2DIR
```

ここで、`INSTHOME` はインスタンスのホーム・ディレクトリーで、`DB2DIR` は最新の DB2 コピーのロケーションです。

- Windows オペレーティング・システムの場合:

```
DB2PATH\db2tss\bin\richTextTool.bat enable DB2PATH
```

ここで、*DB2PATH* は最新の DB2 コピーをインストールしたロケーションです。

4. DB2 Text Search インスタンス・サービスを開始します。サービスを開始するには、**db2ts START FOR TEXT** を実行します。

## タスクの結果

DB2 Text Search サーバーでのリッチ・テキスト・サポートを有効にしました。

## リッチ・テキストおよび専有フォーマットのサポートの無効化

リッチ・テキストおよび専有フォーマットのサポートは、統合 DB2 Text Search サーバー上でいつでも無効にすることができます。

### 始める前に

DB2 Text Search サーバーでのリッチ・テキスト文書のサポートを無効にするには、インスタンス所有者として、**richTextTool** ユーティリティーを **disable** オプション付きで実行する必要があります。

#### 制約事項

**richTextTool disable** コマンドを実行するには、インスタンス所有者としてログインする必要があります。

### 手順

1. インスタンス所有者としてログオンします。
2. DB2 Text Search インスタンス・サービスを停止します。サービスを停止するには、**db2ts "STOP FOR TEXT"** を実行します。このコマンドについては、88 ページの『DB2 Text Search インスタンス・サービスの停止』を参照してください。
3. DB2 コマンド・ウィンドウから **richTextTool** ユーティリティーを実行して、サポートを使用不可にします。
  - Linux および UNIX オペレーティング・システムの場合:  

```
$INSTHOME/sql1lib/db2tss/bin/richTextTool disable DB2-install-directory
```

ここで、*INSTHOME* はインスタンスのホーム・ディレクトリーです。
  - Windows オペレーティング・システムの場合:  

```
DB2PATH\db2tss\bin\richTextTool.bat disable DB2-install-directory
```

ここで、*DB2PATH* は DB2 データベース・サーバーのコピーをインストールした場所です。
4. DB2 Text Search インスタンス・サービスを開始します。サービスを開始するには、**db2ts "START FOR TEXT"** を実行します。このコマンドについては、88 ページの『DB2 Text Search インスタンス・サービスの開始』を参照してください。

## タスクの結果

DB2 Text Search サーバーでのリッチ・テキスト・サポートを無効にしました。

---

## DB2 Text Search インスタンス・サービスの開始

テキスト索引を作成および検索できるようにするには、まず DB2 Text Search インスタンス・サービスを開始する必要があります。

### このタスクについて

統合 DB2 Text Search インスタンス・サービスを開始するには、以下のコマンドを入力します。

```
db2ts "START FOR TEXT"
```

スタンドアロン・テキスト検索サーバーを開始するには、<ECMTS\_HOME> ディレクトリーから開始スクリプトを実行します。

- Windows の場合:

```
<ECMTS_HOME>%bin%startup
```

- Linux および UNIX の場合:

```
<ECMTS_HOME>/bin/startup.sh
```

次のコマンドを使用して、Text Search サーバーの状況を確認できます。

```
db2ts "START FOR TEXT status"
```

---

## DB2 Text Search インスタンス・サービスの停止

DB2 Text Search インスタンス・サービスを停止すると、テキスト検索サーバーは、現在アクティブなコマンドをすべて終了します。

### このタスクについて

アクティブなコマンドは、次のように終了します。

- テキスト検索索引のコレクションの作成を完了します。これは、複数パーティションのセットアップの場合、テキスト検索照会が複数のコレクションにパーティション化されるため、CREATE INDEX FOR TEXT 操作が失敗する可能性があることを意味します。
- ドロップ・コレクションが既に開始されていて、除去したファイルを元に戻せない場合は、ドロップを完了します。それ以外の場合はコマンドをロールバックします。
- 更新は、キュー内の現在の文書进行处理しますが、その他の文書は受け入れません。初期更新は、試行済みとマーク付けされて再開されます。インクリメンタル更新は、ステージング表内のすべての項目の処理を繰り返します。

テキスト検索サーバーが停止処理を実行する間、新しいコマンドは受け入れられません。

### 手順

DB2 Text Search サーバーを停止するには、次のようにします。



- 統合 DB2 Text Search インスタンス・サービスでは、次のコマンドを入力します。

```
db2ts "STOP FOR TEXT"
```

- スタンドアロン・テキスト検索サーバーでは、`<ECMTS_HOME>` ディレクトリーからシャットダウン・スクリプトを実行します。ここで、`<ECMTS_HOME>` は、スタンドアロン・テキスト検索サーバーのインストール・ディレクトリーを表します。

- Windows の場合:

```
<ECMTS_HOME>%bin%shutdown
```

- Linux および UNIX の場合:

```
<ECMTS_HOME>/bin/shutdown.sh
```

---

## データベースで DB2 Text Search を使用可能にする

検索対象にするテキストの列が含まれている各データベースを使用可能にする必要があります。`db2ts ENABLE DATABASE FOR TEXT` コマンドまたは `SYSPROC.SYSTS_ENABLE` ストアド・プロシージャを使用して、データベースで DB2 Text Search を使用できるようにします。

### 始める前に

ステートメントの許可 ID が `SYSTS_ADM` ロールおよび `DBADM` 権限を保持している必要があります。

### このタスクについて

データベースを使用可能にする際、以下のビューを使用して、データベース内のテキスト検索索引やそのプロパティーに関する情報を確認することができます。

#### **SYSIBMTS.TSDEFAULTS**

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。

#### **SYSIBMTS.TSLOCKS**

データベース・レベルおよび索引レベルで設定されているコマンド・ロックに関する情報を示します。

#### **SYSIBMTS.TSINDEXES**

すべてのテキスト検索索引とその設定を示します。

#### **SYSIBMTS.TSCONFIGURATION**

索引の構成パラメーターを示します。

#### **SYSIBMTS.TSCOLLECTIONNAMES**

各索引のコレクション名を示します。

#### **SYSIBMTS.TSSERVERS**

Text Search サーバーの接続情報を表示します。

データベースでテキスト検索を使用可能にした後は、それを明示的に使用不可にしない限り、テキスト検索は使用可能なままになります。

データベースを DB2 Text Search で使用できるように準備するには、以下のいずれかの方法を使用します。

- 以下のコマンドを入力します。

```
db2ts "ENABLE DATABASE FOR TEXT CONNECT TO databaseName"
```

使用可能にする操作によって、SYSIBMTS.TSSERVERS 管理ビューへのテキスト検索サーバーの接続情報の設定が試行されます。ただし、情報は不完全または不十分である可能性があります。コマンドが正常に完了した場合も、使用可能化が不完全で警告を表示して完了した場合も、コマンドの完了後には SYSIBMTS.TSSERVERS ビューの値を確認し、必要に応じて更新を行ってください。

このステップは、各データベースで一度だけ実行する必要があります。インスタンス・サービスを停止および再始動するたびにデータベースを使用可能にする必要はありません。

例えば、SAMPLE という名前のデータベースを使用可能にするには、次のコマンドを入力します。

```
db2ts "ENABLE DATABASE FOR TEXT CONNECT TO SAMPLE"
```

- 次のように、管理 SQL ルーチンの 1 つを呼び出します。
  - CALL SYSPROC.SYSTS\_ADMIN\_CMD  
( 'ENABLE DATABASE FOR TEXT', 'en\_US', ?)
  - CALL SYSPROC.SYSTS\_ENABLE('en\_US', ?)

---

## DB2 Text Search のデータベースを使用不可にする

テキスト検索を行う予定のないデータベースは、使用不可にします。

### このタスクについて

データベースをテキスト検索で使用不可にすると、カタログ表および管理ビューが SYSIBMTS スキーマからドロップされます。

### 手順

データベースをテキスト検索で使用不可にするには、以下のいずれかの方法に従ってください。

1. **DROP INDEX** コマンドを使用して、データベース内で定義されたテキスト検索索引をドロップします。
2. データベースをテキスト検索で使用不可にするには、以下のいずれかの方法に従ってください。
  - 以下のように、**DISABLE DATABASE FOR TEXT** コマンドを発行します。

```
db2ts "DISABLE DATABASE FOR TEXT CONNECT TO databaseName"
```
  - 次のようにして、SYSPROC.SYSTS\_DISABLE プロシージャを呼び出します。
  - CALL SYSPROC.SYSTS\_DISABLE('en\_US', ?)

**注:** テキスト検索索引は、**FORCE** オプションを使用してドロップすることもできます。しかし、データベースを使用不可にした後に、いくらかのデータ、特にテ

キスト検索コレクションが残る可能性があります。これが起こるのは、**FORCE** オプションでは、DB2 Text Search サーバーにアクセスできない場合でも、テキスト検索索引をドロップできるからです。このように残ったコレクションは、CLEANUP 操作で明示的に除去する必要があります。

---

## 孤立した DB2 Text Search コレクションの削除

**db2ts CLEANUP FOR TEXT** コマンドを使用して孤立したコレクションを削除することも、管理ツールを使用して次の処理を実行し、孤立したコレクションを特定して削除することもできます。

### このタスクについて

テキスト検索索引は、非パーティション・データベースまたは単一パーティション・データベースの場合は単一コレクションに関連付けられ、*n* 個の関連データ・パーティションを持つ複数パーティション・データベースの場合は *n* 個のコレクションに関連付けられます。**db2ts** のコマンドおよびプロシーチャーはテキスト検索索引で機能しますが、テキスト検索ツールはテキスト検索コレクションで機能します。テキスト検索索引が存在しなくなったにもかかわらず、対応するテキスト検索コレクションが存在している場合、それは孤立したコレクションと呼ばれます。

以下のシナリオでコレクションが孤立します。

- テキスト索引を含むデータベースのドロップ
- **DISABLE** または **DROP** の索引操作での **FORCE** オプションの使用

これらの操作は、Text Search サーバーが到達不能であっても成功します。

失敗のシナリオによっては、コレクションが孤立したり、無効な状況になったりする場合があります。例えば、ディスクの破損により、テキスト索引メタデータに不整合が生じることがあります。

孤立したコレクションが存在するかどうかを判別するには、次のようにします。

1. 管理ツールでテキスト検索コレクションをすべて報告させます。次のコマンドを発行します。

```
adminTool status -configPath <absolute-path-to-configuration-folder>
```

2. **SYSIBMTS.TSCOLLECTIONNAMES** 管理ビューを照会し、現行データベース上のテキスト検索索引をすべて報告させます。

```
SELECT collectionname FROM SYSIBMTS.TSCOLLECTIONNAMES
```

DB2 Text Search で使用可能にしているすべてのデータベースでこの照会を実行し、その結果を結合して 1 つのリストにします。

管理ツールではすべてのテキスト検索コレクションがリストされますが、**SYSIBMTS.TSCOLLECTIONNAMES** ビューで照会すると、現行データベース上のテキスト検索索引だけがリストされます。

3. 管理ツールから返されたリストと **SELECT** ステートメントから返されたリストを比較します。管理ツールから返されたテキスト検索コレクションのリストにあって **SELECT** ステートメントから返された索引のリストにない項目があれば、

それは孤立したコレクションです。ただし、DB2 Text Search サーバーの開始時に作成されたデフォルト・コレクションだけは例外で、この規則には該当しません。

孤立したテキスト検索コレクションを以下のコマンドで除去します。

```
adminTool delete -configPath <absolute-path-to-configuration-folder>
-collectionName collection-name
```

**重要: adminTool delete** コマンドによって実行されるアクションは、リカバリーできず、索引のドロップや索引の不整合のレンダリングと同等です。

## 例

現在、UNIX システム上で稼働している DBCP1208 というデータベースで DB2 Text Search を使用可能にしています。孤立したテキスト検索コレクションが存在するかどうかを判別するには、管理ツールと SELECT ステートメントを使用します。

```
adminTool.sh status -configPath $HOME/sqlllib/db2tss/config
```

CollectionName	IndexSize	NumOfDocuments	
Default	13,159B	0	
tigertail_DBCP1208_TS542717_0000		13,159B	11
tigertail_DBCP1208_TS012817_0000		13,159B	17
tigertail_DBCP1208_TS082817_0000		13,159B	16
tigertail_DBCP1208_TS152817_0000		13,159B	18
tigertail_DBCP1208_TS212817_0000		13,159B	16
tigertail_DBCP1208_TS302817_0000		13,159B	17
tigertail_DBCP1208_TS392817_0000		13,159B	10
tigertail_DBCP1208_TS462817_0000		13,159B	10
tigertail_DBCP1208_TS542817_0000		13,159B	12
tigertail_DBCP1208_TS022917_0000		13,159B	10
tigertail_DBCP1208_TS112917_0000		13,159B	16
tigertail_DBCP1208_TS192917_0000		13,159B	11
tigertail_DBCP1208_TS262917_0000		13,159B	12
tigertail_DBCP1208_TS867530_0000		13,159B	16

```
db2 select collectionname from sysibmts.tscollectionnames
```

```
COLLECTIONNAME
```

```
-----
tigertail_DBCP1208_TS542717_0000
tigertail_DBCP1208_TS012817_0000
tigertail_DBCP1208_TS082817_0000
tigertail_DBCP1208_TS152817_0000
tigertail_DBCP1208_TS212817_0000
tigertail_DBCP1208_TS302817_0000
tigertail_DBCP1208_TS392817_0000
tigertail_DBCP1208_TS462817_0000
tigertail_DBCP1208_TS542817_0000
tigertail_DBCP1208_TS022917_0000
tigertail_DBCP1208_TS112917_0000
tigertail_DBCP1208_TS192917_0000
tigertail_DBCP1208_TS262917_0000
```

```
13 record(s) selected.
```

2 つの出力を比較すると、テキスト検索コレクション `tigertail_DBCP1208_TS867530_0000` には対応するテキスト検索索引がないことがわかります。管理ツールを使用して、孤立したそのコレクションを削除します。

```
adminTool.sh delete -configPath $HOME/sqlllib/db2tss/config
-collectionName tigertail_DBCP1208_TS867530_0000
```

## DB2 Text Search 用の同義語ディクショナリー

同義語ディクショナリーには、相互に同義となるワードが含まれています。同義語ディクショナリーを使用して、テキスト検索索引内で照会用語の同義語を検索することにより、検索照会の結果を改善できます。

同義語ディクショナリーを使用して、頭字語や技術的な専門用語など、組織の特定のワードを検索できます。

デフォルトでは、同義語ディクショナリーは検索に使用されません。同義語ディクショナリーを使用するには、それを特定のテキスト検索索引に明示的に追加する必要があります。テキスト検索索引を少なくとも 1 回更新してからであれば、同義語ディクショナリーを追加できません。同義語ディクショナリーを追加した後に、任意の頻度でそれを変更できます。

以下の例に示されているように、同義語ディクショナリーは XML ファイル内に定義される同義語グループで構成されます。

```
<?xml version="1.0" encoding="UTF-8"?>
<synonymgroups version="1.0">
  <synonymgroup>
    <synonym>ball</synonym>
    <synonym>globe</synonym>
    <synonym>sphere</synonym>
    <synonym>orb</synonym>
  </synonymgroup>
  <synonymgroup>
    <synonym>worldwide patent tracking system</synonym>
    <synonym>wpts</synonym>
  </synonymgroup>
</synonymgroups>
```

## DB2 Text Search の同義語ディクショナリーの追加

Synonym Tool を使用すると、同義語ディクショナリーをテキスト検索索引に簡単に追加することができます。

### 始める前に

- 同義語ディクショナリーをテキスト検索索引に追加する前に、DB2 Text Search インスタンス・サービスをアクティブにする必要があります。
- テキスト検索索引を少なくとも 1 回更新している必要があります。
- 同義語グループを指定する同義語 XML ファイルも必要です。

### 手順

同義語ディクショナリーを追加するには、以下のようにします。

1. XML ファイルを DB2 Text Search サーバー上の任意のディレクトリーにコピーします。
2. 同義語ディクショナリーの追加先のテキスト検索索引に関連付けられた、テキスト検索コレクションの名前を判別します。以下のように、管理ツールを使用してすべてのテキスト検索コレクションのレポートを参照することができます。

```
adminTool status -configPath absolute-path-to-config-folder
```

3. Synonym Tool を使用して同義語ディクショナリーを特定のテキスト検索索引へ追加します。同義語は、**append** または **replace** モードで追加できます。つまり、同義語を追加するか、またはそのテキスト検索索引に定義されている既存の同義語と置き換えるかのいずれかです。

```
synonymTool importSynonym -synonymFile absolute-path-to-syn-file
  -collectionName collection-name -replace true or false
  -configPath absolute-path-to-config-folder
```

注: XML 形式が無効であるか、または XML ファイルが空である場合、エラーが戻されます。

## 例

例えば、同義語ファイル `synfile.xml` を追加モードで追加するには、以下のコマンドを使用します。

```
synonymTool importSynonym
-synonymFile $HOME/sql1lib/misx/xmlsynfile.xml
-collectionName tigertail_DBCP1208_TS867530_0000
-replace false
-configPath $HOME/sql1lib/db2tss/config
```

## DB2 Text Search の同義語ディクショナリーの除去

同義語ディクショナリーは、コレクションごとに除去する必要があります。このため、あるテキスト検索索引に関して存在するすべてのコレクションに対して Synonym Tool を使用しなければなりません。

### このタスクについて

同義語ディクショナリーを除去するには、以下のコマンドを使用します。

```
synonymTool removeSynonym -collectionName collection-name
-configPath absolute-path-to-config-folder
```

ここで、*collection-name* はテキスト検索コレクションを指定し、*absolute-path-to-config-folder* はテキスト検索構成フォルダーへの絶対パスを指定します。

---

## テキスト検索索引の作成

テキスト検索索引は、テキスト文書から抽出された重要な用語を編集したものです。各用語は、それが抽出された文書に関連付けられています。

検索するテキストを含む列ごとに、テキスト検索索引を 1 回作成します。テキスト検索索引を作成するとき、以下のオブジェクトも作成します。

### ステージング表

この表には、ユーザー表内のすべての変更された行に関する追跡記録が含まれます。

### 補助のステージング表 (オプション)

この表には、整合性処理による、ユーザー表での挿入および更新に関する追跡記録が含まれます。

## イベント表

この表は、索引の更新コマンドの状況またはその処理中に検出したエラーに関する情報を収集します。索引付けの際にエラーが生じた場合、索引更新イベントがイベント表に追加されます。

## ユーザー表でのトリガー

これらは、列内の文書が追加、削除、または変更されたときに、ステージング表に情報を追加します。この情報は、次に索引付けをするときに、索引の同期のために必要となります。

**注:** **LOAD** コマンドを使用して文書にデータを追加した場合、トリガーは活動化されないため、ロードされた文書にインクリメンタルに索引付けすることはできません。その代わりに、トリガーを活動化する **IMPORT** コマンドを使用してください。あるいは、整合性処理用の補助インフラストラクチャーを追加することもできます。これにより、**LOAD INSERT** コマンドなどを使用して、変更が認識されます。

テキスト検索索引は、作成後は空であるため、更新されるまでは検索できません。テキスト検索索引の作成時に、スケジューラーが、テキスト検索索引の更新が必要かどうかを定期的に検査して、必要に応じて更新コマンドを実行する場合に使用する頻度を指定できます。

## テキスト検索索引の作成

データベースで DB2 Text Search を使用可能にしたなら、検索対象にしたいテキストが含まれている列にテキスト検索索引を作成することができます。

### 始める前に

テキスト検索索引を作成するには、次のいずれかの許可レベルが必要です。

- 索引表に対する **CONTROL** 特権
- 索引表に対する **INDEX** 特権と、データベースに対する **IMPLICIT\_SCHEMA** 権限または索引表スキーマに対する **CREATEIN** 特権のいずれか
- **DATAACCESS** を持つ **DBADM** 権限

この表の主キーが存在している必要があります。主キーが存在しない場合は、索引を作成する前に主キーを作成してください。

### このタスクについて

表からテキスト検索索引に文書の変更を手動で適用しない場合は、**UPDATE FREQUENCY** パラメーターを指定して、自動更新をスケジュールできます。**UPDATE MINIMUM** パラメーターを使用して、表に対して行われた変更が最小回数に達した場合のみ更新を実行するかどうかを制御します。例えば、**MYSHEMA.MYTEXTINDEX** を、最低 5 つの変更が行われてから更新するようにし、更新サービスに毎週月曜日と水曜日の深夜 0 時と正午 12 時にチェックを行わせるように指定する場合は、次のコマンドを発行します。

```
db2ts "CREATE INDEX MYSCHEMA.MYTEXTINDEX FOR TEXT ON PRODUCT(NAME)
      UPDATE FREQUENCY d(1,3) h(0,12) m(0) UPDATE MINIMUM 5"
```

```
CALL SYSPROC.SYSTS_CREATE('myschema', 'myTextIndex', 'product (name)',
'UPDATE FREQUENCY D(1,3) H(0,12) M(0)' 'UPDATE MINIMUM 5', 'en_US', ?)
```

索引の作成時に、**LANGUAGE** オプションを使用してロケール (言語と地域) を指定できます。文書を自動的に走査してロケールを判別させる場合は、**LANGUAGE** を **AUTO** に設定します。**LANGUAGE** を指定しない場合は、デフォルト値が使用されます。このデフォルトは、**DEFAULTNAME='LANGUAGE'** になっている

**SYSIBMTS.TSDEFAULTS** の **DEFAULTVALUE** によって決まります。(この場合、**DEFAULTVALUE** はデータベースでテキスト検索が使用可能にされるときに設定されます。データベース・テリトリーがいずれかのサポートされている文書ロケールにマップできる場合、この値はデータベース・テリトリーから決定されます。サポートされている文書ロケールの判別にデータベース・テリトリーが使用できない場合は、**DEFAULTVALUE** は **AUTO** に設定されます。)

#### 制約事項

- 索引内のテキスト列は、以下にあげる、サポートされているタイプのいずれかでなければなりません。
  - CHAR
  - VARCHAR
  - LONG VARCHAR
  - CLOB
  - GRAPHIC
  - VARGRAPHIC
  - LONG VARGRAPHIC
  - DBCLOB
  - BLOB
  - XML
- オブジェクトに関連するテキスト検索は、DB2 の命名規則に従っている必要があるだけでなく、ID に以下の文字のみが含まれている必要もあります。
  - [A-Za-z][A-Za-z0-9@#\$\_]\* または
  - "[A-Za-z ][A-Za-z0-9@#\$\_ ]\*"この制限は、以下のものに適用されます。
  - テキスト検索索引を含むスキーマの名前
  - テキスト検索索引が関連付けられている表の名前
  - テキスト列の名前
  - テキスト検索索引の名前

#### 手順

次のいずれかの方法でテキスト検索索引を作成します。

- **CREATE INDEX** コマンドを発行します。

```
db2ts "CREATE INDEX index-name FOR TEXT ON table-name (column-name)"
```
- **SYSPROC.SYSTS\_CREATE** ストアド・プロシージャを呼び出します。

```
CALL SYSPROC.SYSTS_CREATE('index-schema', 'index-name', 'table-name'
(column-name), 'options', 'locale', ?)
```

注: ストアド・プロシージャを使用する際に、スキーマ名および索引名は大文字と小文字が区別されます。



## 例

例えば、SAMPLE データベースの PRODUCT 表に、PRODUCT ID、NAME、PRICE、DESCRIPTION などの列があるとします。NAME 列に対して MYSCHEMA.MYTEXTINDEX というテキスト検索索引を作成する場合は、次のようにコマンドを発行するかストアード・プロシージャを呼び出します。

```
db2ts "CREATE INDEX MYSCHEMA.MYTEXTINDEX FOR TEXT ON PRODUCT(NAME)"
CALL SYSPROC.SYSTS_CREATE('MYSCHEMA', 'MYTEXTINDEX', 'PRODUCT(NAME)', '', 'en_US',?)
```

同様に、XML 列 DESCRIPTION に対して MYSCHEMA.MYXMLINDEX というテキスト検索索引を作成する場合は、次のコマンドを入力します。

```
db2ts "CREATE INDEX MYSCHEMA.MYXMLINDEX FOR TEXT ON PRODUCT(DESCRIPTION)"
```

または

```
CALL SYSPROC.SYSTS_CREATE('MYXMLINDEX', 'MYXMLINDEX',
  'PRODUCT (DESCRIPTION)', '', 'en_US', ?)
```

## バイナリー・データ・タイプでテキスト検索索引を作成する

テキスト検索索引の作成時に、バイナリー列にコード・ページを指定するオプションがあります。これを指定すると、DB2 Text Search エンジンでは文字のエンコード方式を識別することができます。

### このタスクについて

テキスト検索索引の作成時にコード・ページを指定するには、以下のコマンドを使用します。

```
db2ts "CREATE INDEX index-name FOR TEXT ON table-name
CODEPAGE code-page"
```

BLOB または FOR BIT DATA などのバイナリー・データ・タイプを含む列にデータを保管する場合、データは変換されません。つまり、文書は元のコード・ページを保持するため、テキスト検索索引の作成時に 2 つの異なるコード・ページがあることになり、問題の原因になる場合があります。したがって、データベースのコード・ページを使用するか、または db2ts **CREATE INDEX** コマンドで指定するコード・ページを使用するかを決定しなければなりません。テキスト検索索引の作成にどのコード・ページが使用されたかがわからない場合は、以下のステートメントを使用して検索することができます。

```
db2 "SELECT CODEPAGE FROM SYSIBM.TSINDEXES where INDSHEMA='schema-name'
and INDNAME='index-name'"
```

## サポートされていないデータ・タイプでのテキスト検索索引の作成

文書がサポートされていないデータ・タイプ (ユーザー定義タイプ (UDT) など) の列にある場合は、そのユーザー・タイプを入力としてとり、サポートされているいずれかの出力タイプを提供する関数を、ユーザー自身で用意する必要があります。

### このタスクについて

索引内のテキスト列は、以下にあげる、サポートされているタイプのいずれかでなければなりません。

- CHAR

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

列のデータ・タイプをいずれかの有効なタイプに変換するには、次のいずれかの方法を使用します。

- 変換関数の名前を指定して **db2ts CREATE INDEX** コマンドを実行します。

```
db2ts "CREATE INDEX index-name FOR TEXT ON
      table-name (function-name(text-column-name))"
```

- *function-name* で指定した、テキスト検索でサポートされているタイプのものではない列にあるテキスト文書にアクセスするユーザー定義外部関数 (UDF) を使用して値のデータ・タイプ変換を実行し、サポートされているいずれかのデータ・タイプの値を戻させます。

## 例

以下の例では、UDTTABLE という表に、CLOB(1M) として定義された "COMPRESSED\_TEXT" という名前のユーザー定義タイプ (UDT) の列があります。このデータ・タイプの索引を作成するには、まず、タイプ COMPRESSED\_TEXT の値を受け取る UNCOMPRESS という UDF を作成します。次に、以下のようにしてテキスト検索索引を作成します。

```
db2ts "CREATE INDEX UDTINDEX FOR TEXT ON
      UDTTABLE (UNCOMPRESS(text)) ..."
```

## 例: プレーン・テキストの N-gram および形態素方式の索引の作成

### このタスクについて

以下の指示に従って、DB2 Text Search 索引をセットアップおよび同期して、SAMPLE データベースで形態素方式および N-gram の索引付けを行います。言語的に意味のある中国語を検索します。

### 手順

1. 形態素方式および N-gram の索引付けを行うために表を 2 つ作成します。表には、書名、著者、内容、ISBN 番号、および発行年の列があります。

```
db2 "CREATE TABLE morphobooks (
      isbn VARCHAR(18) not null PRIMARY KEY,
      bookname VARCHAR(30),
      author VARCHAR(30),
      story blob(1G),
      year integer
    )"
```

```
db2 "CREATE TABLE ngrambooks (
      isbn VARCHAR(18) not null PRIMARY KEY,
      bookname VARCHAR(30),
```

```
author VARCHAR(30),
story blob(1G),
year integer
)"
```

2. **CREATE INDEX** コマンドを発行して、MORPHOBOOKS 表の STORY 列のテキスト検索索引を作成します。テキスト検索索引の名前は MORPHOINDEX です。

```
db2ts " CREATE INDEX db2ts.morphoindex FOR TEXT
ON morphobooks (story) LANGUAGE zh_TW
INDEX CONFIGURATION (CJKSEGMENTATION 'morphological')
CONNECT TO sample";
```

3. **CREATE INDEX** コマンドを発行して、NGRAMBOOKS 表の STORY 列のテキスト検索索引を作成します。テキスト検索索引の名前は NGRAMINDEX です。

```
db2ts " CREATE INDEX db2ts.ngramindex FOR TEXT
ON ngrambooks (story) LANGUAGE zh_TW
INDEX CONFIGURATION (CJKSEGMENTATION 'ngram')
CONNECT TO sample";
```

4. 2 つの表にデータをロードします。

```
db2 "import from ./data/books.del of DEL lobs from ./data/
replace into morphobooks";
```

```
db2 "import from ./data/books.del of DEL lobs from ./data/
replace into ngrambooks";
```

books.del ファイルには以下の項目があります。

```
"0-13-086755-4", "book1", "Julie", "books_zh_TW1.lob.0.449/", 2004
```

Books\_zh\_TW1.lob ラージ・オブジェクトの内容は以下のとおりです。

```
■■復■■ 木蘭當■織
不聞機杼聲 唯聞女嘆息
問女何所思 問女何所憶
女亦無所思 女亦無所憶
昨夜見軍帖 可汗大點兵
軍書十二卷 卷卷有爺名
阿爺無大兒 木蘭無長兄
願為市鞍馬 從此替爺征
```

5. 以下のコマンドを発行して、対応する表からのデータとテキスト検索索引を同期します。

```
db2ts "UPDATE INDEX db2ts.morphoindex FOR TEXT CONNECT TO sample";
```

```
db2ts "UPDATE INDEX db2ts.ngramindex FOR TEXT CONNECT TO sample";
```

6. 以上で、形態素方式セグメンテーションと N-gram セグメンテーションの両方で、言語的に意味のある中国語の検索が正常に行われるようになります。

```
db2 "select bookname from morphobooks where contains (story, '軍書') = 1";
```

```
BOOKNAME
-----
book1
```

1 record(s) selected.

```
db2 "select bookname from ngrambooks where contains (story, '軍書') = 1";
```

```
BOOKNAME
-----
book1
```

1 record(s) selected.

形態素方式セグメンテーションの結果は N-gram セグメンテーションと同じであることを、出力が示しています。

- 意味を持たない中国語を検索して、形態素方式セグメンテーションと N-gram セグメンテーションの違いを確認します。

```
db2 "select bookname from morphobooks where contains (story, '書十') = 1";
```

```
BOOKNAME
-----
```

0 record(s) selected.

```
db2 "select bookname from ngrambooks where contains (story, '書十') = 1";
```

```
BOOKNAME
-----
```

book1

1 record(s) selected.

N-gram セグメンテーションのみで書名が返されます。

## 例: リッチ・テキストと専有フォーマットの N-gram および形態素方式の索引の作成

### このタスクについて

以下の指示に従って、DB2 Text Search 索引をセットアップおよび同期して、SAMPLE データベースで形態素方式および N-gram の索引付けを行います。意味を持たない中国語を検索します。

### 手順

- 形態素方式および N-gram の索引付けを行うために表を 2 つ作成します。表には k および b の列があります。ここで、列 k は主キーで、列 b にはリッチ・テキスト・データが含まれます。

```
db2 "create table richtext_morpho(
k varchar(50)not null,
b blob (1G),
primary key(k)
)"
```

```
db2 "create table richtext_ngram(
k varchar(50)not null,
b blob (1G),
primary key(k)
)"
```

- CREATE INDEX** コマンドを発行して、表 RICHTEXT\_MORPHO の列 b のテキスト検索索引を作成します。テキスト検索索引の名前は MORPHOINDEX です。

```
db2ts " CREATE INDEX db2ts.morphoindex FOR TEXT
ON richtext_morpho (b) LANGUAGE zh_CN FORMAT INSO
INDEX CONFIGURATION (CJKSEGMENTATIÖN 'morphological')
CONNECT TO sample";
```

- CREATE INDEX** コマンドを発行して、表 RICHTEXT\_NGRAM の列 b のテキスト検索索引を作成します。テキスト検索索引の名前は NGRAMINDEX です。

```
db2ts " CREATE INDEX db2ts.ngramindex FOR TEXT
ON richtext_ngram (b) LANGUAGE zh_CN FORMAT INSO
INDEX CONFIGURATION (CJKSEGMENTATION 'ngram')
CONNECT TO sample";
```

4. 2 つの表にデータをロードします。

```
db2 "import from ./data/cjk_richtext.del of DEL lobs from ./data/
replace into richtext_morpho ";
```

```
db2 "import from ./data/ cjk_richtext.del of DEL lobs from ./data/
replace into richtext_ngram ";
```

cjk\_richtext.del ファイルには以下の項目があります。

```
"rt_CJK.pdf","rt_CJK.pdf.0.864885/",
"rt_CJK.pdf.doc","rt_CJK.pdf.doc.0.90112/",
"rt_CJK.pdf.txt","rt_CJK.pdf.txt.0.37913/"
```

rt\_CJK.pdf、rt\_CJK.pdf.doc、および rt\_CJK.pdf.txt のファイルの内容はすべて同じです。中国語 (簡体字) のコンテンツには以下のようなセグメントがあります。

```
"如何得可密
是一可密在提供服，可以很方便地生成 Rational
密。但是必成公司的 IBM Rational
License Key Center 的成，才可以可密。下的人被置管理，
并会通子件向其送用于 License
Key Center 的密。有方法可以使成公司的成：
方法 1 - 与下的人系，其使用"成"功能将添加公司成。一旦成功添加，
将收到一封来自 License Key Center
的子件，其中包含了的密和登明。
方法 2 - 除了 License Key Center 管理将添加公司 License Key Center
的成之外，也可以自己行添加"
```

5. 以下のコマンドを発行して、対応する表からのデータとテキスト検索索引を同期します。

```
db2ts "UPDATE INDEX db2ts.morphoindex FOR TEXT
CONNECT TO sample"
```

```
db2ts "UPDATE INDEX db2ts.ngramindex FOR TEXT
CONNECT TO sample"
```

6. 以上で、形態素方式セグメンテーションと N-gram セグメンテーションの両方で、言語的に意味のある中国語の検索が正常に行われるようになります。

```
db2 "select k from richtext_morpho where contains(b,'密')=1"
```

```
K
-----
rt_license.pdf
rt_license.pdf.doc
rt_license.pdf.txt
```

3 record(s) selected.

```
db2 "select k from richtext_ngram where contains(b,'密')=1"
```

```
K
-----
rt_license.pdf
rt_license.pdf.doc
rt_license.pdf.txt
```

3 record(s) selected.

形態素方式セグメンテーションの結果は N-gram セグメンテーションと同じであることを、出力が示しています。

- 意味を持たない中国語を検索して、形態素方式セグメンテーションと N-gram セグメンテーションの違いを確認します。

```
Search for a linguistic meaningless Chinese words.  
db2 "select k from richtext_morpho where contains(b,'可■')=1"  
K  
-----
```

0 record(s) selected.

```
db2 "select k from richtext_ngram where contains(b,'可■')=1"  
K  
-----
```

```
rt_license.pdf  
rt_license.pdf.doc  
rt_license.pdf.txt
```

3 record(s) selected.

N-gram セグメンテーションのみで書名が返されます。

---

## テキスト検索索引の保守

テキスト検索索引を作成した後に、実行する必要がある保守作業がいくつかあります。これらの作業を実行するには、さまざまな管理コマンド、ストアード・プロシージャ、および管理ツールの使用など、いくつかの方法があります。

ルーチンのテキスト検索索引の保守作業には、以下のものが含まれます。

- 定期的な更新の実行

自動更新が実行されるように指定していない場合、テキスト検索索引を更新して、それらが関連付けられた索引付きテキスト列に対する変更を反映させる必要があります。

- イベント表のモニター

イベント表を使用して、文書エラーがあるかどうか、または索引更新の頻度を変更する必要があるかどうかを判別できます。

より頻度の低い保守作業には、テキスト検索索引の変更やドロップなどがあります。

## DB2 Text Search 用の管理コマンド

DB2 Text Search をインスタンス、データベース、表、およびテキスト索引の各レベルで管理するために使用できるいくつかのコマンドがあります。それらのコマンドはすべて、db2ts を使用して実行します。

以下のインスタンス・レベルの管理コマンドを使用して、DB2 Text Search インスタンスの開始と停止、および有用でなくなったテキスト検索索引のクリーンアップを行います。

### db2ts START FOR TEXT

DB2 Text Search インスタンス・サービスを開始します。

**db2ts STOP FOR TEXT**

DB2 Text Search インスタンス・サービスを停止します。

**db2ts CLEANUP FOR TEXT**

有用でなくなったテキスト検索コレクションをクリーンアップします。

以下のデータベース・レベルの管理コマンドを使用して、データベースで DB2 Text Search をセットアップする、または無効にします。また、コマンド・ロックを消去します。

**db2ts ENABLE DATABASE FOR TEXT**

現行データベースでのテキスト検索索引の作成、管理、および使用を有効にします。

**db2ts DISABLE DATABASE FOR TEXT**

データベースに対する DB2 Text Search を無効にし、いくつかのテキスト検索カタログ表およびビューをドロップします。

**db2ts CLEAR COMMAND LOCKS**

データベース内のすべての索引について、コマンド・ロックを削除します。

以下の表レベルおよび索引レベルのコマンドを使用して、表の列に対するテキスト検索索引を作成および操作します。

**db2ts CREATE INDEX**

テキスト検索索引を作成します。

**db2ts DROP INDEX**

テキスト列に関連したテキスト検索索引をドロップします。

**db2ts ALTER INDEX**

テキスト検索索引の特性を変更します。

**db2ts UPDATE INDEX**

テキスト列の現在の内容に基づいて、テキスト検索索引にデータを設定するか、テキスト検索索引を更新します。

**db2ts CLEAR EVENTS FOR TEXT**

索引付けの状況およびエラーに関する情報を提供するイベント・ビューである SYSIBMTS.TSEVENT ビューからイベントを削除します。

**db2ts CLEAR COMMAND LOCKS FOR INDEX**

特定のテキスト検索索引について、すべてのコマンド・ロックを削除します。

**db2ts RESET PENDING FOR TABLE**

テキスト検索のために保守されているすべての従属表を特定し、必要に応じて SET INTEGRITY を実行します。

**db2ts HELP**

**db2ts** コマンド・オプションのリスト、および特定のエラー・メッセージに関する情報を表示します。

## DB2 Text Search のストアード・プロシージャ

DB2 Text Search は、いくつかの管理 SQL ルーチンを提供しています。それらのルーチンを使用して、コマンドを実行し、実行するコマンドの結果メッセージと結果メッセージ理由コードを戻すことができます。

管理 SQL ルーチンを使用して、以下の `db2ts` コマンドを実行できます。

- データベースを有効にする - `SYSPROC.SYSTS_ENABLE`
- データベースを構成する - `SYSPROC.SYSTS_CONFIGURE`
- データベースを無効にする - `SYSPROC.SYSTS_DISABLE`
- テキスト索引を作成する - `SYSPROC.SYSTS_CREATE`
- テキスト索引を更新する - `SYSPROC.SYSTS_UPDATE`
- テキスト索引を変更する - `SYSPROC.SYSTS_ALTER`
- テキスト索引をドロップする - `SYSPROC.SYSTS_DROP`
- テキスト索引のイベントを消去する - `SYSPROC.SYSTS_CLEAR_EVENTS`
- コマンドのロックを解除する - `SYSPROC.SYSTS_CLEAR_COMMANDLOCKS`
- 保留状況をリセットする - `SYSPROC.SYSTS_ADMIN_CMD`
- 非アクティブの索引をクリーンアップする - `SYSPROC.SYSTS_CLEANUP`

## テキスト検索索引の更新

テキスト検索索引は自動または手動で更新できます。自動更新は、テキスト検索索引の更新頻度をどのように定義したかに基づいて実施されます。手動での索引の更新は、コマンドを発行するかストアード・プロシージャを呼び出すことによって行うことができます。

### 始める前に

テキスト検索索引を更新するには、ターゲット表に対する `SYSTS_MGR` ロールと、`CONTROL` 特権または `DATAACCESS` 権限のいずれかが必要です。

### このタスクについて

最初にテキスト検索索引を作成し、更新 (充てん) した後、索引を最新の状態に保つ必要があります。例えば、データベースにテキスト文書を追加したり、データベース内の既存の文書を変更したりする際は、文書を索引付けして、テキスト検索索引の内容とデータベースの内容が同期を保つようにする必要があります。また、データベースからテキスト文書を削除するときは、テキスト検索索引からもその後を除去する必要があります。

テキスト文書の索引付けは時間とリソースを消費するタスクであるため、周期的な索引付けの計画は慎重に立てる必要があります。索引付けにかかる時間は、文書の大きさ、前回のテキスト検索索引の更新以降に追加または変更された文書の数、プロセッサの能力といった、多くの要素に依存します。

`db2ts UPDATE INDEX` コマンドを実行している間、管理ツールの状況オプションを使用して、文書更新の進行状況に関する情報を取得できます。索引の更新がまだ進行中である間に新しい更新を開始すると、新しい更新は失敗します。

- 自動更新



テキスト検索索引の更新を自動的に実行させる場合は、以下のいずれかのコマンドを使用して UPDATE FREQUENCY を設定します。

- db2ts CREATE INDEX
- db2ts ALTER INDEX

UPDATE FREQUENCY パラメーターの最小設定は 5 分です。UPDATE MINIMUM パラメーターは、キューに入っていないなければならないテキスト変更の最少数を指定します。

指定された日数および時間が経過してもステージング表に十分な数の変更がない場合、テキスト検索索引は更新されません。

- 手動更新
- テキスト検索索引を即時に更新したい場合もあります。例えば、テキスト検索索引を作成した後や、索引が空のままになっているとき、データベースに追加したばかりのいくつかのテキスト文書を検索したい場合などです。

テキスト検索索引に表データを追加、つまり表データと索引を同期して索引を更新するには、以下いずれかの方法を使用します。

- UPDATE INDEX コマンドを発行します。  
db2ts "UPDATE INDEX *index-name* FOR TEXT"
- SYSPROC.SYSTS\_UPDATE 管理 SQL ルーチン呼び出します。

## 例

例えば、PRODUCT 表に NAME 列の MYSCHEMA.MYTEXTINDEX と DESCRIPTION 列の MYSCHEMA.MYXMLINDEX という 2 つのテキスト検索索引があるとします。次のようにして、新しい項目を PRODUCT に追加します。

```
INSERT INTO PRODUCT VALUES ('100-104-01', 'Wheeled Snow Shovel', 99.99, NULL, NULL, NULL, XMLPARSE(DOCUMENT '<product xmlns="http://posample.org/wheelshovel" pid="100-104-01"><description><name>Wheeled Snow Shovel</name><details>Wheeled Snow Shovel, lever assisted, ergonomic foam grips, gravel wheel, clears away snow 3 times faster</details><price>99.99</price></description></product>'))
```

新しい項目の情報を検索できるようにするには、次のコマンドを実行します。

```
db2ts "UPDATE INDEX MYSCHEMA.MYTEXTINDEX FOR TEXT"
```

新しい項目の情報を検索できるようにするには、次のストアード・プロシージャを使用します。

```
db2 "call sysproc.systs_update('MYSCHEMA', 'MYXMLINDEX', '', 'en_US', ?)'
```

## 例: 範囲パーティション表での DB2 Text Search 索引のインクリメンタル更新

範囲パーティション表における DB2 Text Search 索引のインクリメンタル更新には、パーティションのアタッチまたはデタッチによる変更を適用するための拡張テキスト保持ステージング・インフラストラクチャーが必要です。

## このタスクについて

拡張ステージング・インフラストラクチャーがテキスト検索索引に対して使用可能になっている場合、文書の更新は更新トリガーによって 1 次ステージング表に取り込まれます。文書の挿入および削除は整合性処理によって補助ステージング表に取り込まれます。

拡張ステージング・インフラストラクチャーが使用可能ではない場合、インクリメンタル更新を使用して、範囲のアタッチまたはデタッチに関連した変更を処理することや、**INSERT** パラメーターを指定した **LOAD** コマンドを使用して追加したパーティションにロードした文書を処理することはできません。基本表と同期化するために、テキスト索引を再作成する必要があります。

デフォルトでは、範囲パーティション表のテキスト検索索引に対して、拡張テキスト保持インフラストラクチャーが追加されます。ただし、インクリメンタル更新でテキスト検索索引がリフレッシュされないシナリオでは、以下の例に示すように、**AUXLOG** オプションを **OFF** に設定した状態でテキスト検索索引を作成できます。

```
db2ts create index sampleix for text on sample(comment) administration tables in
mytablespace index configuration(auxlog off) connect to mydb
```

この場合は、1 次ステージング表のみが追加され、文書の変更はトリガーによって認識されます。アタッチまたはデタッチの操作などからの変更は除外されます。範囲パーティション表の索引の作成時に、**ADMINISTRATION TABLES IN** パラメーターを指定する必要があります。そのようにしないと、エラーが発生します。

## 例

### シナリオ 1: 拡張テキスト検索ステージング・インフラストラクチャーで表のパーティションをアタッチする

1. 範囲パーティション表を作成します。

```
db2 "create table uc_007_customer_archive (pk integer not null
primary key, customer varchar(128) not null,
year integer not null, address blob(1M) not null) partition
by range(year)(starting(2000)ending(2001)every 1)"
```

2. テキスト検索索引を作成します。

```
db2ts "create index uc_007_idx for text on
uc_007_customer_archive (address)
administration tables in mytablespace"
```

3. 索引名とロギング情報を表示します。

```
db2 "select indexname, stagingviewname, auxstagingname
from sysibmts.tsindexes"
```

4. テキスト検索索引を更新します。

```
db2ts "update index uc_007_idx for text"
```

5. 別の表を作成し、その表にデータをインポートします。

```
db2 "create table uc_007_customer_2001 (pk integer not null
primary key,
customer varchar(128) not null, year integer not null,
address blob(1M) not null)"
```

```
db2 "import from uc_007_2001.del of del lobs
from ./data modified by codepage=1208
insert into uc_007_customer_2001"
```

- 新しい表からのデータを新しいパーティションとして追加します。

```
db2 "alter table uc_007_customer_archive attach
partition p2001 starting(2001) ending(2002)
exclusive from uc_007_customer_2001"
```

- 内容を表示します。

```
db2 "select * from sysibmts.sysauxlog_ix253720"
```

出力は以下のとおりです。

PK	GLOBALTRANSID	GLOBALTRANSTIME	OPERATIONTYPE
-----			
0 record(s) selected.			

- 変更を表示できないため、整合性処理が必要です。整合性処理では、従属表はペンディング・モードになります。

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

- 内容を表示します。

```
db2 "select * from sysibmts.sysauxlog_ix253720"
```

以下のエラー・メッセージが返されます。

PK	GLOBALTRANSID	GLOBALTRANSTIME	OPERATIONTYPE
-----			
SQL0668N Operation not allowed for reason code "1" on table "SYSIBMTS"."SYSTSAUXLOG_IX253720". SQLSTATE=57016			

- テキスト検索ステージング表の整合性処理を実行します。コマンドにより、表のすべてのテキスト索引が処理されます。

```
db2ts "reset pending for table uc_007_customer_archive for text"
db2 "select * from sysibmts.sysauxlog_ix253720"
```

出力は以下のとおりです。

PK	GLOBALTRANSID	GLOBALTRANSTIME	OPERATIONTYPE
-----			
1	x'000000000002215B'	x'20081020204612500381000000'	1
2	x'000000000002215B'	x'20081020204612500602000000'	1
3	x'000000000002215B'	x'20081020204612500734000000'	1
5	x'000000000002215B'	x'20081020204612500864000000'	1

- インクリメンタル更新を使用して、新しくアタッチされたパーティションからのデータを処理します。

```
db2ts "update index uc_007_idx for text"
```

## シナリオ 2: 拡張テキスト検索ステージング・インフラストラクチャーで表のパーティションをデタッチする

- パーティションからの表を変更する。

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

以下のメッセージが返されます。

```
SQL3601W ステートメントにより 1 つ以上の表が自動的に SET INTEGRITY
ペンディング状態になりました。
SQLSTATE=01586
```

- RESET PENDING** コマンドを発行して、テキスト検索ステージング表の整合性処理を実行します。

```
db2ts "reset pending for table uc_007_customer_archive for text"
```

インクリメンタル更新を使用して、新しくデタッチされたパーティションからのデータを処理します。

```
db2ts "update index uc_007_idx for text"
```

## テキスト検索索引イベントの消去

索引のイベント・ビュー内のメッセージが不要になった場合、それらを消去 (削除) することができます。

### 始める前に

権限の要件など、詳しくは、**CLEAR EVENTS FOR INDEX** コマンドまたは **SYSTS\_CLEAR\_EVENTS** プロシージャの説明を参照してください。

### このタスクについて

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は、テキスト検索索引のイベント・ビューに保管されます。この情報は、問題の原因を判別するのに役立ちます。

### 手順

テキスト検索索引のイベント・ビューを消去するには、以下のいずれかの方法を使用します。

- 以下のように、**db2ts CLEAR EVENTS FOR INDEX** コマンドを実行する。

```
db2ts "CLEAR EVENTS FOR INDEX index-name FOR TEXT"
```
- 以下のように、**SYSPROC.SYSTS\_CLEAR\_EVENTS** 管理 SQL ルーチンを使用する。

```
CALL SYSPROC.SYSTS_CLEAR_EVENTS('index-schema',  
'index-name', 'local', ?)
```

## テキスト検索索引の変更

テキスト検索索引の更新プロパティを変更することができます。

### 始める前に

権限の要件など、詳しくは、**ALTER INDEX** コマンドまたは **SYSTS\_ALTER** プロシージャの説明を参照してください。

### 手順

索引を変更するには、以下のいずれかの方法に従ってください。

- 次のコマンドを実行します。

```
db2ts "ALTER INDEX index-name FOR TEXT update-characteristics"
```

ここで *update-characteristics* は、テキスト検索索引の更新頻度などの特性です。
- 次のようにして、**SYSPROC.SYSTS\_ALTER** 管理 SQL ルーチンを呼び出します。

```
CALL SYSPROC.SYSTS_ALTER('db2ts', 'myTextIndex', 'alter-option', 'en_US', ?)
```

ここで *alter-option* は、テキスト検索索引の更新頻度などの特性です。

## タスクの結果

テキスト検索索引が他の操作によってロックされている場合を除き、テキスト索引のプロパティは新しい値で更新されます。他の操作でロックされている場合は、エラー・メッセージが表示され、テキスト検索索引が現在ロックされていて変更できないことが通知されます。

## 例

いずれの方法でも、テキスト検索索引の更新頻度と、更新を起動する変更の最少数の両方を変更することができます。(パラメーターを指定しない場合、現行の設定が更新されないまま残されます。) 例えば、テキスト検索索引 MYTEXTINDEX の更新頻度を変更して、100 以上の変更が索引付きの列に対して行われた場合に、月曜日から金曜日の昼の 12 時と午後 3 時に更新されるようにするには、以下のコマンドを発行します。

```
db2ts "ALTER INDEX MYTEXTINDEX FOR TEXT  
UPDATE FREQUENCY d(1,2,3,4,5) h(12,15) m(00) UPDATE MINIMUM 100"
```

MYTEXTINDEX の定期的な更新を停止するには、以下のコマンドを発行します。

```
db2ts "ALTER INDEX MYTEXTINDEX FOR TEXT UPDATE FREQUENCY NONE"
```

## テキスト検索索引状況の表示

データベース内の現行のテキスト検索索引に関する情報を得るには、管理ビューを照会するか、または管理ツールを使用することができます。

### このタスクについて

テキスト検索索引プロパティは、SYSIBMTS.TSINDEXES 管理ビューで表示できます。例えば、すべてのテキスト検索索引をその状態とともにリストするには、以下の照会を発行します。

```
db2 "select indschema, indname, indstatus from SYSIBMTS.TSINDEXES"
```

管理ツールを使用して、すべてのテキスト検索コレクションとそれらのプロパティの状況を確認するには、以下のコマンドを使用します。

```
adminTool status -configPath absolute-path-to-config-folder
```

## DB2 Text Search コレクションのロケーションの変更

例えば、コンピューターおよびディスクの管理やメンテナンスを目的として、コレクションのロケーションの変更が必要な場合があります。

### 始める前に

SYSIBMTS.TSINDEXES 表のコレクション・ロケーションが空の場合にのみ、テキスト検索コレクションのロケーションを変更できます。

### このタスクについて

コレクションのロケーションを変更するには、以下のようになります。

## 手順

1. コレクション・ロケーションが空であることを確認します。

```
db2 "select indschema, indname, collectiondirectory, collectionnameprefix
from sysibmts.tsindexes"
```

2. ターゲットのコレクションにディレクトリー情報がない場合は、DB2 Text Search サーバーを停止します。
3. コレクション構成 collection.xml ファイルを編集します。コレクション構成ファイルのデフォルト・ロケーションは、  
<ECMTS\_HOME>%config%collections%<collection\_name>%collection.xml です。
  - a. 索引データのロケーションを指定します。

```
<indexes>
  <index>
    <type>Text</type>
    <path><directory_name></path>
```

- b. 同義語構成のロケーションを指定します。

```
<indexes>
  <index>
    <type>Synonym</type>
    <path><directory_name></path>
```

### 注:

- XML で必要なエスケープ文字。例えば、「%」を使用して円記号 (Windows でデフォルトのパス区切り記号) をエスケープします。
- コレクション構成および索引データがコレクション・ディレクトリーにある場合は、collection.xml ファイルのロケーションの相対パスを指定できません。以下に例を示します。

```
<indexes>
  <index>
    <type>Synonym</type>
    <path>data/text</path>
```

4. collection.xml ファイルの変更を保存します。
5. DB2 Text Search サービスを再始動します。

## テキスト検索索引のバックアップとリストア

### 手順

- DB2 Text Search 索引があるデータベースをバックアップするには、次のようにします。

1. DB2 Text Search 索引のテキスト索引ロケーションの現在のリストを取得します。

```
db2 "select indschema, indname, collectiondirectory, collectionnameprefix
from sysibmts.tsindexes"
```

collectiondirectory の値が指定されていない場合は、**defaultDataDir** パラメーターを使用してロケーションが設定されます。

2. DB2 Text Search の管理コマンドが実行されていないことを確認します。
3. DB2 Text Search サービスを停止します。

```
db2ts stop for text
```

4. データベースをバックアップします。次のコマンドを発行します。

```
db2 backup database db_name
```

5. テキスト検索構成、索引ディレクトリー、およびサブディレクトリーをバックアップします。

6. DB2 Text Search サービスを再始動します。

• DB2 Text Search 索引があるデータベースをリストアするには、次のようにします。

1. DB2 Text Search の管理コマンドが実行されていないことを確認します。

2. DB2 Text Search サービスを停止します。

```
db2ts stop for text
```

3. データベースをリストアします。 次のコマンドを発行します。

```
db2 restore database db_name
```

4. テキスト検索構成および索引のロケーションのバックアップを前と同じパスにリストアします。

5. DB2 Text Search サービスを再始動します。

```
db2ts start for text
```

## テキスト検索索引のドロップ

テキスト列でテキスト検索を行う予定のない場合、そのテキスト検索索引をドロップすることができます。

### 始める前に

権限の要件など、詳しくは、DROP INDEX のコマンドの説明、またはプロシージャー SYSTS\_DROP を参照してください。

### このタスクについて

テキスト検索索引をドロップすると、以下の他のオブジェクトもドロップされます。

- 索引ステージング表およびイベント表
- ユーザー表でのトリガー

テキスト検索索引が関連付けられたスケジュールを持つ場合、実行しているタスクがないことを確認してください。そうでない場合、スケジュールされたタスクを手動で削除しなければならない場合があります。

表スペースをドロップする前に、常に表のテキスト検索索引をドロップしてください。テキスト検索索引を含む表スペースをドロップすると、「孤立したコレクション」と呼ばれるものを作成することになります。テキスト検索索引を作成すると、コレクション (索引のファイル・システム表現) が自動的に生成される名前で作成されます。索引がドロップされた後にコレクションが残ると、以下の事項も真である場合に、将来の照会で問題が起こります。

- 同じデータベース接続が使用されている
- 表が同じ表名で作成されている
- 同じ名前のテキスト索引が以前にこの表で作成されている
- 以前と同じ照会が再発行されている

この場合、キャッシュされた照会プランが再使用され、照会結果が誤ったものになる場合があります。

**db2ts CLEANUP FOR TEXT** コマンドは、廃止されたコレクションおよび関連するテキスト索引カタログ・レコードのみをドロップできます。この場合は、管理ツールを使用して、孤立したコレクションを除去できます。

テキスト検索に使用可能なデータベースをドロップする場合は、孤立したコレクションを防ぐため、すべてのテキスト検索索引がドロップされたことを確認してください。

## 手順

テキスト検索索引をドロップするには、以下のいずれかの方法に従ってください。

- 次のように **DROP INDEX** コマンドを発行します。

```
db2ts "DROP INDEX index-name FOR TEXT"
```

- 次のようにして、SYSPROC.SYSTS\_DROP ストアド・プロシージャを呼び出します。

```
CALL SYSPROC.SYSTS_DROP('index-schema', 'index-name', 'locale', ?)
```

ここで、*locale* は、en\_US などの 5 文字のロケール・コードであり、ログ・ファイルに書き込まれるメッセージの言語を指定します。

## 次のタスク

**注:** テキスト検索索引をドロップした後に孤立したコレクションが存在する場合、管理ツールを使用してそれらを除去することができます。

テキスト検索索引をドロップした後、同じテキスト列に新規のテキスト検索索引を作成する予定である場合は、まずデータベースから切断し、それから再接続する必要があります。

## 例: DB2 Text Search 索引更新のスケジューリング

DB2 Text Search 索引更新をスケジュールし、実行結果を確認します。

### 始める前に

スケジューラー・ジョブを開始する前に、次のタスクを実行します。

1. ATS\_ENABLE レジストリー変数を設定します。
2. SYSTOOLSPACE 表スペースが存在することを確認します。
3. データベースがアクティブになっていることを確認します。

DB2 Text Search 索引更新のスケジューリングの前提条件について詳しくは、管理用タスク・スケジューラーの設定についてのトピックを参照してください。

### このタスクについて

DB2 スケジューラーを使用してスケジューラー・タスクを作成し、指定された頻度でタスクを実行します。



## 手順

1. テキスト検索索引を作成し、更新頻度を指定します。

```
db2ts "create index simix for text on simple(comment)
update frequency (D(*) H(*) M(30))"
```

2. データベースに接続します。

```
db2 connect to testdb
```

3. スケジューラー・タスク名を検索します。

```
db2 "select indexidentifier from sysibmts.tsindexes"
```

以下の手順では、索引 ID の数字部分は 12345 であるとし、そのため、スケジューラー名は TSSCH\_12345 になります。

4. SYSTOOLS.ADMIN\_TASK\_LIST 管理ビューでスケジューラー・タスクを検索します。

```
db2 "select * from systools.admin_task_list"
```

5. テキスト索引更新の状況を確認します。

```
db2 "select * from sysibmts.tsevent_123456"
```

6. メッセージは表示されないものの、更新にデータを使用できた場合は、スケジューラー・タスクが開始されたことを確認します。

```
db2 "select * from systools.admin_task_status"
```

それ以外の場合は、スケジューラー・タスク名を使用して、SELECT 操作を前述の例の新しいスケジューラー・タスクに属するデータに制限します。

```
db2 "select * from systools.admin_task_status
where name = 'TSSCH_12345'"
```



---

## 第 8 章 テキスト検索索引を使用する検索

テキスト検索索引にデータを追加した後、その索引を検索できます。DB2 Text Search は、SQL、XQuery、および SQL/XML での検索をサポートします。

以下の検索関数を使用できます。

- SQL 関数の CONTAINS および XML 関数の xmlcolumn-contains は、特定の語句の照会を作成します
- SQL 関数の SCORE は、検出したテキスト文書の適合性を取得します

テキスト検索索引での検索は、タイトル内の単一ワードのオカレンスの照会のような簡単なものから、ブール演算子または用語のランキング調整を使用する照会のような複雑なものまで範囲があります。検索の複雑性を洗練するために役立つ演算子に加えて、同義語ディクショナリーや言語学的サポートなどのフィーチャーによって、テキスト検索索引での検索を強化できます。

---

### DB2 Text Search の検索関数

テキスト検索索引を更新した後に、CONTAINS または SCORE SQL スカラー検索関数を使用して、または xmlcolumn-contains 関数を使用して検索できます。

テキスト検索索引での検索は、タイトル内の単一ワードのオカレンスの照会のような簡単なものから、ブール演算子または用語のランキング調整を使用する照会のような複雑なものまで範囲があります。検索の複雑性を洗練するために役立つ演算子に加えて、同義語ディクショナリーや言語学的サポートなどのフィーチャーによって、テキスト検索索引での検索を強化できます。

以下の検索関数を使用できます。

- SQL 関数の CONTAINS および XML 関数の xmlcolumn-contains は、特定の語句の照会を作成します
- SQL 関数の SCORE は、検出したテキスト文書の適合性を取得します

スカラー・テキスト検索関数の CONTAINS および SCORE は、SQL とシームレスに統合されます。検索関数は、SQL 照会内で標準 SQL 式を使用する場合と同じ位置に使用できます。SQL SCORE スカラー関数は、テキスト文書が指定のテキスト検索条件に一致する度合いを示す標識を戻します。SQL 照会の SELECT 句によって、どの情報が戻されるかが決まります。

CONTAINS 関数は語句の一致を検索します。この関数は、ワイルドカード文字とともに使用することで、SQL LIKE 述部と類似した方法でサブストリングの一致を検索したり、SQL = 演算子と類似した方法でストリングの完全一致を検索したりすることができます。ただし、CONTAINS 関数の使用と SQL LIKE 述部または = 演算子の使用との間には、大きな違いがあります。LIKE 述部および = 演算子は文書内でパターンを検索しますが、CONTAINS は言語学的な処理を使用します。つまり、検索語のさまざまな形式を検索します。例えば、ワイルドカード文字を使用しなくても、用語 work を検索すると、working および worked を含む文書も戻され

ます。さらに、テキスト検索索引に同義語ディクショナリーを追加して、検索の範囲を拡大することができます。例えば、laptop および ThinkPad をグループにまとめて、それらが notebook computers の検索結果から戻されるようにすることができます。XML 文書では、XML 検索索引構文によってタグ内および属性内のテキストを検索できます。さらに、XQuery 検索は大/小文字を区別します。

DB2 オプティマイザーは、CONTAINS 述部との一致が予想されるテキスト文書の数、およびさまざまな代替アクセス・プランのコストを推定することに注意してください。オプティマイザーは、コストが最小となるアクセス・プランを選択します。

関数 xmlcolumn-contains は、DB2 Text Search エンジンによって実行されるテキスト検索に基づいて、DB2 XML データ列から XML 文書を戻す組み込み DB2 関数です。xmlcolumn-contains を XQuery 式で使用して、特定の文書エレメントの検索に基づいて文書を取り出すことができます。例えば、販売する玩具の製品説明および価格が XML 文書に含まれている場合、xmlcolumn-contains を XQuery 式で使用して説明および価格エレメントを検索し、用語 outdoors があり pool がなく、価格が \$25.00 未満の文書だけを戻すことができます。

xmlcolumn-contains 関数の使用と XQuery contains 関数の使用の間には、重要な違いがあります。XQuery contains 関数は、ストリング内のサブストリングを検索します。それは検索語句と完全に一致するものを探します。それに対して XQuery xmlcolumn-contains 関数は、CONTAINS 関数に類似した機能を持ちますが、XML 列でのみ機能します。さらに、それは検索語句を含む XML 文書を戻しますが、contains は検索用語が見つかったかどうかを示す 1、0、NULL などの値だけを戻します。

---

## フルテキスト検索の方法

テキスト検索索引を介した検索は、SQL ステートメントや XQuery を使用して行うことができます。

### 手順

テキスト検索索引で特定の語や句を検索するには、以下のいずれかの方法を使用します。

- SQL を使用して検索を行います。

SQL ステートメントを使用してテキスト検索索引で特定の語や句を検索するには、CONTAINS 関数を使用して次のようにします。

```
db2 "SELECT column-name FROM table-name
WHERE CONTAINS (...) = 1"
```

例えば、以下の照会では、PRODUCT 表からさまざまな雪かき (snow shovel) の名前と価格を検索します。

```
db2 "SELECT NAME, PRICE FROM PRODUCT
WHERE CONTAINS (NAME, 'snow shovel') = 1"
```

- XQuery を使用して検索を行います。

XQuery を使用してテキスト検索索引で特定の語や句を検索するには、db2-fn:xmlcolumn-contains() 関数を使用します。

例えば、以下の照会では、PRODUCT 表からさまざまな雪かき (snow shovel) の名前と価格を検索します。

```
db2 "xquery for \${info} in db2-fn:xmlcolumn-contains
('PRODUCT.DESCRPTION','"snow shovel"')
return <result> {\${info}/description/name, \${info}/description/price} </result>"
```

注: 使用しているオペレーティング・システム・シェルによって、可変情報のドル記号の前に違うエスケープ文字が必要な場合もあります。直前の例では、UNIX オペレーティング・システムの場合のエスケープ文字として、バックスラッシュ (\) を使用しています。

## 基本検索

ブール演算子および修飾子を検索照会で使用できます。使用する検索語を特定化するほど、結果は厳密なものになります。

### 例

例 1: 「wizard」および「dragon」の用語を含む文書を検索する。明示的なブール演算子が指定されていない場合、デフォルトの演算子は AND です。

```
select title from books where contains(story, 'dragon wizard')=1
```

例 2: 「dragon wizard」の句を含む文書を検索する。これには、例えば、「dragons」の用語を含む文書は含まれません。

```
select title from books where contains(story, "dragon wizard")=1
```

例 3: 「dragon」の用語を含み、任意に「wizard」の用語を含む文書を検索する。両方の用語を含む文書の方がスコアは高くなります。

```
select title from books where contains(story, 'dragon ?wizard')=1
```

例 4: 「dragon」または「wizard」の用語を含み、「hobbit」の用語は含まない文書を検索する。

```
select title from books where contains(story, '(dragon OR wizard) NOT hobbit')=1
```

## ファジー検索

ファジー検索を使用して、検索語とスペルの似ているワードを含む文書を検索します。

ファジー検索を開始するには、句の末尾に波形 (~) 記号を使用して、ファジーの度合いを 0 から 1 までの小数として指定します。小数値が低いほど、ファジーの度合いが増します。特殊文字はファジー検索照会ではサポートされていません。

### 例

ステップ 1. BOOKS という表を作成します。

```
create table books (  
    isbn varchar(18) not null primary key,  
    author varchar(30),  
    story varchar(100),  
    year integer);
```

ステップ 2. STORY 列にテキスト検索索引を作成します。

```
db2ts "create index bookidx for text on books(story) connect to test";
```

ステップ 3. 表にデータをインポートします。

```
insert into books values ('0-13-086755-1','John','The Blue Can',2001)  
insert into books values ('0-13-086755-2','Mike','Cats and Dogs', 2000)  
insert into books values ('0-13-086755-3','Peter','Hats on the Rack',1999)  
insert into books values ('0-13-086755-4','Agatha','Cat among the Pigeons',1997)  
insert into books values ('0-13-086755-5','Edgar','Cars Unlimited',2010)  
insert into books values ('0-13-086755-6','Roy','Carson and Lemon',2008)
```

ステップ 4. テキスト検索索引を更新します。

```
db2ts "update index bookidx for text connect to test"
```

ステップ 5. CONTAINS 関数を使用して、ファジー検索を実行します。

```
select author, year, story from books where contains(story, 'cat~0.4') = 1
```

以下は出力例です。

```
AUTHOR YEAR STORY  
-----  
John 2001 The Blue Can  
Mike 2000 Cats and Dogs  
Agatha 1997 Cat among the Pigeons  
  
3 record(s) selected.
```

関連するスコアを表示するには、ファジーの度合いが増すように変更された次の照会を発行します。

```
select author, year, story, integer(score(story, 'cat~0.3')*1000) as score  
from books where contains(story, 'cat~0.3') = 1 order by score desc
```

以下は出力例です。

```
AUTHOR YEAR STORY SCORE  
-----  
Agatha 1997 Cat among the Pigeons 32  
John 2001 The Blue Can 17  
Mike 2000 Cats and Dogs 17  
Peter 1999 Hats on the Rack 1  
Edgar 2010 Cars Unlimited 1  
  
5 record(s) selected.
```

## 近接検索

近接検索では、指定された単語間の距離内にある検索語を含む文書を検索します。

近接検索を開始するには、句の末尾に波形 (~) 記号を使用して、単語間の距離を有効な整数として指定します。距離を求める際に、文の切れ目は位置が 10 増加したのものとしてカウントされます。特殊文字は近接検索照会ではサポートされていません。

## 例

ステップ 1. BOOKS という表を作成します。

```
create table books (  
    isbn varchar(18) not null primary key,  
    author varchar(30),  
    story varchar(100),  
    year integer);
```

ステップ 2. STORY 列にテキスト検索索引を作成します。

```
db2ts "create index bookidx for text on books(story) connect to test";
```

ステップ 3. 表にデータをインポートします。

```
insert into books values ('0-13-086755-1','John','Understanding Astronomy.'  
,2001)  
insert into books values ('0-13-086755-2','Mike','The cat hunts some mice.'  
,2000)  
insert into books values ('0-13-086755-3','Peter','Some men were standing  
beside the table.',1999)  
insert into books values ('0-13-086755-4','Astrid','The outstanding  
adventure of Pippi Longst.',1997)  
insert into books values ('0-13-086755-6','Agatha','Cat among the pigeons'  
,2004)  
insert into books values ('0-13-086755-7','John','Pigeons land in the square  
, and a cat plays with a ball',2001)  
insert into books values ('0-13-086755-8','Sam','Pigeon on the roof',2007)
```

ステップ 4. テキスト検索索引を更新します。

```
db2ts "update index bookidx for text connect to test"
```

文書内の cat および pigeon という用語 (互いに 4 語以内にある) について近接検索を実行し、DB2 Text Search の CONTAINS 句内で以下の検索構文を使用します。

```
select author, year, substr(story,1,30) as title from books  
where contains(story, '"cat pigeon"~4') = 1
```

## 特殊文字の検索

特殊文字 (一般的な句読点文字など) はテキスト索引更新の一環として索引付けされます。これらの特殊文字は他の照会用語と同様に検索することができます。

文書内の特殊文字を検索するには、照会式にその特殊文字を組み込みます。場合によっては、特殊文字をエスケープする必要があります。

### 特殊文字のエスケープ

照会構文における特殊文字にはさまざまな機能があります。

照会構文で特殊な機能を持つ特殊文字を検索するには、その特殊文字の前に円記号 (¥) を追加してエスケープする必要があります。例えば、以下のようになります。

- スtring 「where?» を検索するには、「where¥?» のように疑問符をエスケープします。
- スtring 「c:¥temp」を検索するには、「c¥:¥¥temp」のようにコロンと円記号 (¥) をエスケープします。

このような特殊文字をエスケープしないと、構文エラーが発生する場合があります。

表 3. 検索する際にエスケープする必要がある特殊文字

特殊文字	エスケープしない場合の動作についての注意
アンパーサンド (&)	
アスタリスク (*)	ワイルドカード文字として使用されます。
アットマーク (@)	アットマークが照会の先頭文字である場合は、構文エラーが生成されます。 xmlxp 式では、アットマークは属性を参照するために使用されます。
大括弧 [ ]	xmlxp 式で、要素と属性の内容を検索するために使用されます。
中括弧 { }	構文エラーを生成します。
円記号 (¥)	
脱字記号 (^)	重み付け (ランキング調整) 用語に使用されます。
コロンの (:)	フィールドの内容での検索に使用されます。
等号 (=)	構文エラーを生成します。
感嘆符 (!)	感嘆符が照会の先頭文字である場合は、構文エラーが返されます。
スラッシュ (/)	xmlxp 式では、スラッシュは要素パス区切り文字として使用されます。
より大記号 (>)、より小記号 (<)	xmlxp 式で、属性値の比較に使用されます。それ以外の場合は、これらの文字により構文エラーが生成されます。
負符号 (-)	負符号が用語の先頭文字である場合は、用語を含まない文書のみが返されます。
括弧 ( )	グループ化に使用されます。
パーセント記号 (%)	検索語がオプションであることを指定します。
正符号 (+)	
疑問符 (?)	ワイルドカード文字として処理されます。
セミコロン (;)	
単一引用符 (')	単一引用符は xmlxp 式を含める際に使用されます。
波形記号 (~)	近接およびファジー検索演算子として処理されます。
垂直バー ( )	

照会構文において特殊機能を持たない特殊文字のエスケープはオプションです。以下の表に、エスケープする必要がない特殊文字の例を示します。

表 4. エスケープする必要がない特殊文字の例

特殊文字	注
コンマ (,)	
ドル記号 (\$)	



表 4. エスケープする必要がない特殊文字の例 (続き)

特殊文字	注
ピリオド (.)	xmlxp 式では、ピリオドは要素内容の検索に使用されます。
ポンド記号 (#)	
下線 ( )	

## 照会用語に隣接する特殊文字

特殊文字が照会内の単語に隣接している場合、同じ順序の特殊文字と単語を含む文書が返されます。

例えば、「30\$」を検索した場合、「30\$」を含む文書は検出されますが、「\$30」を含む文書は検出されません。ただし、「30 \$」(スペースを含む)を検索した場合は、「30」と「\$」を含むすべての文書が検出されます。この場合、文書には「30\$」と「\$30」の両方が含まれます。

特殊文字が照会内のストップワードに隣接している場合、そのストップワードは照会から除去されません。例えば、「at&t」を検索した場合、ストップワード「at」は除去されません。ただし、スペースを含む「at & t」を検索した場合は、ストップワード「at」が除去されます。

特殊文字で 2 つの単語を区切ると、一連のトークンはシーケンスとして検索されます。例えば、「jack\_jones」を検索した場合、「jack\_jones」を含む文書は検出されますが、「jack\_and\_jones」を含む文書は検出されません。

特殊文字に隣接する単語は見出し語化されます。例えば、英語で「cats&dogs」を検索した場合、「cat&dog」を含む文書が検出されます。

特殊文字はワイルドカード検索式で使用できます。例えば、「ja\*\_」を検索した場合は、「jack\_jones」を含む文書が検出されます。ただし、ワイルドカード文字を使用して特殊文字を検出することはできません。例えば、ca\*s を検索した場合、cats、categories、cast-members、または cas を含む文書は検出されますが、ca\_s を含む文書は検出されません。

## 特殊文字の索引付け

DB2 Text Search では、トークン化および言語処理中に、特殊文字を句読点として識別し、索引付けを行います。

特殊文字はトークンの区切り文字です。例えば、「jack\_jones」は「jack」、「\_」、および「jones」という 3 つの別個のトークンとしてトークン化されます。E メール、URL、およびファイル・パスはトークンに分けられます。例:

- Jack\_jones@ibm.com は、jack \_ jones @ ibm . com としてトークン化されます。
- http://www.ibm.com は、http :// www . ibm . com としてトークン化されます。

特殊文字によって、ファイル内のトークンの位置が占有されることはありません。例えば、「jack\_jones」は、「jack」と同じトークンの位置に下線付きで索引付けさ

れます。また、スペースを含む場合も、トークンの位置が特殊文字によって占有されることはありません。例えば、「jack\_jones」は「jack \_ jones」と同じように索引付けされます。

トークンの位置は、完全一致語句の検索と近接検索で使用されます。例えば、文書に jack\_jones という表現が含まれる場合、完全一致語句「jack jones」を検索すると、その文書が検出されます。

一連の特殊文字を個別に索引付けすると、順不同で検索されます。例えば、「#」を検索した場合、「\$#」を含む文書も検出されます。

## CJK 言語における特殊文字

中国語、日本語、または韓国語 (CJK) の文書で特殊文字を含む文字列を見つけるには、照会式に特殊文字を含める必要があります。これは、空白文字を特殊文字の代わりに使用できる CJK 以外の言語とは異なります。

文書で使用される言語が複数の場合、例えば、中国語の文書に英語の用語がいくつか含まれている場合は、CJK 以外の用語の検索で、空白文字を特殊文字の代わりに使用できます。

例えば、索引付き文書に john\_smith が含まれている場合は、「john\_smith」または「john smith」(下線なしの完全一致) を検索することができ、両方の照会で john\_smith を含む文書が返されます。

注: 文字「?」、「\*」、および「\」は、ワイルドカードやエスケープ文字と同じ意味を持ちますが、特殊文字として検索することはできません。

## XML 文書での構造的フルテキスト検索

DB2 Text Search は、XML 文書を検索するための XML 検索の使用をサポートします。

XPath 言語のサブセットをテキスト検索用の拡張機能と共に使用することにより、XML 検索で XML 文書に索引付けをして検索することができます。構造的要素 (タグ名、属性名、および属性値) は、個別に使用することも、照会内のフリー・テキストと結合することもできます。

以下のリストは、XML 検索の主要なフィーチャーを強調しています。

### XML 構造的検索

XML 検索構文をテキスト検索照会で使用することにより、XML 文書内で、構造的要素 (タグ名、属性名、および属性値) およびそれらの要素によって範囲限定されるテキストを検索できます。通常の実験では、XML 文書内の属性フィールドは検索されないことに注意してください。

### XML 照会のトークン化

XML 検索の述部式で XML 照会用語として使用されるテキストは、非 XML 照会用語のテキストがトークン化されるのと同じ方法でトークン化されます。ただし、スペル修正、フィールド化された用語、およびネストされた XML 検索語はサポートされません。同義語、ワイルドカード文字、句、およびレンマ化はサポートされます。

## XML 名前空間の無視

名前空間接頭部は、XML タグおよび属性名の索引付けに際して保存されません。名前空間を宣言および使用する XML 文書を索引付けおよび検索することができますが、名前空間接頭部は索引付けの際に廃棄されて、XML 検索照会から除去されます。

**数値** 属性値を数値と比較する述部はサポートされています。

## 完全一致

述部にストリング引数がある演算子 = (等号) は、ストリング内のすべてのトークンおよび指定されたテキスト・スパン内のすべてのトークンとの完全一致が要求されること、およびそれらの順序も重要であることを意味します。

XML 検索にインプリメントされた XPath のサブセットは、以下の点で標準の XPath と異なります。

- パス式で反復および範囲をサポートしません。
- フィルター式を除去します。つまり、フィルター操作は述部式でのみ許可され、パス式では許可されません。
- 述部式に絶対パス名を使用できません。
- 1 つの軸 (タグ) だけをインプリメントして、順方向の伝搬だけを許可します。

以下の表には、いくつかの有効な XML 検索照会がリストされています。

表 5. 有効な XML 検索照会

照会	説明
/	ルート・ノード。任意の文書
/sentences	最上位タグが sentences の文書
//sentences	任意のレベルに sentences のタグがある文書
sentences	任意のレベルに sentences のタグがある文書
/sentence/paragraph	最上位のタグが sentences で、直接の子タグが paragraph の文書
/sentence/paragraph/	最上位のタグが sentences で、直接の子タグが paragraph の文書
/book/@author	最上位が book タグで、属性 author を持つ文書
/book//@author	最上位が book タグで、任意のレベルで属性 author の下位タグを持つ文書
/book[@author contains("barnes") and @title contains("lemon")]	最上位が book タグで、属性 author および title に、指定された正規化ストリングを含む値のある文書
/book[@author contains("barnes") and (@title contains("lemon") or @title contains("flaubert"))]	最上位が book タグで、指定された author 属性と、指定された 2 つの title 属性のどちらかを持つ文書
/program[. contains("hello, world.")]	最上位が program タグで、そこに少なくともトークン hello および world を含む文書

表 5. 有効な XML 検索照会 (続き)

照会	説明
/book[paragraph contains("flaubert")]//sentence	最上位タグが book タグで、直接の子タグが paragraph でそこに「flaubert」を含み、book タグに対して任意のレベルに下位タグ sentence を持つ文書
/auto[@price <30000]	最上位が auto タグで、属性 price を持ち、その数値が 30000 未満の文書
//microbe[@size <3.0e-06]	任意のレベルに microbe タグがあり、値が 3.0e-06 未満の size 属性がある文書

注: 以下のものは、XML 検索の構文ではサポートされていません。

- /\*
- //\*
- /@\*
- //@\*

通常の検索では、XML 文書内の属性フィールドは検索されません。

## SCORE を使用してテキスト検索索引を検索する

SCORE 関数を使用して、検索指数に一致するエクステントを見つけることができます。

### このタスクについて

SCORE は、文書がどの程度検索基準を満たしているかを示す、0 から 1 の間の倍精度浮動小数点数を返します。文書が照会と一致していればいるほどスコアの関連性は高くなり、結果の値は大きくなります。

スコアは、照会時のテキスト索引コレクションの内容に基づいて動的に計算されるため、非パーティション・テキスト索引でのみ意味があります。

スコア・アルゴリズムは、テキスト索引の形式や照会タイプが違っていると、異なる可能性があります。削除された文書は、テキスト検索索引から削除されるまで、SCORE によって戻される相対値に影響を与えることに注意してください。ただし、スコアで重要な相違点が認められるのは、データの大きいチャンクが索引から削除された場合のみです。

### 例

SAMPLE データベースから、履歴書に Java または COBOL プログラミングの知識に関する記載がある従業員の番号を検索する場合は、以下の照会を実行します。

```
SELECT EMPNO, INTEGER(SCORE(RESUME, 'programmer AND (java OR cobol)') * 100)
AS RELEVANCE FROM EMP_RESUME WHERE RESUME_FORMAT = 'ascii'
ORDER BY RELEVANCE DESC
```

しかし、CONTAINS を使用する次の照会の方が優れています。DB2 オプティマイザは、WHERE 節の CONTAINS 述部を最初に評価するので、表のすべての行に

対して SELECT リストの SCORE 関数を評価しなくて済みます。なお、この方法は照会の SCORE と CONTAINS 引数が同一でなければ使用できませんのでご注意ください。

```
SELECT EMPNO, INTEGER(SCORE(RESUME, 'programmer AND (java OR cobol)') * 100)
AS RELEVANCE FROM EMP_RESUME WHERE RESUME_FORMAT = 'ascii'
AND CONTAINS(RESUME, 'programmer AND (java OR cobol)') = 1
ORDER BY RELEVANCE DESC
```

---

## DB2 Text Search 引数の構文

検索引数は、テキスト文書を検索するために指定する、空白文字で区切られた 1 つ以上の用語とオプションの検索パラメーターから構成されます。

用語を指定すると、検索エンジンは、その用語およびその用語の変換形を含む文書を戻します (デフォルト)。例えば、用語 king を使用して検索を実行すると、king および kings を含む文書が戻されます。複数の用語を使用して検索すると、検索エンジンはすべての用語を含む文書だけを戻します。

完全一致句を使用して検索する場合は、句を引用符で囲みます。検索語とスペルの似ているワードを含む文書を検索するには、ファジー検索を使用します。ファジー検索を実行する一般的な理由は、ミススペルを含む文書を検索結果に含めるためです。

検索語間の距離が特定の距離以内である文書を取得するには、近接検索を実行します。

### 要確認:

- 検索では大/小文字の区別をしないので、スペイン語での完全一致語 "DOS" を検索すると、DOS または dos を含む文書が戻されることがあります。
- テキスト検索照会は、DB2 SQL 照会の限度を超えることができません。

使用する検索語を特定化するほど、結果は厳密なものになります。ただし、以下のようなオプションを使用することによっても、検索を詳細化できます。

### ブール演算子

指定された用語すべてを含む文書を検索するには、AND 演算子を使用します。AND 演算子は、デフォルトの結合演算子です。2 つの用語の間に論理演算子がない場合は、AND が使用されます。

指定された用語すべてを含む文書を検索するには、AND 演算子を使用します。OR 演算子は 2 つ以上の用語をリンクして、いずれかの用語が文書内に存在すれば一致文書として検出します。

### オカレンス修飾子

正符号 (+) を使用して、必須の用語を指定します。正符号 (+) 修飾子は、2 番目の用語が完全一致でなければならないことを示すため、AND 演算子とは異なります。同義語は使用されません。

負符号 (-) または NOT 修飾子を使用して、禁止の用語を指定します。

### ランキング調整修飾子

特定の用語のオカレンスの重要度をより高くするには、キャレット (^) 文字を使用します。キャレット (^) 文字は、指定された数値が 1 よりも大きい

とき、その直前の用語または句をランキング調整します。戻されるリストでの用語または句のランキングを低くする場合は、0 より大きく 1 より小さい数値を指定します。

ランキング調整修飾子は SCORE 関数または ORDER BY 節と共に使用します。

### ワイルドカード文字

1 文字を検索語に追加できることを指定するには、疑問符 (?) 文字を使用します。任意の数の文字を検索語に追加できることを指定するには、\* (アスタリスク) 文字を使用します。これらのワイルドカード文字を使用すると、検索語やデータのスペル変化形が検索され、検索範囲が広がります。

**重要:** 検索語の先頭にワイルドカードのアスタリスク (\*) を使用すると、検索照会のパフォーマンスに悪影響があります。

アスタリスク (\*) を使用するワイルドカード検索では用語展開を適用して文書を検索します。テキスト索引コレクション内で一致する用語の数が展開制限を超えている場合、条件に一致する文書のサブセットのみが戻されます。詳しくは、テキスト検索指数のトピックを参照してください。また、ワイルドカード検索では、特殊文字ではなく通常の文字が検索されます。例えば、US-\*-abc と検索すると、US-xxx-abc、US-x-abc、US-x#-abc などのストリングが検索されますが、US-#-abc は検索されません。

### パーセント記号 (%)

パーセント記号 (%) を使用して、用語または句がオプションであることを指定します。

### 円記号 (¥) エスケープ文字

円記号 (¥) を使用して、検索に特殊文字を含めます。テキスト検索照会では、以下のすべての文字が特殊文字です。

- <
- >
- &&
- ||
- !
- (
- )
- %
- =
- "
- {
- }
- ~
- \*
- ?
- [
- ]

- :
- ¥
- -

## 二重引用符 ("")

検索語または句を引用符 (") で囲むと、完全一致だけが戻されます。

**括弧** 括弧を使用すると、複数の検索語とそれらの検索語間の関係を単一の項目としてまとめることができます。

XML パーサーに送られる XML 検索照会の場合は、XPath 言語のサブセットであいまい条件を使用して照会を記述します。照会パーサーは、照会で使用された構文からあいまい条件を認識します。

検索中の言語に特定な処理については、検索引数パラメーターとしてロケールが想定されます。照会言語は、検索関数の実行に使用されるテキスト検索索引のロケールです。

検索引数の構文は、以下のとおりです。

*Search argument*

**QualifiedClause** ((Operator) (QualifiedClause))

*Operator*

AND | OR

*QualifiedClause*

(Modifier) **Clause** (^number)

*Modifier*

+ | - | NOT

*Clause* 非修飾の用語 | あいまい条件

- 非修飾の用語は、用語または句です。用語は、king などのワード、"king" などの完全一致語、または king\* や king? などのワイルドカードを含む語とすることができます。同様に、句は cabbages and kings などのワードのグループ、"The King and I" などの完全一致句、all the king's h\* や all the kin?'s horses などのワイルドカードを含む句とすることができます。
- あいまいな照会用語は、言語学的な照会パーサーによって構文解析されません。あいまい条件は、その構文によって識別されます。テキスト検索照会で使用されるあいまい条件は @xpath です。例えば、@xpath:'/TagA/TagB[. contains("king")]' となります。

## 例

表 6. ブール演算子

演算子	例	照会結果
AND King AND Lear King Lear	用語 King および Lear を含む文書を戻します。同義語ディクショナリーが使用可能となっている場合、monarch などのワードも戻されることがあります。	

表 6. ブール演算子 (続き)

演算子	例	照会結果
OR	Hamlet OR Othello	Hamlet または Othello のどちらかを含む文書を戻します。

表 7. オカレンス修飾子

修飾子	例	照会結果
NOT -	Hamlet NOT Othello Hamlet -Othello	Hamlet を含み、Othello を含まない文書を戻します。
+	Lear + King	用語 Lear および King を含む文書を戻します。Lear および monarch を含む文書は戻されません。

表 8. その他の修飾子

修飾子	例	照会結果
<i>term1</i> または <i>phrase1</i> ^ <i>number</i> <i>term2</i> または <i>phrase2</i>	Hamlet^2 Othello Hamlet Othello^.5	Hamlet および Othello を含む文書を戻しますが、Hamlet の方により高い重要度を与えます。どちらの照会でも、用語 Hamlet の各オカレンスには Othello に付与されるものと比較して 2 倍の重要度が付与されます。
*	king* k*ng *ing	ワイルドカード文字を持つ検索語の可能な組み合わせを含む文書を戻します。照会の例では、最初の例で king や kingdom などの結果が戻され、2 番目の例で king や kissing などの結果が戻され、3 番目の例で king や skiing などの結果が戻されます。
*	www.*.com	ワイルドカードを使用した検索では、特殊文字を含む用語は戻されません。例の照会では、www.ibm.com を戻しますが、www.#.com は戻しません。
?	mea? be?n ?ean	ワイルドカード文字を持つ検索語の可能な組み合わせを含む文書を戻します。最初の例では meal や mean などの例が戻され、2 番目の例では bean や been などが戻され、3 番目の例では mean や bean などが戻されます。
%	King James %Edition	king および james の両方を含む文書が戻されますが、edition はオプション用語です。



表 8. その他の修飾子 (続き)

修飾子	例	照会結果
"phrase" "exact term" "phrase with wildcard"	"King Lear" "king" "John * Kennedy" "John ? Kennedy"	完全一致の語または句を含んでいる文書を戻します。最初の例では、King Lear が戻されます。2 番目の例では、king という語だけが戻され、kings や kingly などの他の形の語は戻されません。  引用符はワイルドカードと共に使用できます。3 番目の例では John Kennedy にさまざまなミドルネームやそのイニシャルがある場合とない場合の両方のオカレンスが戻されます。4 番目の例では John イニシャル Kennedy が戻されます。
( )	(Hamlet OR Othello) AND plays	以下の用語を含む文書が戻されます。 • 用語 Hamlet または Othello • 用語 plays
¥	¥(1¥+1¥)¥:2	(1+1):2 を含む文書を戻します。照会構文の一部である特殊文字をエスケープするために、円記号 (¥) 文字を使用します。

## XML 文書用の検索構文

XML 検索式を使用しながら、DB2 Text Search エンジンを使用して DB2 XML 列内の XML 文書の特定部分を検索できます。

### 構文

▶▶@xmlxp: ' | XML 検索照会 | ' ◀◀

#### XML 検索照会:

|—location-path—|  
|—[—search-predicate—]—|

#### @xmlxp:

テキスト検索照会を XML 文書上で開始するキーワード。

注: キーワード @xpath は推奨されていません。

#### XML 検索照会

XML 文書を検索するために DB2 Text Search によって使用されるテキスト検索照会。照会は、単一引用符で囲まれます。XML 検索照会は、検索する XML 文書の部分を指定するロケーション・パスおよび検索条件を指定するオプションの述部からなる XML 検索式です。

### *location-path*

XPath 省略構文のサブセットを使用して XML 文書ノードまたは属性を指定する XML 検索式。詳細については、『ロケーション・パス』セクションで説明されています。

### *search-predicate*

DB2 Text Search が XML 文書を検索するときに使用する、オプションの検索条件。詳細については、『検索述部』セクションで説明されています。

DB2 Text Search エンジンは、XML 文書の指定のノードまたは属性で *search-predicate* に指定されたテキストを見つけた場合に、XML 文書を戻します。

## ロケーション・パス

XML 文書でテキスト検索を行うとき、DB2 Text Search はローカル・ノード、属性名、および XPath 構文のサブセットを使用して、XML 文書内のノードおよび属性を指定します。DB2 Text Search は、以下の XML 検索要素をサポートします。

- ローカル・ノード名または属性名
- . (ピリオド)。現行コンテキスト・ノードとして
- / または //。区切り記号として
- @。属性の省略記号として
- 名前の正規化

XML ノードおよび属性名は、DB2 Text Search エンジンによる使用のために索引付けされる時、正規化されません。それらはいずれかの方法で小文字に変換されたり、トークン化されたり、変更されたりしません。XML ノードおよび属性名で大/小文字の区別は重要なので、照会でノードや属性名に使用するストリングが、文書に含まれている名前と正確に一致しなければ、マッチングは生じません。

- 名前空間処理

テキスト検索索引を作成するとき、XML 名前空間指定子を含む XML 文書を使用できますが、名前空間指定子は索引内に保存されません。例えば、タグ `<nsdoc:heading>` は見出しの下でのみ索引付けされて、照会用語 `term @xmlxp:'/nsdoc:heading'` は `@xmlxp:'/heading'` に構文解析されます。XML ネーム・スペースの接頭部は、照会の構文解析の際に廃棄されます。

## 例

以下の例は、製品情報の説明ノードで用語 `snow shovel` を検索するために XML 検索を使用する、有効なテキスト検索照会です。

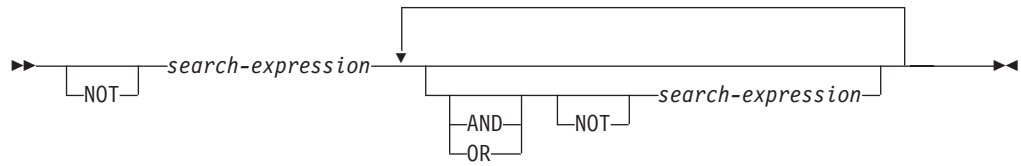
```
@xmlxp:'/info/product/description[. contains("snow shovel")]'
```

以下の例は、`parent::node()` のための XML 検索の省略語である `..` を使用しているため、XML 検索を使用する有効なテキスト検索照会ではありません。

```
@xmlxp:'/info/product/description/../@ID[. contains("A2")]'
```

## 検索述部

### 構文



### *search-expression*

DB2 Text Search の XML 検索照会 DB2 Text Search は、検索式を使用して、XML 文書内でノードまたは属性値を検索します。

以下の演算子を使用して、検索式を作成できます。

- 論理演算子: AND、OR、および NOT
- 包含演算子: 包含および除外します
- 比較演算子: =、>、<、>=、<=、および !=

#### 注:

比較演算子は、属性値のみに適用できます。ノード値には適用されません。

そのため、`<root><aaa id="10">100</aaa><aaa id="11">101</aaa></root>` に対しては、以下の照会は無効です。

```
select id from testtable where contains(item,'@xmlxp:'/root/aaa[. > 20]')>0
```

有効な照会の例としては、以下のようになります。

```
select id from testtable where contains(item,'@xmlxp:'/root/aaa/@id[. > 20]')>0
```

比較演算子および包含演算子を論理演算子 AND、OR、および NOT と組み合わせ、複雑な検索式を作成できます。さらに、括弧を使用して式をグループ化することもできます。

文字列を囲むには、単一引用符または二重引用符を使用します。引用符を含む文字列は、同じタイプの引用符で囲むことができません。例えば、単一引用符で囲まれた文字列は、単一引用符を含むことができません。

XML 検索述部で、比較演算子は論理演算子よりも優先され、すべての論理演算子は同じ優先順位となります。意図する計算順序を確実にするために、括弧を使用できます。

XML 文書内のフリー・テキスト (タグ自体の中ではなくタグ間にあるテキスト) および属性値は、索引付けの前に正規化されます。XML 照会内のフリー・テキスト (包含演算子内にある) は、非 XML 照会内にあるときと同じ方法で正規化されません。

### 例

以下の例では、XML 検索照会を使用して、製品説明に用語 `snow shovel` が含まれ、価格が \$29.99 未満の製品を検索します。

```
@xmlxp:'/info/product [(description contains("snow shovel"))
and (@price < 29.99)]'
```

## 比較式

比較式は、属性の値を指定の値と比較します。

## 構文

▶▶ *path-expression* *operator* *literal* ◀◀

### *path-expression*

XML 検索の省略構文のサブセットを使用して、ノードまたは属性を指定するパス式。

### *operator*

実行する比較のタイプ。演算子は以下のタイプの 1 つにすることができます。

- = *path-expression* 値は、*literal* と等価です。
- > *path-expression* 値は、*literal* よりも大です。
- < *path-expression* 値は、*literal* よりも小です。
- >= *path-expression* 値は、*literal* 以上です。
- <= *path-expression* 値は、*literal* 以下です。
- != *path-expression* 値は、*literal* と等しくありません。

### *literal*

*path-expression* ノードまたは属性値と比較するために使用される文字列または数値。

文字列を単一または二重引用符で囲みます。引用符を含む文字列は、同じタイプの引用符で囲むことができません。例えば、単一引用符で囲まれた文字列は、単一引用符を含むことができません。円記号 (¥) を使用して、二重引用符 (") をエスケープします。

文字列に二重引用符が含まれる場合、その文字列を単一引用符で囲むことができます。次の例では、二重引用符を含む文字列が単一引用符で囲まれている状態を示しています。

```
'he said "Hello, World"'
```

文字列に単一引用符が含まれる場合、その文字列をエスケープした二重引用符で囲むことができます。次の例では、単一引用符を含む文字列が二重引用符で囲まれている状態を示しています。

```
¥"the cat's toy¥"
```

句、ワイルドカード、同義語などの DB2 Text Search フィーチャーは、XML 検索照会ではサポートされていません。

## 例

次の例では、= 演算子を使用して、文字列 100-200-101 に等しい製品 ID を検索します。

```
@xmlxp:'/info/product/@pid[. = "100-200-101" ]'
```

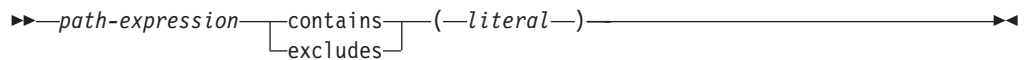
注:

文字列引数でサポートされている比較演算子は `=` と `!=` のみです。  
`<`、`<=`、`>`、`>=` は使用できません。数値引数では、6 つのすべての演算子がサポートされています。数値引数は、属性値との比較のためにはサポートされていますが、タグ (ノード) の内容との比較のためにはサポートされていません。

## 包含式

包含式は、ノードまたは属性の値に指定の値が含まれるかどうかを判別します。

## 構文



*path-expression*

XML ノードまたは属性を指定する XML 検索式。

### contains

*path-expression* 値に *literal* が含まれることを示す式。

### excludes

*path-expression* 値から *literal* が除外されていることを示す式。

*literal*

*path-expression* ノードまたは属性値と比較するために使用される文字列。

文字列を囲むには、単一引用符または二重引用符を使用します。文字列は、それを囲む引用符と同じタイプの引用符を含むことができません。例えば、単一引用符で囲まれた文字列は、単一引用符を含むことができません。円記号 (¥) を使用して、二重引用符 (") をエスケープします。

文字列に二重引用符が含まれる場合、その文字列を単一引用符で囲むことができます。

次の例では、二重引用符を含む文字列が単一引用符で囲まれている状態を示しています。

```
'he said "Hello, World"'
```

文字列に単一引用符が含まれる場合、その文字列をエスケープした二重引用符で囲むことができます。次の例では、単一引用符を含む文字列が二重引用符で囲まれている状態を示しています。

```
¥"the cat's toy¥"
```

## 例

以下の例では、パス式に XQuery 省略構文を使用して、説明ノードが用語 `ice scraper` を除外することを指定します。

```
@xmlxp: '/info/product/description[. excludes('ice scraper')]'
```

---

## フルテキスト照会のパフォーマンスの強化

検索時のパフォーマンスを強化するには、以下にあげるアプローチの 1 つ以上を使用します。

### 手順

- EXPLAIN ステートメントを使用して、SQL で検索を行う際の DB2 オプティマイザーのアクセス・プランをチェックします。
- SCORE 関数を使用する際は必ず CONTAINS 関数を使用するようにします。加えて、処理の重複を避けるために、CONTAINS 関数に指定するストリング (つまり、検索指数と任意の検索オプション) と SCORE 関数に使用するストリング (空白文字を含む) が正確に一致するようにしてください。
- DB2 コンパイラーが適切な表統計を持つようにします。 **RUNSTATS** コマンドを使用して統計を更新します。
- データベースの **STMT\_CONC** 構成パラメーターを確認します。パラメーターが **LITERAL** オプションを使用するよう設定されている場合、テキスト検索照会でパフォーマンスの低下が発生する場合があります。

## 第 9 章 SQL および XML の組み込み検索関数

以下の DB2 組み込み検索関数を DB2 Text Search で使用できます。これらの関数のスキーマは、SYSIBM です。

### CONTAINS

入力テキスト文書がテキスト検索条件と一致するかどうかに応じて、NULL または 0 か 1 の INTEGER 値を返します。

### SCORE

テキスト文書が検索条件に合致する程度を示す、NULL または 0 から 1 の間の DOUBLE 値を返します。

### xmlcolumn-contains

XML データ・タイプの入力テキスト文書がテキスト検索条件と一致するかどうかに応じて、NULL または 0 か 1 の INTEGER 値を返します。

## CONTAINS 関数

CONTAINS 関数は、検索引数に指定した基準を使用してテキスト検索索引を検索し、一致が見つかったかどうかを示す値を返します。

### 関数の構文

```
▶▶ CONTAINS ( (column-name) , (search-argument) [ (1) , (string-constant) ] )▶▶
```

注:

- 1 *string-constant* は、*search-argument-options* の規則に準拠している必要があります。

### search-argument-options:

```
(1)
|-----|
| QUERYLANGUAGE == locale |
| RESULTLIMIT == value   |
|-----|
| SYNONYM == [ OFF ]     |
|                   [ ON ] |
|-----|
```

注:

- 1 同じ節を複数回指定することはできません。

スキーマは SYSIBM です。

## 関数のパラメーター

### *column-name*

検索対象のテキスト検索索引が付けられた列の修飾された名前または無修飾の名前。列は、ステートメント内の **FROM** 節で指定されている表またはビュー内に存在していなければならない、その表の列、またはビューの基礎となる基本表の列はテキスト検索索引と関連付けられていなければなりません (SQLSTATE 38H12)。ビューの列の基礎となる式は、基礎表の列への (直接または別のネストされたビューを通した) 簡単な列参照でなければなりません。

### *search-argument*

検索対象の用語の入った文字列値 (LOB 以外) を戻す式。この文字列値は、すべて空白または空文字列ではありません (SQLSTATE 42815)。式の結果である文字列値は 4096 バイト以下でなければなりません (SQLSTATE 42815)。値は Unicode に変換されてから、テキスト検索索引の検索に使用されます。照会ごとの用語の最大数は 1024 を超過してはなりません (SQLSTATE 38H10)。

### *string-constant*

関数で有効な検索索引オプションを指定する文字列定数。

*search-argument-options* の一部として指定できるオプションは、以下のとおりです。

#### **QUERYLANGUAGE = locale**

DB2 テキスト列に対してテキスト検索を実行する際に DB2 Text Search エンジンが使用するロケールを指定します。値はサポートされる任意のロケールです。**QUERYLANGUAGE** を指定しなかった場合、デフォルトであるテキスト検索索引のロケールが使用されます。テキスト検索索引の **LANGUAGE** パラメーターが **AUTO** である場合、**QUERYLANGUAGE** のデフォルト値は **en\_US** です。

#### **RESULTLIMIT = value**

最適化iererが、**SCORE** を取得するために結果セットの各行に対して検索エンジン呼び出すプランを選択する場合、**RESULTLIMIT** オプションはパフォーマンスに影響を及ぼしません。しかし、検索エンジンが結果セット全体に対して一度に呼び出される場合、**RESULTLIMIT** は **FETCH FIRST** 節のような働きをします。

同じ照会で **RESULTLIMIT** を指定する複数のテキスト検索を使用する場合、同じ *search-argument* を使用してください。異なる *search-argument* 値を使用すると、期待される結果が得られない場合があります。

パーティション・テキスト索引の場合、結果の制限が各パーティションに別々に適用されます。

#### **SYNONYM = OFF | ON**

テキスト検索索引に関連付けられた同義語ディクショナリーを使用するかどうかを指定します。デフォルトは **OFF** です。同義語を使用するには、Synonyms Tool を使用して同義語ディクショナリーをテキスト検索索引に追加します。



**OFF** 同義語ディクショナリーを使用しません。

**ON** テキスト検索索引に関連付けられた同義語ディクショナリーを使用します。

この関数の結果は長精度整数 (large integer) です。2 番目の引数が NULL になる可能性がある場合、結果も NULL になる可能性があります。2 番目の引数が NULL であれば、結果は NULL 値です。3 番目の引数が NULL の場合、結果は 3 番目の引数を指定しなかった場合と同様です。CONTAINS は、文書内に検索索引に指定した基準との一致が含まれる場合、整数値の 1 を戻します。それ以外の場合は、0 を戻します。

CONTAINS は、一律の結果を生じない関数です。

注: テキスト検索関数内でパラメーター・マーカを検索索引として使用する場合は、さらに追加の手順が必要です。パラメーター・マーカは、JDBC および ODBC プログラムでプリコンパイルされた場合、タイプを持ちませんが、テキスト検索関数内の検索索引はストリング値に解決されなければなりません。不明なタイプのパラメーター・マーカはストリング値に解決できないため (SQLCODE -418)、パラメーター・マーカを明示的に VARCHAR データ・タイプにキャストする必要があります。

## 例

- 次の照会は、レジュームに COBOL のあるすべての従業員を検索するために使用されます。テキスト検索索引は、大/小文字を区別しません。

```
SELECT EMPNO
FROM EMP_RESUME
WHERE RESUME_FORMAT = 'ascii'
AND CONTAINS(RESUME, 'COBOL') = 1
```

- 次の C プログラムでは、完全一致語 ate が COMMENT 列で検索されます。

```
char search_arg[100]; /* input host variable */
...
EXEC SQL DECLARE C3 CURSOR FOR
SELECT CUSTKEY
FROM CUSTOMERS
WHERE CONTAINS(COMMENT, :search_arg) = 1
ORDER BY CUSTKEY;
strcpy(search_arg, "ate");
EXEC SQL OPEN C3;
...
```

- 次の照会を使用して、fossil fuel のスペイン語に相当する combustible fósil という句を含むオンライン・エッセーを書いた 10 人の学生を検索します。関連付けられたテキスト検索索引用の同義語ディクショナリーが作成されています。必要なのは 10 人の学生だけなので、基礎となるテキスト検索サーバーからの結果の数を制限する RESULTLIMIT オプションを使用して、照会を最適化します。

```
SELECT FIRSTNAME, LASTNAME
FROM STUDENT_ESsays
WHERE CONTAINS(TERM_PAPER, 'combustible fósil',
'QUERYLANGUAGE= es_ES RESULTLIMIT = 10 SYNONYM=ON') = 1
```

## SCORE 関数

SCORE 関数は、検索引数で指定する基準に従ってテキスト検索索引を検索し、特定の文書が、列内の他の文書と比較してその照会をどの程度満たすかを示す関連性スコアを戻します。

### 関数の構文

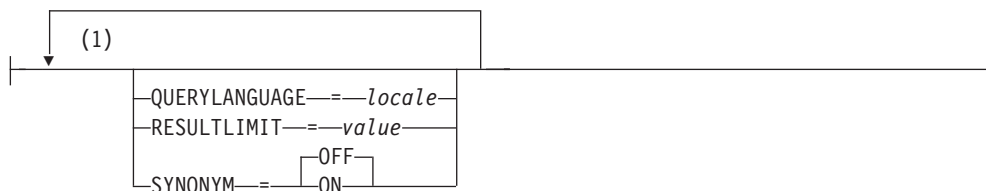
```
SCORE (column-name, search-argument, string-constant)
```

(1)

注:

- 1 *string-constant* は、*search-argument-options* の規則に準拠している必要があります。

### search-argument-options:



注:

- 1 同じ節を複数回指定することはできません。

スキーマは SYSIBM です。

### 関数のパラメーター

#### *column-name*

検索対象のテキスト検索索引が付けられた列の修飾された名前または無修飾の名前。列は、ステートメント内の FROM 節で指定されている表またはビュー内に存在していなければならず、その表の列、またはビューの基礎となる基本表の列はテキスト検索索引と関連付けられていなければなりません (SQLSTATE 38H12)。ビューの列の基礎となる式は、基礎表の列への (直接または別のネストされたビューを通した) 簡単な列参照でなければなりません。

#### *search-argument*

検索対象の用語の入った文字列 (LOB 以外) を戻す式。この文字列値は、すべて空白または空文字列ではありません (SQLSTATE 42815)。式の結果である文字列値は 4096 バイト以下でなければなりません (SQLSTATE 42815)。値は Unicode に変換されてから、テキスト検索索引の検索に使用されます。照会ごとの用語の最大数は 1024 を超過してはなりません (SQLSTATE 38H10)。

#### *string-constant*

関数で有効な検索引数オプションを指定する文字列定数。

*search-argument-options* の一部として指定できるオプションは、以下のとおりです。

**QUERYLANGUAGE = locale**

DB2 テキスト列に対してテキスト検索を実行する際に DB2 Text Search エンジンが使用するロケールを指定します。値はサポートされる任意のロケールです。**QUERYLANGUAGE** を指定しなかった場合、デフォルトであるテキスト検索索引のロケールが使用されます。テキスト検索索引の **LANGUAGE** パラメーターが **AUTO** である場合、**QUERYLANGUAGE** のデフォルト値は **en\_US** です。

**RESULTLIMIT = value**

オブティマイザーが、**SCORE** を取得するために結果セットの各行に対して検索エンジン呼び出すプランを選択する場合、**RESULTLIMIT** オプションはパフォーマンスに影響を及ぼしません。しかし、検索エンジンが結果セット全体に対して一度に呼び出される場合、**RESULTLIMIT** は **FETCH FIRST** 節のような働きをします。同じ照会で **RESULTLIMIT** を指定する複数のテキスト検索を使用する場合、同じ *search-argument* を使用してください。異なる *search-argument* 値を使用すると、期待される結果が得られない場合があります。

パーティション・テキスト索引の場合、結果の制限が各パーティションに別々に適用されます。

注：結果の数が問題である場合は、**RESULTLIMIT** を使用するのではなく、検索項目を調整することによって結果の数を制限します。**RESULTLIMIT** は、指定された数の結果をスコアに関係なく戻すものに過ぎないため、関連性が最も高い文書が含まれない場合があります。

**SYNONYM = OFF | ON**

テキスト検索索引に関連付けられた同義語ディクショナリーを使用するかどうかを指定します。デフォルトは **OFF** です。同義語を使用するには、Synonyms Tool を使用して同義語ディクショナリーをテキスト検索索引に追加します。

**OFF** 同義語ディクショナリーを使用しません。

**ON** テキスト検索索引に関連付けられた同義語ディクショナリーを使用します。

関数の結果は倍精度浮動小数点数になります。2番目の引数が **NULL** になる可能性がある場合、結果も **NULL** になる可能性があります。2番目の引数が **NULL** であれば、結果は **NULL** 値です。3番目の引数が **NULL** の場合、結果は3番目の引数を指定しなかった場合と同様です。

列に、検索引数で指定された検索条件に一致するものが含まれる場合、結果は0より大きく1より小さくなります。一致するものが見つかる頻度が高いほど、結果値も大きくなります。列に一致するものが含まれない場合、結果は0になります。

**SCORE** は一律の結果を生じない関数です。

注: テキスト検索関数内でパラメーター・マーカを検索引数として使用する場合は、さらに追加の手順が必要です。パラメーター・マーカは、JDBC および ODBC プログラムでプリコンパイルされた場合、タイプを持ちませんが、テキスト検索関数内の検索引数はストリング値に解決されなければなりません。不明なタイプのパラメーター・マーカはストリング値に解決できないため (SQLCODE -418)、パラメーター・マーカを明示的に VARCHAR データ・タイプにキャストする必要があります。

## 例

- 以下の照会を使用すると、照会 "programmer AND (java OR cobol)" をより満たす経歴を持つ従業員の順に、従業員と、0 から 100 の間で正規化された関連性値のリストが生成されます。

```
SELECT EMPNO,  
       INTEGER(SCORE(RESUME,  
                   'programmer AND (java OR cobol)') * 100) AS RELEVANCE  
FROM EMP_RESUME  
WHERE RESUME_FORMAT = 'ascii'  
AND CONTAINS(RESUME, 'programmer AND (java OR cobol)') = 1  
ORDER BY RELEVANCE DESC
```

## 使用上の注意

- SCORE 値は、同じテキスト索引コレクションのすべての文書の SCORE 値と比較した場合の、文書の相対的な関連性を反映します。パーティション・データベースの場合、テキスト索引は複数のコレクションから構成されることがありますが、文書のスコアはパーティション間では正規化されません。したがって、テキスト索引コレクション全体で SCORE 値を比較したりソートしたりしても意味はなく、パーティション・テキスト索引内の文書に対する関連性の適切な尺度にはなりません。

---

## xmlcolumn-contains 関数

db2-fn:xmlcolumn-contains 関数は、指定した検索項目について DB2 Text Search エンジンによって実行されるテキスト検索に基づいて、XML データ列から XML 文書のシーケンスを戻します。

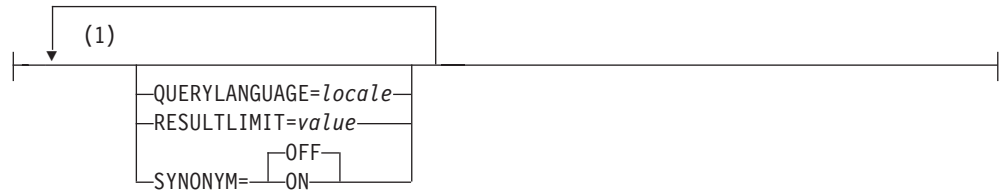
## 構文

```
db2-fn:xmlcolumn-contains(string-literal,search-argument [options-string-literal (1)])
```

注:

- options-string-literal* は、*search-argument-options* の規則に準拠している必要があります。

**search-argument-options:**



注:

- 1 各オプションは一度だけ指定できます。

#### *string-literal*

db2-fn:xmlcolumn-contains で検索する XML データ・タイプ列の名前を指定します。*string-literal* の値は大/小文字が区別されるため、表名と列名は大/小文字を一致させる必要があります。表名またはビュー名を使用して、列名を限定してください。SQL スキーマ名はオプションです。SQL スキーマ名を指定しない場合は、CURRENT SCHEMA の値が使用されます。

列にはテキスト検索索引がなければなりません。

#### *search-argument*

原子文字列値または空のシーケンスを返す式です。文字列を全角スペース文字もしくは空文字列にすることはできません。文字列は、XMLCAST の規則に従って、最大長 4096 バイトでタイプ VARCHAR にキャスト可能でなければなりません。

#### *options-string-literal*

関数に対して有効な検索引数オプションを指定します。

*search-argument-options* の一部として指定できるオプションは、以下のとおりです。

#### **QUERYLANGUAGE = locale**

DB2 テキスト列に対してテキスト検索を実行する際に DB2 Text Search エンジンが使用するロケールを指定します。値はサポートされる任意のロケールです。**QUERYLANGUAGE** を指定しなかった場合、デフォルトであるテキスト検索索引のロケールが使用されます。テキスト検索索引の **LANGUAGE** パラメーターが AUTO である場合、**QUERYLANGUAGE** のデフォルト値は en\_US です。

#### **RESULTLIMIT = value**

最適化エンジンが、SCORE を取得するために結果セットの各行に対して検索エンジン呼び出しプランを選択する場合、**RESULTLIMIT** オプションはパフォーマンスに影響を及ぼしません。しかし、検索エンジンが結果セット全体に対して一度に呼び出される場合、**RESULTLIMIT** は FETCH FIRST 節のような働きをします。

同じ照会で **RESULTLIMIT** を指定する複数のテキスト検索を使用する場合、同じ *search-argument* を使用してください。異なる *search-argument* 値を使用すると、期待される結果が得られない場合があります。

パーティション・テキスト索引の場合、結果の制限が各パーティションに別々に適用されます。複数のテキスト検索を使用した場合に何が起きるかの例、およびその解決方法については、『例』を参照してください。

#### SYNONYM = OFF | ON

テキスト検索索引に関連付けられた同義語ディクショナリーを使用するかどうかを指定します。デフォルトは OFF です。同義語を使用するには、Synonyms Tool を使用して同義語ディクショナリーをテキスト検索索引に追加します。

**OFF** 同義語ディクショナリーを使用しません。

**ON** テキスト検索索引に関連付けられた同義語ディクショナリーを使用します。

## 戻り値

戻り値は、*string-literal* で指定された列内の NULL 以外の XML 値を連結したシーケンスです。NULL 以外の XML 値は、非決定論的な順序で戻されます。XML 値は、*string-literal* で指定された列に *search-argument* を使用する SQL CONTAINS 関数が 1 を返す XML 文書です。該当する XML 値がない場合は、空のシーケンスが戻されます。

*search-argument* が空のシーケンスである場合は、空のシーケンスが戻されます。*search-argument* が空文字列であるか、全角スペース文字を含む文字列である場合は、エラーが戻されます。3 番目の引数が NULL の場合、結果は 3 番目の引数を指定しなかった場合と同様です。

*string-literal* を使用して指定した列にテキスト検索索引がない場合は、エラーが戻されます。

db2-fn:xmlcolumn-contains 関数は db2-fn:sqlquery 関数と関連しており、双方の関数は同じ結果を生成する場合があります。ただし、2 つの関数の引数は、大/小文字の区別の点で異なります。db2-fn:xmlcolumn-contains 関数の最初の引数 *string-literal* は、XQuery によって処理されるため、大/小文字を区別します。DB2 データベース内の表名および列名はデフォルトで大文字になっているため、db2-fn:xmlcolumn-contains の最初の引数は、通常、大文字です。db2-fn:sqlquery 関数の最初の引数は SQL によって処理されるため、ID が自動的に大文字に変換されません。

PRODUCT 表が現行で CURRENT SCHEMA に割り当てられているスキーマにあることを前提とした場合、以下の関数呼び出しは等価であり、同じ結果を返します。

```
db2-fn:xmlcolumn-contains("PRODUCT.DESCRPTION", "snow shovel")
```

```
db2-fn:sqlquery("select description from product  
where contains(description, 'snow shovel') = 1")
```

## 例

以下の例では、DB2 Text Search エンジンを使用して検索を実行します。検索される列は XML 列で、この列にはテキスト検索索引があります。

最初の関数は、PRODUCT.DESCRPTION 列に保管されている XML 文書の中から snow と shovel という単語を含む XML 文書を検索します。関数では、返す文書の最大数が 2 に設定されています。テキスト検索が大量の文書を返す場合は、**RESULTLIMIT** オプションを使用して返す文書の最大数を制限することにより、検索を最適化することができます。

```
db2-fn:xmlcolumn-contains('PRODUCT.DESCRPTION', 'snow shovel', 'RESULTLIMIT=2')
```

関数は、検索基準と一致する XML 文書を返します。文書には、単なる製品説明以上のものが含まれることがあります。例えば、以下は XML 文書の一部ですが、この部分は 1 つの XML 列からの 2 つの製品説明で成っています。各文書には、製品説明に加えて製品の名前、価格、重量、および製品 ID などの情報が含まれています。

```
<product xmlns="http://posample.org" pid="100-100-01">
  <description>
    <name>Snow Shovel, Basic 22 inch</name>
    <details>Basic Snow Shovel, 22 inches wide, straight handle with
      D-Grip</details>
    <price>9.99</price>
    <weight>1 kg</weight>
  </description>
</product>
<product xmlns="http://posample.org" pid="100-101-01">
  <description>
    <name>Snow Shovel, Deluxe 24 inch</name>
    <details>A Deluxe Snow Shovel, 24 inches wide, ergonomic curved handle
      with D-Grip</details>
    <price>19.99</price>
    <weight>2 kg</weight>
  </description>
</product>
```

次の関数では、XML 列 STUDENT\_ESSAYS.ABSTRACTS から、combustible fossil (スペイン語で化石燃料の意) という句を含む 10 人の生徒の小論文を検索します。この関数では、テキスト検索に使用する言語として es\_ES (スペインで話されるスペイン語) を指定し、関連するテキスト検索索引用に作成された同義語ディクショナリーを使用します。この関数では、**RESULTLIMIT** を使用して結果の数を制限することによって、検索を最適化します。

```
db2-fn:xmlcolumn-contains('STUDENT_ESSAYS.ABSTRACTS', 'combustible fossil',
  'QUERYLANGUAGE=es_ES RESULTLIMIT=10 SYNONYM=ON')
```

次の例では、db2-fn:xmlcolumn-contains を使用して、PRODUCT.DESCRPTION 列に保管されている XML 文書の中から ergonomic という単語を含む XML 文書を検索します。この式では、価格が 20 より低い製品の名前を返します。

```
xquery
declare default element namespace "http://posample.org";
db2-fn:xmlcolumn-contains(
  'PRODUCT.DESCRPTION', 'ergonomic')/product/description[price < 20]/name
```

上の式では、返される XML 文書から名前前のエレメントだけが返されます。例えば、ergonomic という語が「Snow Shovel, Deluxe 24 inch」という製品の製品説明に含まれている場合、式はおおよそ以下のようなエレメント名を返します。

```
<name xmlns="http://posample.org" >Snow Shovel, Deluxe 24 inch</name>
```

次の式では、db2-fn:xmlcolumn-contains を使用して、PRODUCT.DESCRPTION 列から ice と scraper という言葉を含む XML 文書を検出します。式では、製品説明

の製品 ID を使用して、PURCHASEORDER 表からその製品 ID を含む購入オーダーを検出します。この式は、検索に一致する XML 記述文書中のその製品 ID を含む購入オーダーから、顧客 ID を返します。

```
xquery
declare default element namespace "http://posample.org";
for $po in db2-fn:sqlquery('
  select XMLElement(Name "po", XMLElement(Name "custid", purchaseorder.custid),
    XMLElement(Name "porder", purchaseorder.porder))
  from purchaseorder')
let $product := db2-fn:xmlcolumn-contains('PRODUCT.DESCRPTION',
  'ice scraper')/product
where $product/@pid = $po/porder/PurchaseOrder/item/partid
order by $po/custid
return $po/custid
```

式は顧客 ID を含む custid 要素を返します。要素は昇順で返されます。例えば、一致する購入オーダーが 3 件あり、それらの購入オーダーに顧客 ID 1001、1002、および 1003 が含まれていた場合、式は以下の要素を返します。

```
<custid xmlns="http://posample.org">1001</custid>
<custid xmlns="http://posample.org">1002</custid>
<custid xmlns="http://posample.org">1003</custid>
```

同じ照会内に複数のテキスト検索がある場合、DB2 Text Search エンジンでは複数のテキスト検索の結果を結合して返します。例えば、次の SELECT ステートメントでは、ruby on rails および ajax web と完全に一致する句を含む従業員の履歴書を検索します。このステートメントの WHERE 文節には 2 つのテキスト検索が含まれています。それぞれのテキスト検索が最大 10 件の結果を返し、それぞれのテキスト検索は異なる検索引数を使用して従業員の履歴書を検索します。このステートメントでは、両方の句を含む従業員の履歴書が 10 件よりも多く検出されていても、返される従業員 ID の数が 10 件に満たないこともあります。

```
SELECT EMPNO FROM EMP_RESUME
WHERE XMLEXISTS('db2-fn:xmlcolumn-contains(''EMP_RESUME.XML_FORMAT'',
  ''"ruby on rails"', 'RESULTLIMIT=10''))
AND XMLEXISTS('db2-fn:xmlcolumn-contains(''EMP_RESUME.XML_FORMAT'',
  ''"ajax web"', 'RESULTLIMIT=10''))
```

上のステートメントの場合、DB2 Text Search は、それぞれのテキスト検索ごとに最大 10 行を返します。ただし、返された行の履歴書に一方の句しか含まれていない (つまり両方の句が含まれていない) 場合は、従業員 ID は返されません。

SELECT ステートメントを変える 1 つの方法として、WHERE 文節にある 2 つのテキスト検索を結合して 1 つのテキスト検索にすることができます。次のステートメントでは、1 つのテキスト検索を使用して、履歴書に ruby on rails と ajax web の両方の句が含まれる従業員の ID を返します。

```
SELECT EMPNO FROM EMP_RESUME
WHERE XMLEXISTS('db2-fn:xmlcolumn-contains(''EMP_RESUME.XML_FORMAT'',
  ''"ruby on rails" AND "ajax web"', 'RESULTLIMIT=10''))
```

XQuery の属性のコロンをエスケープするには、次のように単一の円記号を使用します。

```
xquery for $i in db2-fn:xmlcolumn-contains('DBCP1208.T_AUTO.T_XML',
  '@xpath:'//en//en[. contains("purpose") and @a1 = "value for en¥:attribute1"
  and @slope = "9"] '' ') return $i/*/*fn:string
```



---

## 第 10 章 DB2 Text Search 用の管理コマンド

DB2 Text Search をインスタンス、データベース、表、およびテキスト索引の各レベルで管理するために使用できるいくつかのコマンドがあります。それらのコマンドはすべて、db2ts を使用して実行します。

以下のインスタンス・レベルの管理コマンドを使用して、DB2 Text Search インスタンスの開始と停止、および有用でなくなったテキスト検索索引のクリーンアップを行います。

### **db2ts START FOR TEXT**

DB2 Text Search インスタンス・サービスを開始します。

### **db2ts STOP FOR TEXT**

DB2 Text Search インスタンス・サービスを停止します。

### **db2ts CLEANUP FOR TEXT**

有用でなくなったテキスト検索コレクションをクリーンアップします。

以下のデータベース・レベルの管理コマンドを使用して、データベースで DB2 Text Search をセットアップする、または無効にします。また、コマンド・ロックを消去します。

### **db2ts ENABLE DATABASE FOR TEXT**

現行データベースでのテキスト検索索引の作成、管理、および使用を有効にします。

### **db2ts DISABLE DATABASE FOR TEXT**

データベースに対する DB2 Text Search を無効にし、いくつかのテキスト検索カタログ表およびビューをドロップします。

### **db2ts CLEAR COMMAND LOCKS**

データベース内のすべての索引について、コマンド・ロックを削除します。

以下の表レベルおよび索引レベルのコマンドを使用して、表の列に対するテキスト検索索引を作成および操作します。

### **db2ts CREATE INDEX**

テキスト検索索引を作成します。

### **db2ts DROP INDEX**

テキスト列に関連したテキスト検索索引をドロップします。

### **db2ts ALTER INDEX**

テキスト検索索引の特性を変更します。

### **db2ts UPDATE INDEX**

テキスト列の現在の内容に基づいて、テキスト検索索引にデータを設定するか、テキスト検索索引を更新します。

### **db2ts CLEAR EVENTS FOR TEXT**

索引付けの状況およびエラーに関する情報を提供するイベント・ビューである SYSIBMTS.TSEVENT ビューからイベントを削除します。

#### db2ts CLEAR COMMAND LOCKS FOR INDEX

特定のテキスト検索索引について、すべてのコマンド・ロックを削除します。

#### db2ts RESET PENDING FOR TABLE

テキスト検索のために保守されているすべての従属表を特定し、必要に応じて SET INTEGRITY を実行します。

#### db2ts HELP

db2ts コマンド・オプションのリスト、および特定のエラー・メッセージに関する情報を表示します。

---

## db2ts ALTER INDEX

このコマンドは、索引の更新特性を変更します。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

### 許可

ステートメントの許可 ID が保持する特権には、SYSTS\_MGR ロールと以下の少なくとも 1 つの権限が含まれている必要があります。

- DBADM 権限
- 基本スキーマに対する ALTERIN 特権
- そのテキスト検索索引が定義されている基本表に対する CONTROL または ALTER 特権

既存のスケジュールを変更する場合、許可 ID は索引作成者と同じであるか、または DBADM 権限を持っている必要があります。

### 必要な接続

データベース

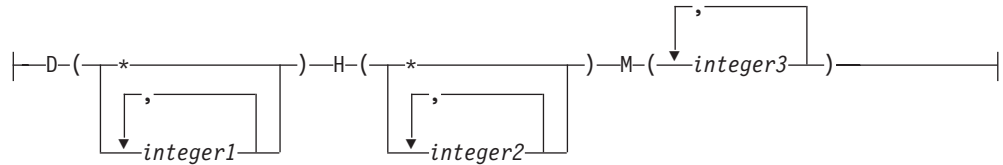
### コマンド構文

```
▶▶ALTER INDEX—index-name—FOR TEXT—| update characteristics |—| options |—▶▶  
▶| connection options |—————▶▶
```

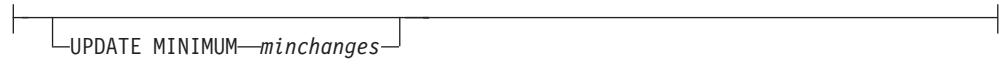
#### update characteristics:

```
|—————▶▶  
|—| UPDATE FREQUENCY—| NONE—|  
|—| update frequency |—|  
▶| incremental update characteristics |—————▶▶
```

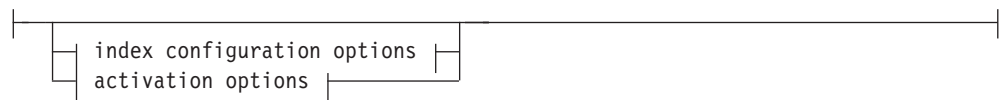
#### update frequency:



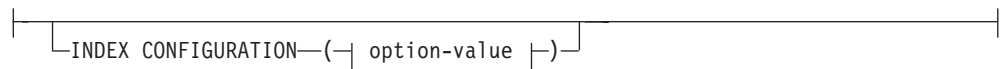
### incremental update characteristics:



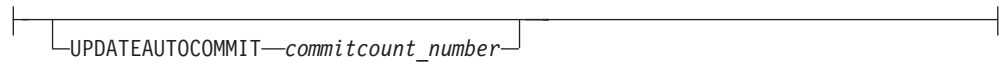
### オプション:



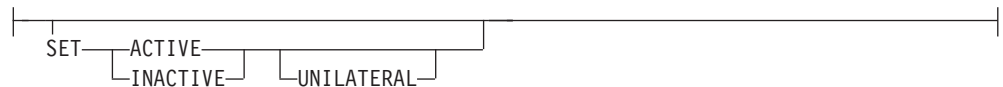
### index configuration options:



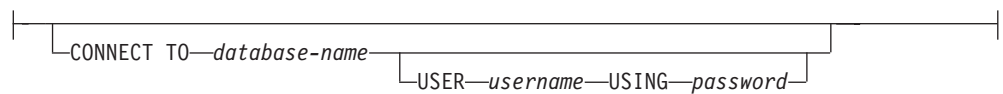
### option-value:



### activation options:



### connection options:



## コマンド・パラメーター

### ALTER INDEX *index-name*

**CREATE INDEX** コマンドで指定されている索引のスキーマおよび名前。データベース内のテキスト検索索引を固有に識別します。

### UPDATE FREQUENCY

索引の更新が行われる頻度を指定します。変更の数が少なくとも **UPDATE MINIMUM** に設定された値である場合に、索引が更新されます。更新頻度 **NONE** は、索引の更新はそれ以降行われないことを意味します。これは、変更されない

データを持つ表のテキスト列に役立ちます。さらに、これはユーザーが (**UPDATE INDEX** コマンドを使用して) 手動で索引を更新する予定の場合にも役立ちます。**START FOR TEXT** コマンドが実行されていて、DB2 Text Search インスタンス・サービスが実行中の場合にのみ、自動更新が行われます。

デフォルトの頻度値は、ビュー **SYSIBMTS.TSDEFAULTS** の **DEFAULTNAME='UPDATEFREQUENCY'** のところから取得されます。

#### **NONE**

自動更新は、テキスト索引に適用されません。以降のすべての索引更新は手動で開始する必要があります。

#### **D** 索引が更新される曜日。

\* 毎日。

*integer1*

日曜日から土曜日の特定の曜日: 0 から 6

#### **H** 索引が更新される特定の日の時。

\* 毎時。

*integer2*

午前 0 時から午後 11 時までの特定の時: 0 から 23

#### **M** 索引が更新される特定の時刻の分。

*integer3*

正時 (0) として、または正時の後に 5 分の増分の倍数として指定:  
0、5、10、15、20、25、30、35、40、45、50、または 55

**UPDATE FREQUENCY** オプションを指定しない場合、頻度の設定は変更されないままとなります。

#### **UPDATE MINIMUM** *minchanges*

索引が増分的に更新される前に生じなければならない、テキスト文書への変更の最小数を指定します。同じテキスト文書への複数の変更は、別個の変更として扱われます。**UPDATE MINIMUM** オプションを指定しない場合、設定は変更されないままとなります。

#### **INDEX CONFIGURATION** (*option-value*)

バージョン 9.7 フィックスパック 3 以降のフィックスパックでは、これは、テキスト索引構成設定を変更できる **VARCHAR(32K)** タイプのオプションの入力引数です。以下のオプションがサポートされています。

表9. オプション値の仕様

オプション	値	データ・タイプ	説明
SERIALUPDATE	<i>updatemode</i>	整数	パーティション化されたテキスト検索索引の更新処理を並列モードと直列モードのどちらで行うのかを指定します。並列モードの場合、実行が複数のデータベース・パーティションに分散され、それぞれのノードで独立して行われます。直列モードでは、実行は分散されずに行われ、障害が検出されると停止します。通常、直列モード実行の方が時間はかかりますが、必要とするリソースがかなり少なく済みます。 <ul style="list-style-type: none"> <li>• 0 = 並列モード</li> <li>• 1 = 直列モード</li> </ul>
UPDATEAUTOCOMMIT	<i>commitcount_number</i>	整数	ここで指定された回数の索引更新が行われると、コミットが実行され、イニシャル更新またはインクリメンタル更新のためにそれまでの作業が自動的に保存されます。 <ul style="list-style-type: none"> <li>• イニシャル更新では、データ更新をキャプチャーするトリガーがアクティブ化された後の基本表からの文書のバッチを、索引更新が処理します。文書更新の回数が COMMITCOUNT に達すると、サーバーは暫定コミットを実行します。処理されていない文書により生成されたログ項目は、ステージング表から除去されます。イニシャル・テキスト索引更新に UPDATEAUTOCOMMIT オプションを使用すると、実行時間が大幅に増加します。</li> <li>• インクリメンタル更新では、処理されているログ項目は、各暫定コミットで対応するステージング表から除去されます。COMMITCOUNT は更新される文書の数であり、ステージング表項目の数ではありません。</li> </ul>

*activation options*

バージョン 9.7 フィックスパック 3 以降のフィックスパックでは、この整数タイプの入力引数はテキスト索引の状況を設定します。

**ACTIVE**

テキスト索引状況をアクティブに設定します

**INACTIVE**

テキスト索引状況を非アクティブに設定します

**UNILATERAL**

DB2 Text Search 索引の状況にのみ影響する UNILATERAL 変更を指定します。この引数が指定された場合、DB2 Text Search 索引の状況だけがアクティブまたは非アクティブに変更されます。UNILATERAL 引数がない場合、DB2 Text Search 索引および DB2 Net Search Extender 索引のアクティブ化の状況はともに切り替えられ、テキスト索引の一方だけがアクティブ化されます。

### CONNECT TO database-name

この節は、接続が確立される先のデータベースを指定します。データベースはローカル・システム上になければなりません。この節が指定されている場合は、環境変数 DB2DBDFT よりも優先されます。以下のすべてが該当する場合、この節は省略できます。

- DB2DBDFT 環境変数が有効なデータベース名に設定されている。
- コマンドを実行しているユーザーが、データベース・サーバーに接続するのに必要な権限を持っている。

### USER username USING password

この節は、接続を確立するのに使用するユーザー名およびパスワードを指定します。

## 使用上の注意

DB2 のデータベース・オブジェクトおよび照会に適用されるすべての制限および命名規則は、DB2 Text Search のフィーチャーおよび照会にも適用されます。DB2 Text Search 関連の ID は、DB2 の命名規則に準拠している必要があります。さらに、追加の制限事項がいくつかあります。例えば、これらの ID は以下の形式のみになります。

[A-Za-z][A-Za-z0-9@#\$\_]\*

または

"[A-Za-z ][A-Za-z0-9@#\$\_ ]\*"

複数のコマンドが競合する場合、テキスト検索索引でそれらを並行して実行することができないことに注意してください。競合するコマンドの実行中に、このコマンドが発行される場合、エラーが発生してコマンドは失敗します。その後、コマンドの実行を再試行することができます。以下に、競合するコマンドをいくつか示します。

- ALTER INDEX
- CLEAR EVENTS FOR INDEX
- DROP INDEX
- UPDATE INDEX
- DISABLE DATABASE FOR TEXT

データベースへの変更: DB2 Text Search のカタログ情報を更新します。

索引をアクティブにした結果は、元の索引の状況により異なります。次の表は、結果を示したものです。

表 10. 無効索引無しの状況変更:

イニシャルの DB2 Text Search または Net Search Extender の状況	ACTIVE 要求	ACTIVE UNILATERAL 要求	INACTIVE 要求	INACTIVE UNILATERAL 要求
アクティブ / 非アクティブ	変更なし	変更なし	非アクティブ / アクティブ	非アクティブ / 非アクティブ

表 10. 無効索引無しの場合の変更: (続き)

イニシャルの DB2 Text Search または Net Search Extender の状況	ACTIVE 要求	ACTIVE UNILATERAL 要求	INACTIVE 要求	INACTIVE UNILATERAL 要求
非アクティブ / アクティブ	アクティブ / 非アクティブ	エラー	変更なし	変更なし
非アクティブ / 非アクティブ	アクティブ / 非アクティブ	アクティブ / 非アクティブ	非アクティブ / アクティブ	変更なし

SQL20427N および CIE0379E エラー・メッセージが、索引のアクティブ化競合の場合に戻されます。

## db2ts CLEANUP FOR TEXT

インスタンス内またはデータベース内の DB2 Text Search コレクションをクリーンアップします。

データベースに対してクリーンアップ操作を実行すると、無効なテキスト索引とそれらの関連コレクションがドロップされます。インスタンスに対してクリーンアップ操作を実行すると、廃止コレクションが削除されます。テキスト検索索引が作成されたデータベースにおいて、DB2 Text Search が無効化される前にデータベースがドロップされると、コレクションは廃止される可能性があります。

**注:** コマンドの操作対象はテキスト検索索引ですが、テキスト検索サーバーのツールの操作対象はテキスト検索コレクションです。テキスト検索コレクションとは、テキスト検索索引の基礎となる表現を指します。テキスト検索索引とその関連コレクションの関係は、非パーティション・セットアップの場合は 1:1 で、パーティション・セットアップの場合は 1:n です (n はデータ・パーティションの数)。テキスト検索索引のテキスト検索コレクションを調べるには、SYSIBMTS.TSCOLLECTIONNAMES カタログ表を照会してください。追加情報については、DB2 Text Search の管理ツールに関するトピックを参照してください。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

### 許可

このコマンドをインスタンス・レベルで実行するためには、テキスト検索サーバー・プロセスの所有者でなければなりません。統合型テキスト検索サーバーの場合は、インスタンス所有者であることが必要です。

このコマンドをデータベース・レベルで実行するためには、ステートメントの許可 ID が持つ特権に、SYSTS\_ADM ロールおよび DBADM 権限が含まれている必要があります。

### 必要な接続

このコマンドは DB2 データベース・サーバーから発行されなければなりません。

## コマンド構文

Instance level

▶▶—CLEANUP FOR TEXT—▶▶

Database level

▶▶—CLEANUP FOR TEXT—connection-options—▶▶

## コマンド・パラメーター

なし

---

## db2ts CLEAR COMMAND LOCKS

特定のテキスト検索索引について、またはデータベース内のすべてのテキスト検索索引について、コマンド・ロックをすべて除去します。コマンド・ロックはテキスト検索索引コマンドの開始時に作成され、コマンドが完了すると破棄されます。それにより、異なるコマンド間の望ましくない競合が避けられます。

このコマンドの使用は、予期しないシステム動作のためにロックがそのまま残ってしまい、明示的にクリーンアップされる必要があるというまれなケースで、必要になります。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

### 許可

索引のロックをクリアするために使用されるステートメントの許可 ID が保持する特権には、以下の両方の権限が含まれている必要があります。

- SYSTS\_MGR ロール
- DBADM 権限、または索引が定義されている基本表に対する CONTROL 特権

データベース接続のロックをクリアするために使用されるステートメントの許可 ID が保持する特権には、SYSTS\_ADM ロールが含まれている必要があります。

### 必要な接続

データベース

## コマンド構文

▶▶—CLEAR COMMAND LOCKS—  
└─FOR INDEX—*index-name*—┘—FOR TEXT—▶▶

▶▶—connection options—▶▶

**connection options:**



```
CONNECT TO database-name
          USER username USING password
```

## コマンド・パラメーター

**FOR INDEX** *index-name*

**CREATE INDEX** コマンドで指定されている索引の名前。

**CONNECT TO** *database-name*

この節は、接続が確立される先のデータベースを指定します。データベースはローカル・システム上になければなりません。この節が指定されている場合は、環境変数 **DB2DBDFT** よりも優先されます。以下のすべてが該当する場合、この節は省略できます。

- **DB2DBDFT** 環境変数が有効なデータベース名に設定されている。
- コマンドを実行しているユーザーが、データベース・サーバーに接続するのに必要な権限を持っている。

**USER** *username* **USING** *password*

この節は、接続を確立するのに使用する許可名およびパスワードを指定します。

## 使用上の注意

コマンド・ロックを所有している処理が非活動であるために、このコマンドを呼び出します。このケースでは、(ロックにより表される) コマンドが完了していない場合や、索引が操作できない場合があります。適切なアクションを取る必要があります。例えば、**DROP INDEX** コマンドを実行している処理が突然停止したとします。いくつかの索引データは削除されましたが、すべてのカタログおよびコレクション情報が削除された訳ではありません。コマンド・ロックは、そのまま残されています。**DROP INDEX** コマンド・ロックをクリアした後、**DROP INDEX** コマンドを再実行することができます。別の例として、**UPDATE INDEX** コマンドを実行しているプロセスが中断したとします。一部の文書の処理は完了しましたが、全部ではありません。また、コマンドのロックがまだ掛かっています。テキスト検索索引の状況を確認し、**UPDATE INDEX** コマンドのロックを解除した後、**UPDATE INDEX** コマンドを再実行できます。

このコマンドが発行されるとき、DB2 Text Search ビュー SYSIBMTS.TSLOCKS の内容が更新されます。

---

## db2ts CLEAR EVENTS FOR TEXT

このコマンドは、管理用に使用されている索引のイベント表から索引付けイベントを削除します。この表の名前は、ビュー SYSIBMTS.TSINDEXES の列 EVENTVIEWNAME にあります。

少なくとも 1 つの文書を処理する索引の更新操作が行われるごとに、通知項目と、場合によってはエラー項目がイベント表に生成されます。自動更新の場合、この表を定期的に検査する必要があります。文書の固有のエラーは訂正されなければなりません (文書内容を変更することにより)。エラーを訂正した後、イベントをクリアすることができます (スペースを消費しすぎないために、クリアすべきです)。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

## 許可

このステートメントの許可 ID には、以下の両方の特権が含まれている必要があります。

- SYSTS\_MGR ロール
- 索引が定義されている表に対する DATAACCESS 権限または CONTROL 特権を持つ DBADM

## 必要な接続

データベース

## コマンド構文

```
▶▶ CLEAR EVENTS FOR INDEX—index-name—FOR TEXT—| connection options |▶▶▶▶
```

### connection options:

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

## コマンド・パラメーター

*index-name*

**CREATE INDEX** コマンドで指定されている索引の名前。索引名は、DB2 索引の命名上の制約に従う必要があります。

**CONNECT TO** *database-name*

この節は、接続が確立される先のデータベースを指定します。データベースはローカル・システム上になければなりません。この節が指定されている場合は、環境変数 DB2DBDFT よりも優先されます。以下のすべてが該当する場合、この節は省略できます。

- DB2DBDFT 環境変数が有効なデータベース名に設定されている。
- コマンドを実行しているユーザーが、データベース・サーバーに接続するのに必要な権限を持っている。

**USER** *username* **USING** *password*

この節は、接続を確立するのに使用する許可名およびパスワードを指定します。

## 使用上の注意

DB2 のデータベース・オブジェクトおよび照会に適用されるすべての制限および命名規則は、DB2 Text Search のフィーチャーおよび照会にも適用されます。DB2 Text Search 関連の ID は、DB2 の命名規則に準拠している必要があります。さらに、追加の制限事項がいくつかあります。例えば、これらの ID は以下の形式のみになります。

```
[A-Za-z][A-Za-z0-9@#$_]*
```

または

```
"[A-Za-z ][A-Za-z0-9@#$_]*"
```

通常の更新がスケジュールされている場合 (**CREATE INDEX** または **ALTER INDEX** コマンドの **UPDATE FREQUENCY** オプションを参照)、イベント表を定期的にチェックする必要があります。テキスト検索索引用の DB2 Text Search のイベント表をクリーンアップするには、そのイベントの理由をチェックし、エラーの原因を除去した後に、**CLEAR EVENTS FOR INDEX** コマンドを使用します。

イベント表で参照されているすべての行に、必ず変更を行ってください。ユーザー表の行を変更することにより、次の **UPDATE INDEX** の試行が、以前にエラーのあった文書を確実に正常に再索引付けできるようにします。

複数のコマンドが競合する場合、テキスト検索索引でそれらを並行して実行することができないことに注意してください。競合するコマンドの実行中に、このコマンドが発行される場合、エラーが発生してコマンドは失敗します。その後、コマンドの実行を再試行することができます。以下に、競合するコマンドをいくつか示します。

- **CLEAR EVENTS FOR INDEX**
- **UPDATE INDEX**
- **ALTER INDEX**
- **DROP INDEX**
- **DISABLE DATABASE FOR TEXT**

データベースへの変更: イベント表がクリアされます。

---

## db2ts CREATE INDEX

**db2ts CREATE INDEX** コマンドは、テキスト列のテキスト検索索引を作成します。その後、テキスト検索関数を使用して、列データを検索することができます。

テキスト検索の **UPDATE INDEX** コマンドを実行するか、または索引に定義された更新頻度に従って DB2 管理タスク・スケジューラーが **UPDATE INDEX** コマンドを実行するまで、テキスト検索索引の中にデータが入ることはありません。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

### 許可

ステートメントの許可 ID には、**SYSTS\_MGR** ロール、データベースの **CREATETAB** 権限、および以下のいずれかの権限または特権が含まれている必要があります。

- その索引の定義されている表に対する **CONTROL** 特権。
- 索引が定義される表に対する **INDEX** 特権と、以下の権限のいずれか。
  - データベースに対する **IMPLICIT\_SCHEMA** 権限 (索引の暗黙または明示のスキーマ名が存在しない場合)
  - スキーマに対する **CREATEIN** 特権 (索引のスキーマ名が存在する場合)
- **DBADM** 権限

## 必要な接続

データベース

## コマンド構文

```
▶▶ CREATE INDEX index-name FOR TEXT ON schema-name table-name
▶ ( text-column-name )
  ( function-name ( text-column-name ) )
▶ | text default information | | update characteristics |
▶ | storage options | | index configuration options | | partition options |
▶ | connection options |
```

### text default information:

```
| CODEPAGE code-page | LANGUAGE locale | FORMAT format |
```

### update characteristics:

```
| UPDATE FREQUENCY NONE
| | update frequency
▶ | incremental update characteristics |
```

### update frequency:

```
| D ( * ) H ( * ) M ( integer3 )
| | integer1 | | integer2 |
| | | | | |
```

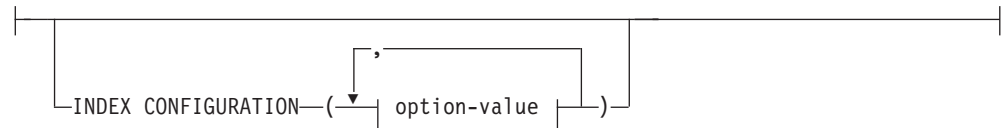
### incremental update characteristics:

```
| UPDATE MINIMUM minchanges |
```

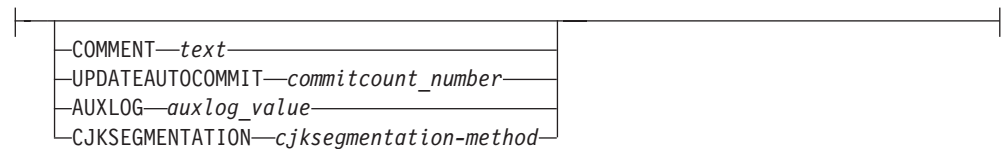
### storage options:

```
| COLLECTION DIRECTORY directory
▶ | ADMINISTRATION TABLES IN tablespace-name |
```

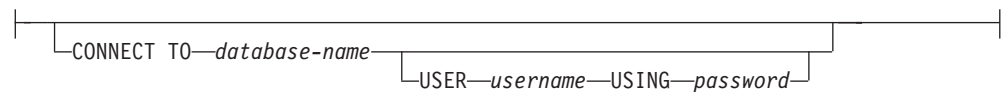
### index configuration options:



### option-value:



### connection options:



## コマンド・パラメーター

### CREATE INDEX *index-name*

作成される索引の名前を指定します (スキーマ修飾はオプション)。その名前はデータベース内のテキスト検索索引を固有に識別します。索引名は、DB2 索引の命名上の制約に従う必要があります。

### ON *table-name*

テキスト列を含む表名。テキスト検索索引を、以下の表に作成することはできません。

- フェデレーテッド表
- マテリアライズ照会表
- ビュー

### *text-column-name*

索引を付ける列の列名。列は、

CHAR、VARCHAR、CLOB、DBCLOB、BLOB、GRAPHIC、

VARGRAPHIC、または XML のデータ・タイプのいずれかでなければなりません。列のデータ・タイプがこれらのいずれのデータ・タイプでもない場合、

*function-schema.function-name* で指定された変換関数を使用して、列タイプを有効なタイプのいずれかに変換します。あるいは、索引付けされるテキスト文書にアクセスするユーザー定義の外部関数を指定することができます。1 つのテキスト検索索引のみが、列に作成されます。

### *function-name* (*text-column-name*)

DB2 命名規則に準拠して、テキスト検索でサポートされるタイプではない列内のテキスト文書にアクセスする外部スカラー関数のスキーマ修飾名を指定します。その値のデータ・タイプ変換を実行し、テキスト検索でサポートされるデー

タ・タイプの 1 つとして値を戻します。このタスクは、列タイプの変換を実行します。この関数は、1 つのパラメーターのみを取り、1 つの値のみを戻す必要があります。

#### **CODEPAGE** *code-page*

テキスト文書を索引付けするときに使用される DB2 コード・ページ (CODEPAGE) を指定します。デフォルト値は、ビュー SYSIBMTS.TSDEFAULTS の DEFAULTNAME='CODEPAGE' のところの値で指定されます (これがデータベース・コード・ページである場合もあります)。この引数はバイナリー・データ・タイプにのみ適用されます。例えば列タイプまたは変換関数からの戻りの型は、BLOB または文字タイプの FOR BIT DATA でなければなりません。

#### **LANGUAGE** *locale*

索引付け中に、文書の言語固有の処理のために DB2 Text Search によって使用される言語を指定します。ロケールを指定しない場合は、データベース・ Territory が **LANGUAGE** のデフォルト設定を判別します。ロケールを判別するために、文書が自動的にスキャンされるようにするには、*locale* を **AUTO** に指定してください。

#### **FORMAT** *format*

列内のテキスト文書のフォーマットを指定します。サポートされているフォーマットには TEXT、XML、HTML、および INSO が含まれます。文書を索引付けするときに、DB2 Text Search はこの情報を必要とします。フォーマットを指定しない場合、デフォルト値が使用されます。デフォルト値は、ビュー SYSIBMTS.TSDEFAULTS の DEFAULTNAME='FORMAT' のところの値です。データ・タイプ XML の列の場合は、DEFAULTNAME の値に関係なく、デフォルトのフォーマット 'XML' が使用されます。INSO フォーマットを使用するためには、リッチ・テキスト・サポートがインストールされている必要があります。

#### **UPDATE FREQUENCY**

索引の更新が行われる頻度を指定します。変更の数が少なくとも **UPDATE MINIMUM** に設定された値である場合に、索引が更新されます。更新頻度 **NONE** は、索引の更新はそれ以降行われなことを意味します。**NONE** の更新頻度は、変更されないデータを持つ表のテキスト列に役立ちます。さらに、これはユーザーが (**UPDATE INDEX** コマンドを使用して) 手動で索引を更新する予定の場合にも役立ちます。**START FOR TEXT** コマンドが実行されていて、DB2 Text Search インスタンス・サービスが実行中の場合にのみ、自動更新が行われます。

デフォルトの頻度値は、ビュー SYSIBMTS.TSDEFAULTS の DEFAULTNAME='UPDATEFREQUENCY' のところから取得されます。

#### **NONE**

これ以上の索引更新は行われません。更新を手動で開始する必要があります。

#### **D** 索引が更新される曜日。

\* 毎日。

*integer1*

日曜日から土曜日の特定の曜日: 0 から 6

**H** 索引が更新される指定日の時。

\* 毎時。

*integer2*

午前 0 時から午後 11 時までの特定の時: 0 から 23

**M** 索引が更新される指定時の分。

*integer3*

正時 (0) として、または正時の後に 5 分の増分の倍数として指定:

0、5、10、15、20、25、30、35、40、45、50、または 55

#### **UPDATE MINIMUM** *minchanges*

**UPDATE FREQUENCY** で指定された時間に索引が増分的に更新される前に、テキスト文書に加えらるる変更の最小数を指定します。正整数値だけを指定できます。デフォルト値はビュー `SYSIBMTS.TSDEFAULTS` から取られます。ここで、`DEFAULTNAME='UPDATEMINIMUM'` です。

注: この値は、**UPDATE INDEX** コマンドの実行中は無視されます (**USING UPDATE MINIMUM** オプションがそこで使用されていない場合)。小さい値は、表列とテキスト検索索引の間の整合性を増します。ただし、これによってシステムに余分の負荷がかかることにもなります。

#### **COLLECTION DIRECTORY** *directory*

テキスト検索索引コレクションを保管するディレクトリーを指定します。絶対パスを指定する必要があります。絶対パス名の最大長は 215 文字です。テキスト検索サーバーのインスタンス・サービスのプロセス所有者は、このディレクトリーに対する読み取りおよび書き込み権限を持っている必要があります。

**COLLECTION DIRECTORY** パラメーターは、統合テキスト検索サーバーのセットアップでのみサポートされます。コレクションの場所に関する追加情報については、使用上の注意を参照してください。

#### **ADMINISTRATION TABLES IN** *tablespace-name*

索引用に作成される管理表のための、**TEMPORARY** 表スペース以外の既存の表スペースの名前を指定します。

非パーティション・データベースでは、表スペースが指定されていない場合は、索引を作成する基本表の表スペースが使用されます。

パーティション・データベースでは、テキスト索引を作成する際には、**ADMINISTRATION TABLES** パラメーターを明示的に使用する必要があります。この表スペースは、テキスト検索索引のステージング表が対応する基本表と同じ方法で分散されるようにするには、基本表の表スペースと同じパーティション・グループ内になければなりません。

#### **INDEX CONFIGURATION** (*option-value*)

追加の索引関連値をオプション値のストリングのペアとして指定します。以下の値がサポートされています。

表 11. オプション値の仕様

オプション	値	データ・タイプ	説明
<b>COMMENT</b>	<i>text</i>	512 バイトより少ないストリング値	DB2 Text Search カタログ・ビュー TSINDEXES にある REMARKS 列にストリング・コメント値を追加します。ストリング・コメント値をコレクションの説明としても追加します。
<b>UPDATEAUTOCOMMIT</b>	<i>commitcount_number</i>	整数	このオプションは索引更新の回数を指定するもので、その回数に達すると、コミットが実行され、イニシャル更新またはインクリメンタル更新のためにそれまでの作業が自動的に保存されます。 <ul style="list-style-type: none"> <li>イニシャル更新では、データ更新をキャプチャーするトリガーがアクティブ化された後の基本表からの文書のバッチを、索引更新が処理します。文書更新の回数が COMMITCOUNT の数に達すると、サーバーは暫定コミットを実行します。処理されていない文書により生成されたログ項目は、ステージング表から除去されます。イニシャル・テキスト索引更新に <b>UPDATEAUTOCOMMIT</b> オプションを使用すると、実行時間が大幅に増加します。</li> <li>インクリメンタル更新では、処理されているログ項目は、各暫定コミットで対応するステージング表から除去されます。COMMITCOUNT は更新される文書の数であり、ステージング表項目の数ではありません。</li> </ul>
<b>AUXLOG</b>	<i>auxlog_value</i>	ストリング	トリガーに認識されない変更をキャプチャーするための、追加のログ・インフラストラクチャーの作成を制御します。範囲パーティション表のデフォルト設定は ON であり、それ以外の場合は OFF になります。非範囲パーティション表の場合は AuxLogNorm、範囲パーティション表の場合は AuxLogPart の設定によって、デフォルト表におけるデフォルト値を変更できます。
<b>CJKSEGMENTATION</b>	<i>CJKSEGMENTATION_method</i>	512 バイトより少ないストリング値	中国語、日本語、および韓国語 (zh_CN, zh_TW, ja_JP, ko_KR ロケール・セット) の文書のセグメンテーション方式を指定します。自動言語検出が有効 (LANGUAGE AUTO) になっている場合のこれらの文書も対象になります。オプションを指定しない場合、デフォルト表の <b>CJKSEGMENTATION</b> の値が適用されます。サポートされる値は、以下のとおりです。 <ul style="list-style-type: none"> <li>MORPHOLOGICAL</li> <li>NGRAM</li> </ul> 指定したセグメンテーション方式は SYSIBMTS.TSCONFIGURATION 管理ビューに追加されます。テキスト索引が作成された後に変更することはできません。



**要確認:** コメントのような非数値は、単一引用符で囲む必要があります。ストリング値内の単一引用符の文字は、2 つの連続した単一引用符によって表記される必要があります。

**例:** INDEX CONFIGURATION (COMMENT 'Index on User''s Guide column')

## パーティション・オプション

IBM の内部使用のために予約済みです。

### CONNECT TO *database-name*

この節は、接続が確立される先のデータベースを指定します。データベースはローカル・システム上になければなりません。この節が指定されている場合は、環境変数 **DB2DBDFT** よりも優先されます。以下のすべてが該当する場合、この節は省略できます。

- **DB2DBDFT** 環境変数が有効なデータベース名に設定されている。
- コマンドを実行しているユーザーが、データベース・サーバーに接続するのに必要な権限を持っている。

### USER *username* USING *password*

この節は、接続を確立するのに使用する許可名およびパスワードを指定します。

## 使用上の注意

DB2 のデータベース・オブジェクトおよび照会に適用されるすべての制限および命名規則は、DB2 Text Search のフィーチャーおよび照会にも適用されます。DB2 テキスト関連の ID は、DB2 の命名規則に準拠している必要があります。さらに、追加の制限事項がいくつかあります。例えば、これらの ID は以下の形式のみになります。

```
[A-Za-z][A-Za-z0-9@#$_]*
```

または

```
"[A-Za-z ][A-Za-z0-9@#$_ ]*"
```

**CREATE INDEX** コマンドが正常に実行されると、以下のようになります。

- DB2 Text Search のサーバー・データが更新されます。以下の例のように、*instance\_database-name\_index-identifier\_number* という名前のコレクションがデータベース・パーティションごとに作成されます。

```
tigertail_MYTSDB_TS250517_0000
```

コレクション名は SYSIBMTS.TSCOLLECTIONNAMES ビュー (列 COLLECTIONNAME) から取られます。

- DB2 Text Search のカタログ情報が更新されます。索引のステージング表が、適切な DB2 索引のある指定された表スペースに作成されます。さらに、索引のイベント表が指定された表スペースに作成されます。auxlog on オプションが設定されている場合には、保全性処理を介して変更をキャプチャーするために、第 2 のステージング表が作成されます。
- DB2 Text Search と DB2 Net Search Extender が共に存在し、テーブル列のアクティブ Net Search Extender 索引が存在している場合、新規テキスト索引は非アクティブに設定されます。

- 新規に作成されたテキスト検索索引には、自動的にデータが追加されません。テキスト検索索引にデータが追加されるように、**UPDATE INDEX** コマンドが手動でまたは自動的に (**UPDATE FREQUENCY** オプションの指定により索引に定義されている更新スケジュールの結果として) 実行される必要があります。
- 頻度が指定されている場合、DB2 管理スケジューラー用のスケジュール・タスクが作成されます。

#### 使用上の制限

- 主キーが表に対して定義されている必要があります。DB2 Text Search では、複数列の DB2 主キーをタイプの制限なしで使用できます。主キー列の数は、DB2 によって許可されている主キー列の数より 2 列少ない数に制限されています。
- DB2 Text Search 索引のある表のすべての主キー列の合計長は、DB2 によって許可されている主キー列の最大合計長より 15 バイト少ない長さに制限されています。DB2 CREATE INDEX ステートメントの制約事項を参照してください。

複数のコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。競合するコマンドの実行中に、このコマンドが発行される場合、エラーが発生してコマンドは失敗します。その後、コマンドの実行を再試行することができます。以下のコマンドが競合します。

- **DISABLE DATABASE FOR TEXT**

索引の作成後に、テキスト索引の補助ログ・プロパティを変更することはできません。

パーティション・データベース環境では、以下の条件が満たされている必要があります。

- テキスト検索索引に固有の管理表 (ステージング表など) およびテキスト検索索引は、対応する基本表と同様の仕方で分散されます。テキスト検索索引の作成中に **ADMINISTRATION TABLES IN** 節を使用すると、指定された表スペースが、基本表の表スペースと同じパーティション・グループに入れられます。

**CJKSEGMENTATION** オプションは、中国語、日本語、および韓国語のための、zh\_CN、zh\_TW、ja\_JP、および ko\_KR ロケール・セットに適用されます。セグメンテーション方式に指定された MORPHOLOGICAL または NGRAM オプションは、SYSIBMTS.TSCONFIGURATION 管理ビューに追加されます。

**LANGUAGE** パラメーターに AUTO オプションを設定して索引を作成すると、**CJKSEGMENTATION** 指定がオプションとして許可されます。指定されたセグメンテーション方式は、中国語、日本語、および韓国語の文書に適用されます。いったん索引作成が完了したら、**CJKSEGMENTATION\_method** 値の値セットを変更することはできません。

**LANGUAGE** パラメーターに AUTO を設定し、**CJKSEGMENTATION** に MORPHOLOGICAL を設定してテキスト検索索引を作成した場合、形態素索引で有効なストリングを検索すると、予期される結果が返されないことがあります。このような場合は、以下のサンプル・ステートメントに示すように **CONTAINS** 関数の **QUERYLANGUAGE** オプションを使用して結果を取得してください。

```
select bookname from ngrambooks where contains (story, '■■■','QUERYLANGUAGE=zh_CN') = 1
```

## db2ts DISABLE DATABASE FOR TEXT

このコマンドは、コマンド **ENABLE DATABASE FOR TEXT** によって行われた変更を元に戻します (例えば、テキスト検索関連の表およびビューをドロップします)。

このコマンドが発行される時、以下のことを行います。

- データベースの DB2 Text Search フィーチャーを使用不可にします。
- テキスト検索のカタログ表とビュー、および関連するデータベース・オブジェクトをドロップします。
- **FORCE** オプションを指定した場合、すべてのテキスト索引情報はデータベースから削除され、すべての関連する集合も削除されます。『db2ts DROP INDEX コマンド』を参照してください。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

### 許可

このステートメントの許可 ID には、以下の両方の特権が含まれている必要があります。

- DBADM と DATAACCESS を併せ持つ権限。
- SYSTS\_ADM ロール。

### 必要な接続

データベース

### コマンド構文

```
►►—DISABLE DATABASE FOR TEXT—┌──┴──┐ | connection options | ───────────►
```

#### connection options:

```
┌──────────────────────────────────────────────────────────────────────────┐
```

```
└──CONNECT TO—database-name—┌──┴──┐
```

```
└──USER—username—USING—password—┘
```

### コマンド・パラメーター

#### FORCE

すべてのテキスト検索索引がデータベースから強制的にドロップされるように指定します。

このオプションが指定されておらず、テキスト検索索引がこのデータベースに対して定義されている場合、このコマンドは失敗します。

このオプションが指定されていても、DB2 Text Search サービスが開始されていない (db2ts **START FOR TEXT** コマンドが発行されていない) 場合、テキスト検索索引 (コレクション) はドロップされません。 **db2ts CLEANUP** コマンドを使用して手動でクリーンアップする必要があります。

#### **CONNECT TO** *database-name*

この節は、接続が確立される先のデータベースを指定します。データベースはローカル・システム上になければなりません。この節が指定されている場合は、環境変数 **DB2DBDFT** よりも優先されます。以下のすべてが該当する場合、この節は省略できます。

- **DB2DBDFT** 環境変数が有効なデータベース名に設定されている。
- コマンドを実行しているユーザーが、データベース・サーバーに接続するのに必要な権限を持っている。

#### **USER** *username* **USING** *password*

この節は、接続を確立するのに使用する許可名およびパスワードを指定します。

### 使用上の注意

このコマンドは、データベースにおける **DB2 Net Search Extender** の使用可能化状況には影響しません。このコマンドは **ENABLE FOR TEXT** コマンドにより作成された **DB2 Text Search** のカタログ表およびビューを削除します。

テキスト検索索引の定義を持つ **DB2** データベースをドロップする前に、このコマンドを発行し、テキスト索引およびコレクションが正常に除去されたことを確認してください。

**FORCE** オプションを使用して、いくつかの索引が削除できなかった場合、コレクション名が **db2diag** ログ・ファイルに書き込まれます。

注：結果がオーファン集合（例えば残された集合がテキスト検索サーバー上に定義された状態であるが、**DB2** には使用されない）となる使用法は推奨されていません。以下は、孤立したコレクションの原因となるいくつかのケースです。

- **DISABLE DATABASE FOR TEXT** コマンドを実行せずに **DROP DATABASE CLP** コマンドを実行するとき。
- **FORCE** オプションを使用して **DISABLE DATABASE FOR TEXT** コマンドを実行するとき。
- その他のいくつかのエラー条件。

複数のコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。競合するコマンドの実行中に、このコマンドが発行される場合、エラーが発生してコマンドは失敗します。その後、コマンドの実行を再試行することができます。以下に、競合するコマンドをいくつか示します。

- **DROP INDEX**
- **UPDATE INDEX**
- **CLEAR EVENTS FOR INDEX**
- **ALTER INDEX**
- **DISABLE DATABASE FOR TEXT**

---

## db2ts DROP INDEX

**db2ts DROP INDEX** コマンドは、既存のテキスト検索索引をドロップします。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

## 許可

ステートメントの許可 ID が持つ特権には、SYSTS\_MGR ロールと、以下のいずれかの特権または権限が含まれている必要があります。

- その索引の定義されている表に対する CONTROL 特権。
- その索引の定義されているスキーマに対する DROPIN 特権。
- テキスト検索索引に既存のスケジュールがある場合、許可 ID は索引作成者と同じであるか、または DBADM 権限を持っている必要があります。

## 必要な接続

データベース

## コマンド構文

```
▶▶ DROP INDEX index-name FOR TEXT
```

```
└── drop options ─┘ | connection options |
```

### connection options:

```
└── CONNECT TO database-name ─┘
```

```
└── USER username USING password ─┘
```

## コマンド・パラメーター

### DROP INDEX *index-name* FOR TEXT

**CREATE INDEX** コマンドで指定されている索引のスキーマおよび名前。データベース内のテキスト検索索引を固有に識別します。

### *drop\_options*

IBM の内部使用のために予約済みです。

### CONNECT TO *database-name*

この節は、接続が確立される先のデータベースを指定します。データベースはローカル・システム上になければなりません。この節が指定されている場合は、環境変数 DB2DBDFT よりも優先されます。以下のステートメントすべてが該当する場合、この節は省略できます。

- DB2DBDFT 環境変数が有効なデータベース名に設定されている。
- コマンドを実行しているユーザーが、データベース・サーバーに接続するのに必要な権限を持っている。

### USER *username* USING *password*

この節は、接続を確立するのに使用する許可名およびパスワードを指定します。

## 使用上の注意

複数のコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。競合するコマンドの実行中に、このコマンドが発行

される場合、エラーが発生してコマンドは失敗します。その後、コマンドの実行を再試行することができます。以下のコマンドは、一般的な競合コマンドです。

- **DROP INDEX**
- **UPDATE INDEX**
- **CLEAR EVENTS FOR INDEX**
- **ALTER INDEX**
- **DISABLE DATABASE FOR TEXT**

**DROP** 操作と並行して **STOP FOR TEXT** コマンドを実行しても、コマンドの競合に関するメッセージは出されません。その代わりに、**DROP** によってコレクションが削除される前にテキスト検索サーバーがシャットダウンされると、テキスト検索サーバーが使用できないというエラーが返されます。

テキスト検索索引がドロップされた後、テキスト検索は対応するテキスト列で使用できなくなります。同じテキスト列に新規のテキスト検索を作成する予定の場合、まずデータベースから切断し、再接続してから、新規のテキスト検索索引を作成する必要があります。

**db2ts DROP INDEX FOR TEXT** コマンドは、データベースに以下の変更を行います。

- DB2 Text Search のカタログ情報を更新します。
- 索引ステージング表およびイベント表をドロップします。
- ユーザー・テキスト表のトリガーを削除します。
- DB2 Text Search 索引定義に関連付けられたコレクションを破棄します。

---

## db2ts ENABLE DATABASE FOR TEXT

**db2ts ENABLE DATABASE FOR TEXT** コマンドは、現行データベースで DB2 Text Search を使用できるようにします。このコマンドは管理表および管理ビューを作成して各パラメーターにデフォルト値を設定します。データベース内の表の列にテキスト検索索引を作成できるようにするには、このコマンドを正常に実行する必要があります。コマンド行でコマンドの前に **db2ts** を付ける必要があります。

データベースを使用可能にした後、**SYSIBMTS.TSSERVERS** ビュー内の Text Search サーバーの接続情報を指定する必要があります。使用可能にする操作には、サーバーにデータを設定しようとする操作が含まれるので、サーバー構成にアクセスできない場合には警告が表示されます。どのような場合であれ、ビューの接続情報を確認することをお勧めします。詳細については、DB2 Text Search サーバー情報の更新に関するトピックを参照してください。

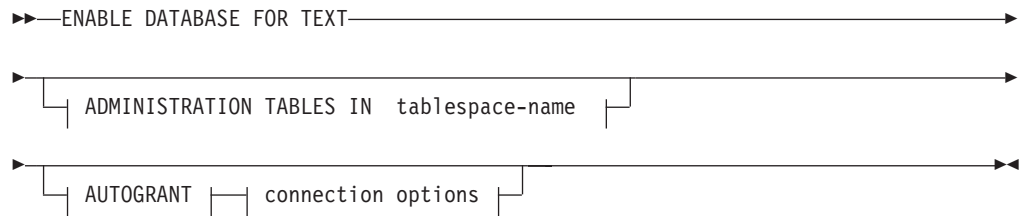
### 許可

- このステートメントの許可 ID が持つ特権には、**SYSTS\_ADM** ロールおよび **DBADM** 権限が含まれている必要があります。

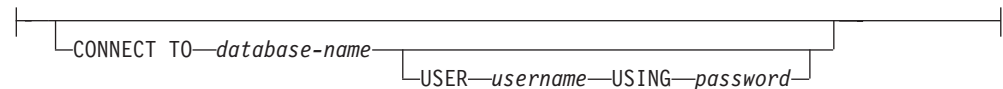
### 必要な接続

データベース

## コマンド構文



### connection options:



## コマンド・パラメーター

### ADMINISTRATION TABLES IN *tablespace-name*

データベースを DB2 Text Search に使用可能にする際に作成される管理表のための、既存の REGULAR 表スペースの名前を指定します。この表スペースはデータベース・パーティション・グループ IBMCATGROUP 内にあることが推奨されています。パーティション・データベースの場合、バッファ・プールおよび表スペースは 32 KB ページ・サイズで定義されている必要があります。

この節を指定しない場合、SYSTOOLSPACE がデフォルトの表スペースとして使用されます。この場合、SYSTOOLSPACE が既に存在していることを確認してください。存在しない場合は、SYSPROC.SYSINSTALLOBJECTS プロシージャを使用して作成できます。

注: 大/小文字の区別のある表スペース名を指定するには、引用符を使用してください。

### AUTOGRANT

このオプションは非推奨で、インスタンス所有者に特権を付与しなくなりました。使用はもはや推奨されておらず、今後のリリースで除去される可能性があります。

### CONNECT TO *database-name*

この節は、接続が確立される先のデータベースを指定します。データベースはローカル・システム上になければなりません。この節が指定されている場合は、環境変数 DB2DBDFT よりも優先されます。以下のステートメントすべてが該当する場合、この節は省略できます。

- DB2DBDFT 環境変数が有効なデータベース名に設定されている。
- コマンドを実行しているユーザーが、データベース・サーバーに接続するのに必要な権限を持っている。

### USER *username* USING *password*

この節は、接続を確立するのに使用する許可名およびパスワードを指定します。

## 例

例 1: tsspace という名前の表スペース内に管理表を作成してデータベースを DB2 Text Search に使用できるようにし、すべてのエラー・メッセージを英語で戻します。

```
CALL SYSPROC.SYSTS_ENABLE('ADMINISTRATION TABLES IN tsspace', 'en_US', ?)
```

以下はこの照会の出力例です。

```
Value of output parameters
-----
Parameter Name   : MESSAGE
Parameter Value  : Operation completed successfully.

Return Status = 0
```

## 使用上の注意

正常に実行されるとき、このコマンドは以下のアクションを行います。

- データベースで DB2 Text Search フィーチャーを使用可能にします。
- ビュー SYSIBMTS.TSDEFAULTS に DB2 Text Search のデータベース構成デフォルト値を確立します。
- 以下の DB2 Text Search 管理ビューを SYSIBMTS スキーマに作成します。
  - SYSIBMTS.TSDEFAULTS
  - SYSIBMTS.TSLOCKS
  - SYSIBMTS.TSINDEXES
  - SYSIBMTS.TSCONFIGURATION
  - SYSIBMTS.TSCOLLECTIONNAMES
  - SYSIBMTS.TSSERVERS

---

## db2ts HELP

**db2ts HELP** は、使用可能な DB2 Text Search コマンドのリスト、または個々のコマンドの構文を表示します。

**db2ts HELP** コマンドを使用して、特定のエラー・メッセージに関するヘルプも表示できます。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

## 許可

なし。

## コマンド構文





## コマンド・パラメーター

### HELP | ?

コマンドまたは理由コードのヘルプ情報を表示します。

#### *command*

DB2 Text Search コマンドを識別する最初のキーワード:

- ALTER
- CLEANUP
- CLEAR (CLEAR COMMAND LOCKS と CLEAR EVENTS FOR INDEX の両方)
- CREATE
- DISABLE
- DROP
- ENABLE
- RESET PENDING
- START
- STOP
- UPDATE

#### *sqlcode*

(管理ストアード・プロシージャの内部または外部で) db2ts コマンドまたはテキスト検索照会によって戻されるメッセージの SQLCODE。

#### *sqlstate*

コマンド、管理ストアード・プロシージャ、またはテキスト検索照会で戻される sqlstate。

#### *error-identifier*

ID は、エラー・メッセージに組み込まれる *text-search-error-msg* の一部です。この ID は 'CIE' で始まり、CIE $n$  の形式になります。 $n$  は数値です。この ID は、テキスト検索中のエラーの際に戻される特定のエラーを表します。これはまた、テキスト検索コマンドの終了時の通知メッセージ、またはテキスト検索管理プロシージャの完了時に印刷されるメッセージによって戻されることもあります。ID が 'CIE' から始まっていない場合、**db2ts help** は *error-identifier* に関する情報を提供できません。例えば、db2ts は IQQR0012E などの *error-identifier* によるメッセージのヘルプを提供できません。

## 使用上の注意

UNIX シェルを使用するときは、以下の例のように、二重引用符を使って **db2ts** に引数を指定する必要がある場合があります。

```
db2ts "? CIE00323"
```

引用符がないと、シェルはワイルドカードを作業ディレクトリーの内容と一致させようとするので、予期しない結果が生じることがあります。

いずれかの db2ts コマンドの最初のキーワードが指定された場合、識別されたコマンドの構文が表示されます。2 つの db2ts コマンドが同じ最初のキーワードを共有

する場合 (**CLEAR COMMAND LOCKS** と **CLEAR EVENTS FOR INDEX**)、`db2ts help clear` を発行すると両方のコマンドの構文が表示されますが、`db2ts help clear events` などのように 2 番目のキーワードを追加してそれらを区別すると、各コマンドを特定して表示することができます。? または **HELP** の後にパラメーターが指定されていない場合、`db2ts` は使用可能なすべての `db2ts` コマンドをリストします。

`sqlcode`、`sqlstate`、または `CIE error-identifier` を指定すると、そのコード、状態、またはエラー ID に関する情報が戻されます。例えば、

```
db2ts help SQL20423
```

または

```
db2ts ? 38H10
```

または

```
db2ts ? CIE00323
```

---

## db2ts RESET PENDING コマンド

特定の表に関連付けられた、すべてのテキスト保守ステージング表に対して **SET INTEGRITY** ステートメントを発行します。

一部のコマンドによって DB2 Text Search ステージング表が保留モードに入れられ、他のデータベース操作やテキスト検索操作がブロックされることがあります。**db2ts RESET PENDING** コマンドを使用すれば、すべてのテキスト索引とそれに関連するステージング表を検索してから、表ごとに **SET INTEGRITY** ステートメントを発行する必要はありません。

範囲パーティションをデタッチした後は、**RESET PENDING** コマンドを発行してステージング表の内容を更新する必要があります。

### 許可

このコマンドは、**SYSTS\_MGR** ロールと、以下の権限または特権の少なくとも 1 つを必要とします。

- **DATAACCESS** 権限
- テキスト索引が作成されている基本表に対する **CONTROL**

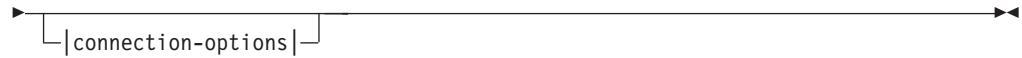
注: 現行では、新規索引表の作成またはドロップを許可する **ALL** 特権が **SYSTS\_MGR** に付与されます。ただし、索引表に、索引のような従属オブジェクトが暗黙的に作成された場合、許可は伝搬されません。従属オブジェクトを削除するためには、ユーザーに **CONTROL** 特権を付与します。

### 必要な接続

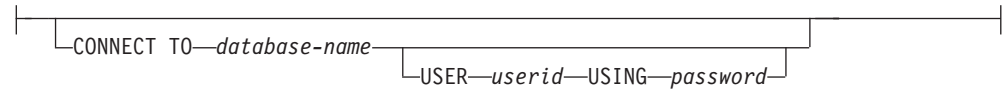
このコマンドは DB2 データベース・サーバーから発行しなければなりません。

### コマンド構文

```
►►—RESET PENDING FOR TABLE— table-schema.table-name—FOR TEXT—►►
```



### Connection-options:



## コマンド・パラメーター

### *table-name*

テキスト保守のステージング・インフラストラクチャーが追加されて保全性処理を必要とする表の名前。

### *table-schema*

コマンドが発行された結果として、保留モードに入った表のスキーマ。

## 使用上の注意

**RESET PENDING** コマンドは、基礎表が保留モードに入れられる原因となるコマンド (**INSERT** パラメーター付きの **LOAD** コマンドなど) を発行した後に、または **SET INTEGRITY** ステートメントで従属表をリフレッシュすることを必要とするコマンド (**ALTER TABLE ... DETACH** ステートメントなど) を発行した後に使用します。

## db2ts START FOR TEXT

**db2ts START FOR TEXT** コマンドは、他の DB2 Text Search 管理コマンドおよび SQL 照会内のテキスト検索索引を参照する機能をサポートする DB2 Text Search インスタンス・サービスを開始します。

サーバーにリッチ・テキスト・サポートが構成されている場合、**db2ts START FOR TEXT** コマンドで実行される操作には、DB2 Text Search サーバーを実行するホスト・マシン上のリッチ・テキスト・サポートのためのプロセスを開始する操作も含まれます。

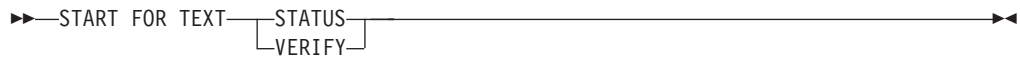
このコマンドは DB2 データベース・サーバーから発行されなければなりません。

統合テキスト検索のセットアップを使用しているパーティション・データベース環境でインスタンス・サービスを開始する場合は、統合テキスト検索サーバー・ホスト・マシン上でコマンドを実行する必要があります。デフォルトでは、統合テキスト検索サーバー・ホスト・マシンは、最も小さい番号のデータベース・パーティション・サーバーのホストです。

## 許可

インスタンス所有者。データベース特権は要求されていない

## コマンド構文



## コマンド・パラメーター

### STATUS

DB2 Text Search サーバーの状況を検査します。サーバーの **STARTED** 状況または **STOPPED** 状況を示す詳細情報メッセージが返されます。

### VERIFY

DB2 Text Search サーバーの開始状況を検査し、操作が成功した場合にはそのことを示す戻りコード **0** を標準メッセージと共に返して終了します。テキスト検索サーバーのその他の状況の場合、または状況を検査できない場合には、ゼロ以外のコードが戻されます。

## 例

- テキスト検索サーバーが開始されたことを検査します。

```
Linux/UNIX:  
$ db2ts START FOR TEXT VERIFY  
CIE00001 Operation completed successfully.
```

```
$ echo $?  
0
```

```
Windows:  
C:¥> db2ts START FOR TEXT VERIFY  
CIE00001 Operation completed successfully.
```

```
C:¥> echo %ERRORLEVEL%  
0
```

## 使用上の注意

- パーティション・データベース環境では、いずれか 1 つのパーティション・サーバー・ホストに対して、**STATUS** および **VERIFY** パラメーターを付けて **db2ts START FOR TEXT** コマンドを発行できます。インスタンス・サービスを開始するには、統合テキスト検索サーバー・ホスト・マシン上で **db2ts START FOR TEXT** コマンドを実行する必要があります。統合テキスト検索サーバーのホスト・マシンは、最も小さい番号のデータベース・パーティション・サーバーのホストです。カスタム・コレクション・ディレクトリーが使用される場合、これより番号の小さいパーティションが後で作成されないようにしてください。この制限は特に、Linux および UNIX プラットフォームで該当します。インスタンスの作成時に DB2 Text Search を構成する場合には、構成によって最初に統合テキスト検索サーバー・ホストが判別されます。その構成が常に、最も小さい番号のデータベース・パーティション・サーバーのホストでなければなりません。
- Windows プラットフォームでは、DB2 Text Search 用の DB2 インスタンスごとに、関連付けられた Windows サービスが 1 つずつあります。次のコマンドを発行することにより、サービス名を判別することができます。

```
DB2TS - <instance name>[-<partition number>]
```

・ **db2ts START FOR TEXT** コマンドを使用することのほかに、「コントロール パネル」または **NET START** コマンドを使用してサービスを開始することもできます。

---

## db2ts STOP FOR TEXT

**db2ts STOP FOR TEXT** コマンドは、DB2 Text Search インスタンス・サービスを停止します。実行中のサービスに、リッチ・テキスト・サポート用のプロセスが含まれている場合には、それらのサービスも停止されます。

このコマンドは DB2 データベース・サーバーから発行されなければなりません。

このコマンドをコマンド行から実行するときは、DB2 コマンド行で、コマンドの接頭部として **db2ts** を付けてください。

このコマンドを使用すると、独立型のテキスト検索サーバーを簡単に停止できます。そのサーバーがインストールされている環境で、付属スクリプトを使用して停止することもできます。インスタンス・サービスが既に停止している場合には、このコマンドはその状況をチェックしてユーザーに報告するだけです。

### 許可

インスタンス所有者。データベース特権は要求されていない

### コマンド構文

▶ **STOP FOR TEXT** STATUS  
VERIFY →

### コマンド・パラメーター

#### STATUS

DB2 Text Search サーバーの状況を検査します。サーバーの **STARTED** または **STOPPED** 状況を示す詳細情報メッセージが戻されます。

#### VERIFY

DB2 Text Search サーバーの停止状況を検査します。コマンドが正常に実行されたことを示す場合は、標準メッセージおよび戻りコード **0** で終了します。それ以外の場合は、テキスト検索サーバーはゼロでないコードを戻して失敗を示します。

### 使用上の注意

- ・ 現在実行中のコマンドの実行の中断を避けるため、**db2ts STOP FOR TEXT** コマンドを発行する前に、依然としてアクティブな **db2ts UPDATE INDEX FOR TEXT** コマンドのような他の管理コマンドがないことを確認してください。
- ・ パーティション・データベース環境では、いずれか 1 つのパーティション・サーバー・ホストに対して、**STATUS** および **VERIFY** パラメーターを付けて **db2ts START FOR TEXT** コマンドを発行できます。
- ・ 統合テキスト検索サーバーを使用している Windows プラットフォーム上のパーティション・データベース環境では、統合テキスト検索サーバー・ホスト・マシン上で **db2ts STOP FOR TEXT** コマンドを発行することにより、インスタンス・サ

ービスを停止します。デフォルトでは、統合テキスト検索サーバー・ホスト・マシンは、最も小さい番号のデータベース・パーティション・サーバーのホストです。統合テキスト検索サーバー・ホスト・マシン上でこのコマンドを実行すれば、すべてのプロセスおよびサービスが確実に停止します。別のパーティション・サーバー・ホスト上でコマンドが実行される場合には、NET STOP などのコマンドを使用して、DB2TS サービスを停止する必要があります。

---

## db2ts UPDATE INDEX

**db2ts UPDATE INDEX** コマンドはテキスト検索索引 (DB2 Text Search のコレクション) を更新して、索引に関連付けられたテキスト列の現在の内容を反映します。更新が実行されている間も、検索は可能です。更新が完了するまでは、検索は部分的に更新された索引に対して実行されます。

実行するには、コマンド行でコマンドの前に **db2ts** を付ける必要があります。

### 許可

ステートメントの許可 ID が保持する特権には、SYSTS\_MGR ロールと以下の少なくとも 1 つの権限が含まれている必要があります。

- DATAACCESS 権限
- そのテキスト索引の定義されている表に対する CONTROL 特権。
- テキスト索引が定義されている基本表に対する、INDEX 特権および SELECT 特権。

また、初期更新の場合は、**CREATE TRIGGER** ステートメントで概略されている権限の要件が適用されます。

### 必要な接続

データベース

### コマンド構文

```
▶▶ UPDATE INDEX—index-name—FOR TEXT—  
└──────────────────┬──────────────────┘  
UPDATE OPTIONS  
▶└── connection options ───────────────────▶
```

### connection options:

```
└──────────────────────────────────────────┘  
└── CONNECT TO—database-name—  
└──────────┬──────────┘  
USER—username—USING—password
```

### コマンド・パラメーター

**UPDATE INDEX** *index-name*

更新されるテキスト検索索引の名前を指定します。索引名は、DB2 索引の命名上の制約に従う必要があります。

## UPDATE OPTIONS

更新オプションを指定する、タイプ VARCHAR(32K) の入力引数。オプションを指定しない場合、更新は無条件に開始されます。可能な値は次のとおりです。

UPDATE OPTIONS の値	説明
USING UPDATE MINIMUM	このオプションは、テキスト検索索引に定義されている UPDATE MINIMUM 値を使用するように強制して、指定された最小数の変更が発生したら更新を処理します。
FOR DATA REDISTRIBUTION	このオプションは、データ・パーティションが追加または削除され、後続のデータ再配分操作が完了した後に、パーティション・データベース内のテキスト検索索引をリフレッシュするように指定します。FOR DATA REDISTRIBUTION オプションでテキスト検索索引が更新されるまでは、検索結果に矛盾が生じる可能性があります。
ALLROWS	このオプションは、初期更新を無条件に試行するように指定します。
UPDATEAUTOCOMMIT	このオプションは索引更新の回数を指定するもので、初期更新またはインクリメンタル更新のいずれかにおいてこの回数に達すると、コミットが実行され、それまでの作業が自動的に保存されます。 <ul style="list-style-type: none"><li>• イニシャル更新では、データ更新をキャプチャーするトリガーがアクティブ化された後の基本表からの文書のバッチを、索引更新が処理します。文書更新の回数が COMMITCOUNT の数に達すると、サーバーは暫定コミットを実行します。処理されていない文書により生成されたログ項目は、ステージング表から除去されません。イニシャル・テキスト索引更新に UPDATEAUTOCOMMIT オプションを使用すると、実行時間が大幅に増加します。</li><li>• インクリメンタル更新では、処理されているログ項目は、各暫定コミットで対応するステージング表から除去されます。COMMITCOUNT は更新される文書の数であり、ステージング表項目の数ではありません。</li></ul>

## CONNECT TO *database-name*

この節は、接続が確立される先のデータベースを指定します。データベースはローカル・システム上になければなりません。この節が指定されている場合は、環境変数 DB2DBDFT よりも優先されます。以下のステートメントすべてが該当する場合、この節は省略できます。

- DB2DBDFT 環境変数が有効なデータベース名に設定されている。

- コマンドを実行しているユーザーが、データベース・サーバーに接続するのに必要な権限を持っている。

**USER** *username* **USING** *password*

この節は、接続を確立するのに使用する許可名およびパスワードを指定します。

## 使用上の注意

DB2 のデータベース・オブジェクトおよび照会に適用されるすべての制限および命名規則は、DB2 Text Search のフィーチャーおよび照会にも適用されます。DB2 Text Search 関連の ID は、DB2 の命名規則に準拠している必要があります。さらに、追加の制限事項がいくつかあります。例えば、これらの ID は以下の形式のみになります。

[A-Za-z][A-Za-z0-9@#\$\_]\*

または

"[A-Za-z ][A-Za-z0-9@#\$\_ ]\*"

シノニム・ディクショナリーがテキスト索引用に作成されている場合には、ALLROWS および FOR DATA REDISTRIBUTION 更新オプションを実行すると、既存のコレクションからディクショナリーが削除されます。データベース・パーティションが追加された後には、テキスト索引に関連付けられた新しいコレクションを設けることができます。関連するすべてのコレクションのためのシノニム・ディクショナリーを再び追加する必要があります。

このコマンドは、すべての索引更新処理が完了するまで戻りません。この期間は、これから索引付けされる文書の数および既に索引付けされている文書の数に応じて異なります。索引のコレクション名は SYSIBMTS.TSCOLLECTIONNAMES ビュー (列 COLLECTIONNAME) から取られます。

複数のコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。競合するコマンドの実行中に、このコマンドを実行すると、エラーが発生してコマンドは失敗します。その後、コマンドの実行を再試行することができます。以下のコマンドは、一般的な競合コマンドの一部です。

- **UPDATE INDEX**
- **CLEAR EVENTS FOR INDEX**
- **ALTER INDEX**
- **DROP INDEX**
- **DISABLE DATABASE FOR TEXT**

**注:** 個々の文書エラーの場合、文書を訂正する必要があります。エラーがある文書の主キーは、索引のイベント表で参照できます。ユーザー表の対応する行が変更される場合、次の **UPDATE INDEX** コマンドはこれらの文書を再処理します。

データベースへの変更:

- イベント表に行を挿入します (DB2 Text Search からのパーサー・エラー情報を含む)。
- インクリメンタル更新の場合に、索引のステージング表から削除します。
- 最初の更新の前に、ユーザー・テキスト表にトリガーを作成します。



- コレクションが更新されます。
- 新規または変更された文書が構文解析および索引付けされます。
- 削除された文書が索引から破棄されます。



---

## 第 11 章 DB2 Text Search のストアード・プロシージャー

DB2 Text Search は、いくつかの管理 SQL ルーチンを提供しています。それらのルーチンを使用して、コマンドを実行し、実行するコマンドの結果メッセージと結果メッセージ理由コードを戻すことができます。

管理 SQL ルーチンを使用して、以下の `db2ts` コマンドを実行できます。

- データベースを有効にする - `SYSPROC.SYSTS_ENABLE`
- データベースを構成する - `SYSPROC.SYSTS_CONFIGURE`
- データベースを無効にする - `SYSPROC.SYSTS_DISABLE`
- テキスト索引を作成する - `SYSPROC.SYSTS_CREATE`
- テキスト索引を更新する - `SYSPROC.SYSTS_UPDATE`
- テキスト索引を変更する - `SYSPROC.SYSTS_ALTER`
- テキスト索引をドロップする - `SYSPROC.SYSTS_DROP`
- テキスト索引のイベントを消去する - `SYSPROC.SYSTS_CLEAR_EVENTS`
- コマンドのロックを解除する - `SYSPROC.SYSTS_CLEAR_COMMANDLOCKS`
- 保留状況をリセットする - `SYSPROC.SYSTS_ADMIN_CMD`
- 非アクティブの索引をクリーンアップする - `SYSPROC.SYSTS_CLEANUP`

---

### SYSTS\_ADMIN\_CMD プロシージャー - テキスト検索管理コマンドの実行

`SYSTS_ADMIN_CMD` プロシージャーは、SQL CALL ステートメントを使用してテキスト検索管理コマンドを実行するアプリケーションで使用されます。

#### 許可

`SYSTS_ADMIN_CMD` プロシージャーに対する EXECUTE 特権、および要求される操作用にリストされている必要な権限。

#### デフォルトの PUBLIC 特権

制限のないデータベースでは、このプロシージャーが自動的に作成されると、EXECUTE 特権が PUBLIC に付与されます。

#### 構文

```
▶▶—SYSTS_ADMIN_CMD—(—command-string—,—message-locale—,—message—)————▶▶
```

スキーマは SYSPROC です。

#### プロシージャー・パラメーター

##### *command-string*

実行される単一の DB2 Text Search 索引管理コマンドを指定する、タイプ VARCHAR (32 K) の入力引数。コマンド構文は、このプロシージャーではサボ

ートされていない接続オプションを除き、DB2 Text Search コマンドと同じです。このプロシージャから発行されるコマンドは、現在接続を使用します。このプロシージャは、以下の DB2 Text Search コマンドをサポートします。

- ALTER INDEX
- CLEAR COMMAND LOCKS
- CLEAR EVENTS
- CREATE INDEX
- DISABLE DATABASE
- DROP INDEX
- ENABLE DATABASE
- RESET PENDING
- UPDATE INDEX

#### *message-locale*

戻されるすべてのエラー・メッセージ・テキスト用に使用される言語を指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

#### *message*

正常と見なされる操作に対する警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

以下の例は、スキーマ DB2TS の MYTEXTINDEX テキスト検索索引を更新し、すべてのエラー・メッセージを英語で戻します。

```
CALL SYSPROC.SYSTS_ADMIN_CMD  
('UPDATE INDEX DB2TS.MYTEXTINDEX FOR TEXT','en_US', ?)";
```

出力例を以下に示します。

```
Value of output parameters  
-----  
Parameter Name : MESSAGE  
Parameter Value : CIE00001 Operation completed successfully.  
  
Return Status = 0
```

## 使用上の注意

コマンドが正常に実行されない場合、SQLCODE -20427 および SQLSTATE 38H14 がテキスト検索特有のエラー・メッセージとともに戻されます。例えば、索引 MYTEXTINDEX が存在しており、以下のステートメントが発行された場合:

```
CALL SYSPROC.SYSTS_ADMIN_CMD ('CREATE INDEX MYTEXTINDEX FOR TEXT  
ON DB2TS.TEXTBOOKS (STORY)', 'en_US', ?)
```

索引の作成が、次のエラー・メッセージで失敗します。

```
SQL20427N An error occurred during a text search administration  
procedure or command. The error message is "CIE00201 Text search  
index "DB2TS ". "MYTEXTINDEX" already exists. ". SQLSTATE=38H14
```

SQLCODE がプロシージャーにより戻される場合、メッセージは切り捨てられる可能性があります。詳細なメッセージ情報は **db2diag** ログ・ファイル内にあります。

## SYSTS\_ALTER プロシージャー - 索引の更新特性の変更

このプロシージャーは、索引の更新特性を変更します。

このプロシージャーは **ALTER INDEX** テキスト検索管理コマンドをデータベース・サーバー上で発行します。

### 許可

ステートメントの許可 ID が保持する特権には、SYSTS\_MGR ロールと以下の少なくとも 1 つの権限が含まれている必要があります。

- DBADM 権限
- 基本スキーマに対する ALTERIN 特権
- そのテキスト検索索引が定義されている基本表に対する CONTROL または ALTER 特権

既存のスケジュールを変更するには、許可 ID が索引の作成者と同じであるか、DBADM 権限が含まれている必要があります。

### デフォルトの PUBLIC 特権

なし

### 構文

```
► SYSTS_ALTER ( ( index_schema , index_name , update characteristics )
► options ( , message_locale , message ) )
```

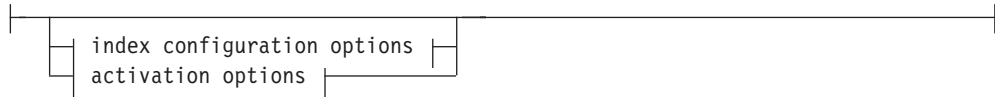
#### update characteristics:

```
|-----|
| UPDATE FREQUENCY NONE |
| | update frequency |
|-----|
| UPDATE MINIMUM minchanges |
|-----|
```

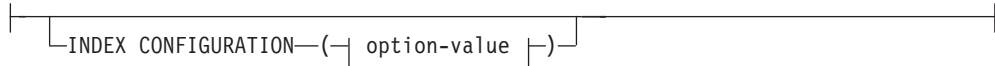
#### update frequency:

```
|-----D-----|-----H-----|-----M-----|
| * | * | integer3 |
| | | |
| integer1 | integer2 |
|-----|-----|-----|
```

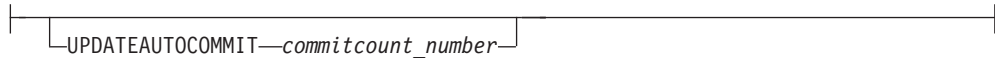
## オプション:



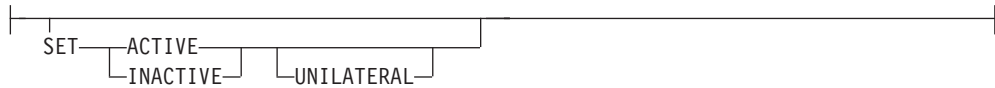
## index configuration options:



## option-value:



## activation options:



スキーマは SYSPROC です。

## プロシージャ・パラメーター

### *index\_schema*

テキスト検索索引のスキーマを指定する、タイプ VARCHAR(128) の入力引数。 *index\_schema* は、DB2 スキーマ名の命名上の制約に従う必要があります。引数が NULL または空ストリングである場合、値 CURRENT SCHEMA が使用されます。 *index\_schema* には、大文字小文字の区別があります。

### *index\_name*

索引の名前を指定する、タイプ VARCHAR (128) の入力引数。 *index\_schema* とともに使用すると、データベースのテキスト検索索引を一意的に識別します。 *index\_name* には、大文字小文字の区別があります。

## update characteristics

変更オプションを指定する、タイプ VARCHAR(32K) の入力引数。指定可能な変更オプションは以下のとおりです。

### UPDATE FREQUENCY

索引の更新が行われる頻度を指定します。変更回数が **UPDATE MINIMUM** に設定された値以上である場合、索引は更新されます。更新頻度 **NONE** は、索引の更新はそれ以降行われないことを意味します。これは、変更されないデータを持つ表のテキスト列に役立ちます。さらに、これはユーザーが (**UPDATE INDEX** コマンドを使用して) 手で索引を更新する予定の場合にも役立ちます。自動更新を行えるのは、DB2\_ATS\_ENABLE レジストリー変数が設定されていて、**START FOR TEXT** コマンドが発行される場合のみです。

デフォルトの頻度値はビュー SYSIBMTS.TSDEFAULTS から取られ、DEFAULTNAME='UPDATEFREQUENCY' となります。

**NONE**

自動更新は、テキスト索引に適用されません。以降のすべての索引更新は手動で開始する必要があります。

**D** 索引が更新される曜日。

- \* 毎日。

*integer1*

日曜日から土曜日の特定の曜日: 0 から 6

**H** 索引が更新される特定の日の時。

- \* 毎時。

*integer2*

午前 0 時から午後 11 時までの特定の時: 0 から 23

**M** 索引が更新される特定の時刻の分。

*integer3*

**UPDATE FREQUENCY** オプションを指定しない場合、頻度の設定は変更されな  
いままとなります。

**UPDATE MINIMUM** *minchanges*

索引が増分的に更新される前に生じなければならない、テキスト文書への変更の最小数を指定します。同じテキスト文書への複数の変更は、別個の変更として扱われます。**UPDATE MINIMUM** オプションを指定しない場合、設定は変更されな  
いままとなります。

**INDEX CONFIGURATION** (*option-value*)

バージョン 9.7 フィックスパック 3 以降のフィックスパックでは、これは、テキスト索引構成設定を変更できる VARCHAR(32K) タイプのオプションの入力引数です。以下のオプションがサポートされています。

表 12. オプション値の仕様

オプション	値	データ・タイプ	説明
SERIALUPDATE	<i>updatemode</i>	整数	<p>パーティション化されたテキスト検索索引の更新処理を並列モードと直列モードのどちらで行うかを指定します。並列モードの場合、実行が複数のデータベース・パーティションに分散され、それぞれのノードで独立して行われます。直列モードでは、実行は分散されずに行われ、障害が検出されると停止します。通常、直列モード実行の方が時間はかかりますが、必要とするリソースがかなり少なくて済みます。</p> <ul style="list-style-type: none"> <li>• 0 = 並列モード</li> <li>• 1 = 直列モード</li> </ul>



表 12. オプション値の仕様 (続き)

オプション	値	データ・タイプ	説明
UPDATEAUTO COMMIT	<i>commitcount</i> <i>_number</i>	整数	<p>ここで指定された回数の索引更新が行われると、コミットが実行され、イニシャル更新またはインクリメンタル更新のためにそれまでの作業が自動的に保存されます。</p> <ul style="list-style-type: none"> <li>イニシャル更新では、データ更新をキャプチャーするトリガーがアクティブ化された後の基本表からの文書のバッチを、索引更新が処理します。文書更新の回数が COMMITCOUNT に達すると、サーバーは暫定コミットを実行します。処理されていない文書により生成されたログ項目は、ステージング表から除去されます。イニシャル・テキスト索引更新に UPDATEAUTOCOMMIT オプションを使用すると、実行時間が大幅に増加します。</li> <li>インクリメンタル更新では、処理されているログ項目は、各暫定コミットで対応するステージング表から除去されます。COMMITCOUNT は更新される文書の数であり、ステージング表項目の数ではありません。</li> </ul>

*activation options*

バージョン 9.7 フィックスパック 3 以降のフィックスパックでは、この整数タイプの入力引数はテキスト索引の状況を設定します。

**ACTIVE**

テキスト索引状況をアクティブに設定します

**INACTIVE**

テキスト索引状況を非アクティブに設定します

## UNILATERAL

DB2 Text Search 索引の状況にのみ影響する UNILATERAL 変更を指定します。この引数が指定された場合、DB2 Text Search 索引の状況だけがアクティブまたは非アクティブに変更されます。UNILATERAL 引数がない場合、DB2 Text Search 索引および DB2 Net Search Extender 索引のアクティブ化の状況はともに切り替えられ、テキスト索引の一方だけがアクティブ化されます。

### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 以下の例では、テキスト検索索引の更新特性が変更されます。この索引は、最初は *index\_schema* 'db2ts' および *index\_name* 'myTextIndex' で作成されています。関連する表列に変更が加えられないことが予想されるため、テキスト検索索引にそれ以上の更新をしないという意図で UPDATE FREQUENCY NONE を使用します。すべてのエラー・メッセージは英語で戻すことが要求されます。このプロシージャが成功すると、操作の成功を示す出力パラメーター・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_ALTER('db2ts', 'myTextIndex',  
'UPDATE FREQUENCY NONE', 'en_US', ?)
```

以下はこの照会の出力例です。

```
Value of output parameters  
-----  
Parameter Name : MESSAGE  
Parameter Value : Operation completed successfully.  
  
Return Status = 0
```

例 2: 以下の例では、SYSTS\_ALTER ストアード・プロシージャが呼び出され、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex2' があるテキスト検索索引の更新特性が変更されます。この意図は、索引への更新が毎時行われるということです。ただし、この索引は存在せず、結果としてエラーを戻します。

```
CALL SYSPROC.SYSTS_ALTER('db2ts', 'myTextIndex2',  
'update frequency D(*) H(*) M(0)', 'en_US', ?)
```

以下はこの照会の出力例です。

```
SQL20427N An error occurred during a text search administration  
procedure or command. The error message is "CIE00316 Text search  
index "db2ts"."myTextIndex2" does not exist. ". SQLSTATE 38H14
```

## 使用上の注意

- テキスト検索管理プロシージャは、データベースへの既存の接続を使用します。テキスト検索管理プロシージャ内のコミットまたはロールバックからの予

期しない影響を避けるため、プロシージャーを実行する前に、すべてのトランザクション変更をコミットすることをお勧めします。これを行う 1 つの方法は、AUTOCOMMIT をオンにすることです。

- 複数のプロシージャーまたはコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。競合するプロシージャーおよびコマンドの一部は以下のとおりです。
  - SYSTS\_ALTER プロシージャーまたは ALTER INDEX db2ts コマンド
  - SYSTS\_CLEAR\_EVENTS プロシージャーまたは CLEAR EVENTS FOR INDEX db2ts コマンド
  - SYSTS\_DISABLE プロシージャーまたは DISABLE DATABASE FOR TEXT db2ts コマンド
  - SYSTS\_DROP プロシージャーまたは DROP INDEX db2ts コマンド
  - SYSTS\_UPDATE プロシージャーまたは UPDATE INDEX db2ts コマンド
  - SYSTS\_CONFIGURE プロシージャー

競合がある場合、プロシージャーは SQLCODE -20426 および SQLSTATE 38H13 を戻します。

- 頻度を変更するためにこのプロシージャーを実行する場合、テキスト索引用にスケジュール・タスクが作成、変更、またはドロップされます。
- 索引をアクティブにした結果は、元の索引の状況により異なります。次の表は、結果を示したものです。

表 13. 無効索引無し of 状況変更:

イニシャルの DB2 Text Search または Net Search Extender の状況	ACTIVE 要求	ACTIVE UNILATERAL 要求	INACTIVE 要求	INACTIVE UNILATERAL 要求
アクティブ / 非アクティブ	変更なし	変更なし	非アクティブ / アクティブ	非アクティブ / 非アクティブ
非アクティブ / アクティブ	アクティブ / 非アクティブ	エラー	変更なし	変更なし
非アクティブ / 非アクティブ	アクティブ / 非アクティブ	アクティブ / 非アクティブ	非アクティブ / アクティブ	変更なし

SQL20427N および CIE0379E エラー・メッセージが、索引のアクティブ化競合の場合に戻されます。

## SYSTS\_CLEANUP プロシージャー - 無効なテキスト検索索引の削除

このプロシージャーは、無効なテキスト検索索引とそれに関連するコレクションをデータベースから削除します。表のすべての内容に影響するデータベース操作 (例えば、切り捨て) が実行されると、索引は無効になります。

このプロシージャーは、データベースを有効範囲とする場合に **db2ts CLEANUP FOR TEXT** コマンドの代替として機能します。インスタンス・レベルでの廃止コレクションを削除するには、コマンドを使用する必要があります。

## 許可

このステートメントの許可 ID が持つ特権に、DBADM および DATAACCESS 権限のある SYSTS\_ADM ロールが含まれている必要があります。

## デフォルトの PUBLIC 特権

なし

## 構文

```
▶▶SYSTS_CLEANUP(—options—,—message_locale—,—message—)▶▶
```

スキーマは SYSPROC です。

## プロシージャ・パラメーター

### *options*

タイプ VARCHAR(32K) の入力引数。 NULL または空パラメーターのみが受け入れられます。 IBM の内部使用のために予約済みです。

### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 次の例では、現行データベース内の無効なテキスト検索索引がすべてドロップされます。メッセージ言語が英語に設定され、プロシージャが成功すると、操作の成功を示す出力パラメーター・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_CLEANUP('', 'en_US', ?)
```

```
Value of output parameters
```

```
-----  
Parameter Name : MESSAGE
```

```
Parameter Value : Operation completed successfully.
```

```
Return Status = 0
```

例 2: エラーが戻される出力:

```
SQL0462W Command or routine "SYSTS_CLEANUP" (specific name "*N") has returned a warning SQLSTATE, with diagnostic text "CIE00212W 2 of 20 collections could not be deleted. Check db2diag.log for details.". SQLSTATE=01H14
```

## SYSTS\_CLEAR\_COMMANDLOCKS プロシージャ - テキスト検索索引のコマンド・ロックの削除

このプロシージャは、データベース内の特定のテキスト検索索引またはすべてのテキスト検索索引のすべてのコマンド・ロックを解除します。

### 許可

索引のロックをクリアするために使用されるステートメントの許可 ID が保持する特権には、以下の両方の権限が含まれている必要があります。

- SYSTS\_MGR ロール
- DBADM 権限またはその索引が定義されている基本表に対する CONTROL 特権

データベース接続のロックをクリアするために使用されるステートメントの許可 ID が保持する特権には、SYSTS\_ADM ロールが含まれている必要があります。

### デフォルトの PUBLIC 特権

なし

コマンド・ロックはテキスト検索索引コマンドの開始時に作成され、コマンドの完了時に破棄されます。それにより、異なるコマンド間の望ましくない競合が避けられます。まれなケースとして、予期しないシステム動作によってロックがそのまま残り、明示的にクリーンアップする必要があるときに、このプロシージャの使用が必要とされます。

このプロシージャは、データベース・サーバーで **CLEAR COMMAND LOCKS** テキスト検索管理コマンドを発行します。

### 構文

```
▶▶ SYSTS_CLEAR_COMMANDLOCKS ( (—index_schema—, —index_name—, —————▶▶  
▶—message_locale—, —message—) —————▶▶
```

スキーマは SYSPROC です。

### プロシージャ・パラメーター

#### *index\_schema*

テキスト索引のスキーマを指定する、タイプ VARCHAR(128) の入力引数。

*index\_schema* は、DB2 スキーマ名の命名上の制約に従う必要があります。引数が NULL または空ストリングである場合、値 CURRENT SCHEMA が使用されます。 *index\_schema* には、大文字小文字の区別があります。

#### *index\_name*

索引の名前を指定する、タイプ VARCHAR (128) の入力引数。 *index\_schema* とともに使用すると、データベースのテキスト検索索引を一意的に識別します。引数が NULL または空ストリングである場合、このプロシージャはデータベース内のすべてのテキスト検索索引のコマンド・ロックを削除します。

*index\_name* には、大文字小文字の区別があります。

### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 以下の例では、SYSTS\_CLEAR\_COMMANDLOCKS が、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex' があるテキスト検索索引に対して発行されます。エラー・メッセージは英語で戻ることが要求されます。このプロシージャが成功すると、操作の成功を示す出力パラメーター・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_CLEAR_COMMANDLOCKS('db2ts', 'myTextIndex', 'en_US', ?)
```

以下はこの照会の出力例です。

```
Value of output parameters
-----
Parameter Name  : MESSAGE
Parameter Value : Operation completed successfully.

Return Status = 0
```

例 2: 以下の例では、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex2' があるテキスト検索索引のコマンド・ロックをクリアするために、SYSTS\_CLEAR\_COMMANDLOCKS が呼び出されます。この索引は存在せず、プロシージャはエラー・メッセージを戻します。

```
CALL SYSPROC.SYSTS_CLEAR_COMMANDLOCKS('db2ts', 'myTextIndex2', 'en_US', ?)
```

以下はこの照会の出力例です。

```
SQL20427N An error occurred during a text search administration
procedure or command. The error message is "CIE00316 Text search
index "db2ts"."myTextIndex2" does not exist. ". SQLSTATE 38H14
```

## 使用上の注意

- テキスト検索管理プロシージャは、データベースへの既存の接続を使用します。テキスト検索管理プロシージャにおけるコミットまたはロールバックからの予期しない影響を避けるために、そのプロシージャを実行する前に、すべてのトランザクション変更をコミットすることをお勧めします。これを行う 1 つの方法は、AUTOCOMMIT をオンにすることです。
- コマンド・ロックを所有するプロセスが非活動であるため、このプロシージャを呼び出す場合があります。このケースでは、(ロックにより表される) コマンドが完了していない場合や、索引が操作できない場合があります。適切なアクションを取る必要があります。例えば、DROP INDEX コマンドを実行している処理が突然停止したとします。いくつかの索引データは削除されましたが、すべてのカタログおよびコレクション情報が削除された訳ではありません。コマンド・ロックは、そのまま残されています。DROP INDEX コマンド・ロックをクリアし

た後に、SYSTS\_DROP プロシージャを再実行することができます。別の例として、UPDATE INDEX コマンドを実行しているプロセスが中断したとします。一部の文書は処理済みであるものの、すべてが処理されたわけではなく、依然としてコマンド・ロックが有効になっています。テキスト検索索引の状況を検討し、UPDATE INDEX コマンド・ロックをクリアした後に、UPDATE INDEX コマンドを再実行できます。

- このプロシージャを実行すると、DB2 テキスト検索ビュー SYSIBMTS.TSLOCKS の内容は更新されます。

---

## SYSTS\_CLEAR\_EVENTS プロシージャ - 索引のイベント表からの索引付けイベントの削除

このプロシージャは、管理に使用される索引のイベント表から、索引付けイベントを削除します。

イベント表の名前は、列 EVENTVIEWNAME のビュー SYSIBMTS.TSINDEXES にあります。少なくとも 1 つの文書処理する索引の更新操作が行われるごとに、通知項目と、場合によってはエラー項目がイベント表に生成されます。自動更新の場合、イベント表は定期的に検査する必要があります。文書固有のエラーは、文書内容を変更することで訂正する必要があります。エラーを訂正した後、イベントをクリアすることができます (スペースを消費しすぎないために、クリアすべきです)。

このプロシージャは **CLEAR EVENTS FOR INDEX** テキスト検索管理コマンドをデータベース・サーバー上で発行します。

### 許可

このステートメントの許可 ID には、以下の両方の特権が含まれている必要があります。

- SYSTS\_MGR ロール
- その索引が定義されている表に対する DATAACCESS と DBADM を併せ持つ権限または CONTROL 特権

### デフォルトの PUBLIC 特権

なし

### 構文

```
▶▶ SYSTS_CLEAR_EVENTS ( ( index_schema , index_name , message_locale , message ) )
```

スキーマは SYSPROC です。

### プロシージャ・パラメーター

*index\_schema*

テキスト検索索引のスキーマを指定する、タイプ VARCHAR(128) の入力引数。 *index\_schema* は、DB2 スキーマ名の命名上の制約に従う必要があります。

す。引数が NULL または空ストリングである場合、値 CURRENT SCHEMA が使用されます。 *index\_schema* には、大文字小文字の区別があります。

#### *index\_name*

索引の名前を指定する、タイプ VARCHAR (128) の入力引数。 *index\_schema* とともに使用すると、データベースのテキスト検索索引を一意的に識別します。 *index\_name* には、大文字小文字の区別があります。

#### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

#### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 以下の例では、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex' で作成されたテキスト検索索引に対して SYSTS\_CLEAR\_EVENTS が呼び出されます。すべてのエラー・メッセージは英語で戻すことが要求されます。このプロシージャが成功すると、操作の成功を示す出力パラメーター・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_CLEAR_EVENTS('db2ts', 'myTextIndex', 'en_US', ?)
```

以下はこの照会の出力例です。

```
Value of output parameters
-----
Parameter Name : MESSAGE
Parameter Value : Operation completed successfully.

Return Status = 0
```

例 2: 以下の例では、SYSTS\_CLEAR\_EVENTS が呼び出され、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex2' があるテキスト検索索引のイベント表項目がクリアされます。この索引は存在せず、結果としてエラーを戻します。

```
CALL SYSPROC.SYSTS_CLEAR_EVENTS('db2ts', 'myTextIndex2', 'en_US', ?)
```

以下はこの照会の出力例です。

```
SQL20427N An error occurred during a text search administration
procedure or command. The error message is "CIE00316 Text search
index "db2ts"."myTextIndex2" does not exist. ". SQLSTATE 38H14
```

## 使用上の注意

- テキスト検索管理プロシージャは、データベースへの既存の接続を使用します。現行トランザクションは、プロシージャの完了結果に応じてコミットされるかまたはロールバックされる場合があります。そのため、そのようなコミットまたはロールバックからの予期しない影響を避けるために、すべてのトランザクション変更をコミットすることができます。これを行う 1 つの方法は、AUTOCOMMIT をオンにすることです。



- 複数のプロシージャーまたはコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。競合するプロシージャーおよびコマンドの一部は以下のとおりです。
  - SYSTS\_ALTER プロシージャーまたは ALTER INDEX db2ts コマンド
  - SYSTS\_DISABLE プロシージャーまたは DISABLE DATABASE FOR TEXT db2ts コマンド
  - SYSTS\_DROP プロシージャーまたは DROP INDEX db2ts コマンド
  - SYSTS\_CONFIGURE プロシージャー
  - SYSTS\_UPDATE プロシージャーまたは UPDATE INDEX db2ts コマンド
 競合がある場合、プロシージャーは SQLCODE -20426 および SQLSTATE 38H13 を戻します。
- 定期的な更新がスケジュールされている場合 (SYSTS\_CREATE または SYSTS\_ALTER プロシージャーの UPDATE FREQUENCY オプションを参照)、イベント表を定期的に検査する必要があります。
- テキスト検索索引の DB2 テキスト検索イベントをクリーンアップするには、イベントの理由を検査し、エラーの原因を除去した後に、SYSTS\_CLEAR\_EVENTS プロシージャーまたは **CLEAR EVENTS FOR INDEX** db2ts コマンドを使用します。
- イベント表で参照されるすべての行に変更が加えられていることを確認します。ユーザー表内の行を変更することで、SYSTS\_UPDATE プロシージャーまたは **UPDATE INDEX** db2ts コマンドを再度実行するときに、エラーがある文書に索引付けが再試行されます。
- このコマンドの発行時に、イベント表はクリアされます。

---

## SYSTS\_CONFIGURE プロシージャー - 現行データベースをテキスト検索用に構成する

**SYSTS\_CONFIGURE** プロシージャーは、テキスト検索サーバー接続情報をテキスト検索カタログに適用します。

テキスト検索サーバーの一部のプロパティを、テキスト検索管理表 (テキスト検索カタログ) に反映させる必要があります。これらのプロパティは、索引の作成時にテキスト索引に関連付けられます。トークンなどのパラメーターが定期的に更新される場合は、変更をデータベースに反映させて、すべての索引のプロパティを更新する必要もあります。

このプロシージャーは、以下の場合に最初に必要となります。

- 不完全な使用可能化
- スタンドアロン・テキスト検索サーバーのセットアップ
- パーティション・データベース
- および後続のテキスト検索サーバー接続情報の更新

後続の更新の場合には、アクティブなテキスト検索管理操作がないことを確認して、現在構成されているテキスト検索サーバーをシャットダウンしておく必要があります。

データベースの使用可能化のときに、統合テキスト検索サーバーの接続情報を使って SYSIBMTS.TSSERVER 管理ビューが更新されます。SYSIBMTS.TSSERVER ビューのテキスト・サーバー情報を確認し、該当するテキスト検索サーバー・データを使って更新した後、SYSTS\_CONFIGURE プロシージャを実行して更新済み情報を適用してください。インスタンス内にデータベースが複数ある場合は、同じテキスト検索サーバーの情報を使って各データベースを構成します。

一般的に、操作の順序は以下のようになります。

1. テキスト検索サーバーを構成します。インストール時または DB2 インスタンスの作成時に、統合テキスト検索サーバーを構成できます。スタンドアロン・テキスト検索サーバーは、別個に構成されます。
2. **db2ts ENABLE** コマンド、あるいは **ENABLE** オプションを指定した **SYSTS\_ENABLE** または **SYSTS\_ADMIN\_CMD** プロシージャを使用して、データベースをテキスト検索可能にします。
3. **SYSIBMTS.TSSERVERS** ビューでの必要に応じて、テキスト検索サーバーのパラメーターを決定します。
4. **SYSIBMTS.TSSERVERS** 管理ビューをテキスト検索サーバーのパラメーターで更新します。
  - ビューが空の場合、**INSERT** ステートメントを使用します。例:  

```
INSERT INTO SYSIBMTS.TSSERVERS (HOST, PORT, TOKEN, SERVERSTATUS)
VALUES ('localhost', 55000, '9kfsjg48=', 0);
```
  - ビューに既に行が含まれている場合は、**SQL UPDATE** ステートメントを使用します。例:  

```
UPDATE SYSIBMTS.TSSERVERS SET (HOST, PORT, TOKEN) =
('tsmach1.ibm.com', 55002, 'k3j4fjk9u=')
```
5. **SYSTS\_CONFIGURE** プロシージャを実行します。

## 許可

このステートメントの許可 ID が持つ特権に、**SYSTS\_ADM** ロールが含まれている必要があります。

## デフォルトの PUBLIC 特権

なし

## 構文

▶▶—SYSTS\_CONFIGURE—(—options—, —message-locale—, —message—)————▶▶

スキーマは **SYSPROC** です。

## プロシージャ・パラメーター

### *options*

使用するオプションを指定する、タイプ **VARCHAR(32K)** の入力引数。オプションを必要としない場合は、引数を **NULL** にするか、または空ストリングにすることができます。

### *message-locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、en\_US が使用されます。

### *message*

成功したと見なされる操作に対する警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: データベースをテキスト検索サービスのために準備し、すべてのメッセージを英語に戻します。

```
CALL SYSTS_ENABLE('', 'en_US', ?)"  
  
INSERT INTO SYSIBMTS.TSSERVERS  
  (HOST,PORT,TOKEN,KEY,LOCALE,SERVERTYPE,SERVERSTATUS)  
  VALUES ('tsmach1.ibm.com', 55000, '9kfsjg48=', 'en_US', 0);  
  
CALL SYSPROC.SYSTS_CONFIGURE('', 'en_US', ?)
```

この照会の出力例:

Value of output parameters

```
-----  
Parameter Name : MESSAGE  
Parameter Value : Operation completed successfully.
```

Return Status = 0

## 使用上の注意

- データベースがテキスト検索可能に設定されるとき、SYSIBMTS.TSSERVERS ビューが作成されます。このビューは、データベースの使用可能化時に統合テキスト検索サーバーについての情報で更新されます。テキスト検索サーバーの構成を取得できない場合、使用可能化の操作は「不完全な使用可能化」警告と共に終了します。
- SYSIBMTS.TSSERVERS で行の挿入や更新が行われた場合は、必ず SYSTS\_CONFIGURE プロシージャを発行する必要があります。SYSIBMTS.TSSERVERS を更新する前に、アクティブなテキスト検索管理操作がないことを確認して、テキスト検索サーバーをシャットダウンする必要があります。
- データベース内の SYSIBMTS.TSSERVERS を更新するときは、すべてのテキスト検索可能データベースを同じパラメーターで更新する必要があります。1 つの DB2 インスタンスでサポートされるテキスト検索サーバーは 1 つだけです。
- データベースに対して SYSTS\_CONFIGURE プロシージャを実行すると、インスタンスでの構成済みテキスト検索サーバーの使用が登録されます。このプロシージャを実行しない場合、重大エラーにはなりませんが、いくつかのコマンドに対して予期しない応答が発生することがあります。
- 複数のプロシージャまたはコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。競合するプロシージャおよびコマンドの一部は以下のとおりです。

- SYSTS\_ALTER プロシージャ
- SYSTS\_DISABLE プロシージャ
- SYSTS\_CONFIGURE プロシージャ
- SYSTS\_UPDATE プロシージャ

競合がある場合、プロシージャは SQLCODE -20426 および SQLSTATE 38H13 を戻します。

- テキスト検索のインストールと DB2 インスタンスのテキスト検索構成に関する一部の設定を更新する必要があります。これには、以下が含まれます。
  - DB2 インスタンスによって使われる検索サーバーが統合される (DB2 によって DB2 インスタンスの一部として構成される) のか、あるいは ECMTS サーバーの別個のスタンドアロン・インストールであるかの指示。
  - テキスト検索のセットアップでリッチ・テキスト・サポートが可能であるかどうかの指示。

---

## SYSTS\_CREATE プロシージャ - 列でのテキスト検索索引の作成

SYSTS\_CREATE プロシージャは、データベース・サーバー上で DB2 Text Search **CREATE INDEX** コマンドを発行して、テキスト列のテキスト検索索引を作成します。テキスト検索索引を作成して更新した後に、テキスト検索関数を使用して列データを検索できます。

索引更新操作が処理されるまでは、索引には何のデータも入れられません。ストアード・プロシージャ・インターフェースまたはコマンド・ライン・インターフェースのいずれかを使用して、更新操作を開始できます。後者の場合、DB2 Text Search の **UPDATE INDEX** コマンドを明示的に発行することができます。そうしない場合、索引に定義されている更新頻度に従って、DB2 管理タスク・スケジューラーによって、コマンドは暗黙的に発行されます。

### 許可

ステートメントの許可 ID には、SYSTS\_MGR ロールおよびデータベースに対する CREATETAB 権限と、以下のいずれかの権限または特権が含まれていなければなりません。

- その索引の定義されている表に対する CONTROL 特権。
- 索引が定義される表に対する INDEX 特権と、以下の権限のいずれか。
  - データベースに対する IMPLICIT\_SCHEMA 権限 (索引の暗黙または明示のスキーマ名が存在しない場合)
  - スキーマに対する CREATEIN 特権 (索引のスキーマ名が存在する場合)
- DBADM 権限

### デフォルトの PUBLIC 特権

なし

## 構文

```

▶▶ SYSTS_CREATE (—index_schema—, —index_name—, — text source | —, —————▶
▶ | options | —, —message_locale—, —message—) —————▶▶
  
```

### text source:

```

|—————| table-name (— | text column name | —), —————|
|—————|
| table-schema |
  
```

### text column name:

```

|—————|
|—————|
| column-name |
| function-name (column-name) |
  
```

### オプション:

```

|—————|
|—————|
| text default information |
|—————|
| update characteristics |
|—————|
| storage options |
|—————|
| index configuration options |
|—————|
| partition options |
  
```

### text default information:

```

|—————|
|—————|
| CODEPAGE—code-page | | LANGUAGE—language | | FORMAT—format |
|—————|
  
```

### update characteristics:

```

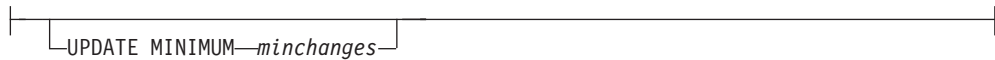
|—————|
|—————|
| UPDATE FREQUENCY—NONE— |
|—————|
| update frequency |
▶ | incremental update characteristics |—————▶
  
```

### update frequency:

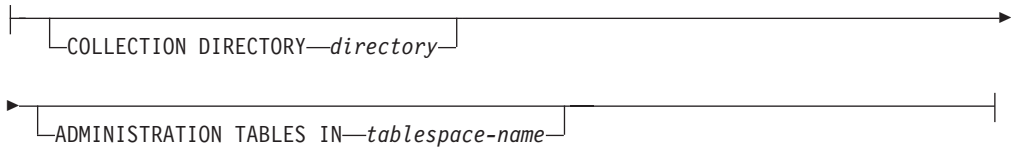
```

|—————|
|—————|
| D (— * —) — H (— * —) — M (— integer3 —) —————|
|—————|
|—————|
| integer1 |
|—————|
|—————|
| integer2 |
|—————|
|—————|
|—————|
|—————|
  
```

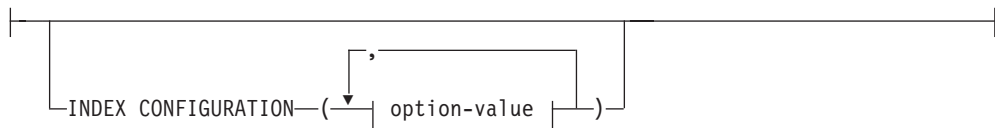
### incremental update characteristics:



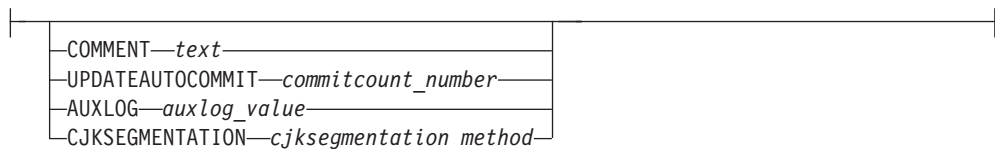
### storage options:



### index configuration options:



### option-value:



スキーマは SYSPROC です。

## プロシージャ・パラメーター

### *index\_schema*

テキスト検索索引のスキーマを指定する、タイプ VARCHAR(128) の入力引数。 *index\_schema* は、DB2 スキーマ名の命名上の制約に準拠する必要があります。引数が NULL または空ストリングである場合、スキーマのデフォルト値が使用されます。 *index\_schema* には、大文字小文字の区別があります。

### *index\_name*

索引の名前を指定する、タイプ VARCHAR (128) の入力引数。 *index\_name* を *index\_schema* と共に使用すると、データベースのテキスト検索索引を一意的に識別します。 *index\_name* には、大文字小文字の区別があります。

### text source

索引付けされた列の名前を指定する、タイプ VARCHAR (1024) の入力引数。オプションは以下のとおりです。

#### *table-schema*

テキスト検索索引が作成される表のスキーマ。

#### *table-name*

テキスト列がある表の名前を指定します。フェデレーテッド表、マテリアラ

イズ照会表、またはビューに対するテキスト検索索引を作成することはできません。*table-name* には、大文字小文字の区別があります。

#### **text column name**

索引を付ける列の名前を指定します。

##### *column-name*

索引を付ける列の名前を指定します。列は、CHAR、VARCHAR、CLOB、DBCLOB、BLOB、GRAPHIC、VARGRAPHIC、または XML のデータ・タイプのいずれかでなければなりません。列のデータ・タイプがこれらのいずれのデータ・タイプでもない場合、*function-schema.function-name* という名前の変換関数を使用して、列タイプを有効なタイプのいずれかに変換できます。(詳しくは、*function-name (column-name)* パラメーターを参照してください。)あるいは、索引付けされるテキスト文書にアクセスするユーザー定義の外部関数を指定することができます。1 つの列に作成できるテキスト検索索引は 1 つのみです。*column-name* には、大文字小文字の区別があります。

##### *function-name (column-name)*

テキスト検索用にサポートされていない列データ・タイプを、テキスト検索用にサポートされているデータ・タイプに変換する、外部スカラー関数のスキーマ修飾名を指定します。この関数は、1 つのパラメーターのみを取り、1 つの値のみを戻す必要があります。関数の名前は、DB2 命名規則に準拠していなければなりません。*function-name(column-name)* パラメーターには大文字小文字の区別があります。

#### **オプション**

使用するオプションを指定する、タイプ VARCHAR(32) の入力引数。オプションを必要としない場合は、このパラメーターを NULL にするか、または空ストリングにすることができます。オプションは以下のとおりです。

##### **CODEPAGE** *code-page*

テキスト文書を索引付けするときに使用される DB2 コード・ページを指定します。デフォルトのコード・ページ値は、SYSIBMTS.TSDEFAULTS ビューの、DEFAULTNAME 列値が CODEPAGE の行から取られます。新規索引の初期デフォルトのコード・ページは、データベース・コード・ページです。このパラメーターはバイナリー・データ・タイプにのみ適用されます。つまり列タイプや変換関数からの戻りの型は、BLOB または文字タイプの FOR BIT DATA でなければなりません。

##### **LANGUAGE** *language*

索引付け中に、文書の言語固有の処理のために DB2 Text Search によって使用される言語を指定します。ロケールを指定しない場合、データベース・テリトリーを使用して **LANGUAGE** パラメーターのデフォルト設定が判別されます。ロケールを判別するために文書をスキャンさせる場合は、*locale* を AUTO と指定してください。

##### **FORMAT** *format*

列内のテキスト文書のフォーマットを指定します。サポートされているフォーマットは TEXT、XML、HTML、および INSO です。列のデータ・タイプが XML でない場合は、デフォルトのフォーマットは、SYSIBMTS.TSDEFAULTS ビューの、DEFAULTNAME 列値が **FORMAT** の

行から取られます。列のデータ・タイプが XML の場合は、デフォルトのフォーマットは常に XML です。INSO フォーマットを使用する場合は、リッチ・テキスト・サポートをインストールしなければなりません。

#### UPDATE FREQUENCY

索引の更新頻度を指定します。変更の数が **UPDATE MINIMUM** パラメーターに設定された値以上になると、索引が更新されます。自動更新を行えるのは、**DB2\_ATS\_ENABLE** レジストリー変数が設定されていて、**START FOR TEXT** コマンドが発行される場合のみです。

デフォルトのフォーマットは、**SYSIBMTS.TSDEFAULTS** ビューの、**DEFAULTNAME** 列値が **UPDATEFREQUENCY** の行から取られます。

#### NONE

これ以上の索引更新は行われません。この値は、変更されないデータを持つ表のテキスト列に役立ちます。さらに、**UPDATE INDEX** コマンドを使用して手動で索引を更新する予定になっている場合にも役立ちます。

#### D 索引が更新される曜日。

\* 毎日。

*integer1*

日曜日から土曜日の特定の曜日: 0 から 6。

#### H 索引が更新される指定日の時。

\* 毎時。

*integer2*

午前 0 時から午後 11 時までの特定の時: 0 から 23。

#### M 索引が更新される指定時の分。

*integer3*

#### UPDATE MINIMUM *minchanges*

**UPDATE FREQUENCY** パラメーターで指定された時間に索引が増分的に更新される前に、テキスト文書に加えられている必要のある変更の最小数を指定します。正の整数値のみが許可されます。デフォルト値は、

**SYSIBMTS.TSDEFAULTS** ビューの、**DEFAULTNAME** 列値が

**UPDITEMINIMUM** の行から取られます。値が小さい場合は、表列とテキスト検索索引の間の整合性が増しますが、システムの負荷も増えます。

**UPDATE INDEX** コマンドは、**USING UPDATE MINIMUM** パラメーターが指定されない限りこのパラメーターの値を無視します。

#### COLLECTION DIRECTORY *directory*

テキスト検索索引のコレクションが保管されるディレクトリーを指定します。絶対パスを指定する必要があり、絶対パス名の最大長は 215 文字です。Text Search サーバー・インスタンス・サービスのプロセス所有者は、このディレクトリーに対する読み取りおよび書き込み権限を持っている必要があります。

**COLLECTION DIRECTORY** パラメーターは、統合テキスト検索サーバーのセットアップでのみサポートされています。コレクションの場所に関する追加情報については、使用上の注意を参照してください。



## ADMINISTRATION TABLES IN *table-space-name*

索引用に作成された管理表の既存の表スペース (TEMPORARY 表スペース以外) の名前を指定します。デフォルトでは、索引を作成する基本表の表スペースが使用されます。

パーティション表またはパーティション・データベース内の表に対してテキスト索引を作成する場合、この引数は必須です。パーティション・データベースの場合、この表スペースは基本表の表スペースと同じパーティション・グループでなければなりません。テキスト索引の管理表は、対応する基本表と同じように配分されます。

## INDEX CONFIGURATION (*option-value*)

追加の索引関連オプションを、オプションと値のペアとして指定します。以下のオプションがサポートされています。

表 14. INDEX CONFIGURATION パラメーターのオプションと値のペア

オプション	値	データ・タイプ	説明
COMMENT	<i>text</i>	512 バイト未満のストリング値	DB2 Text Search の SYSIBMTS.TSINDEXES カタログ・ビューにある REMARKS 列にストリング・コメントを追加します。コメントをコレクションの説明としても使用します。
UPDATEAUTOCOMMIT	<i>commitcount_number</i>	整数	ここで指定された回数の索引更新が行われると、コミットが実行され、イニシャル更新またはインクリメンタル更新としてそれまでの作業が保存されます。 <ul style="list-style-type: none"><li>イニシャル更新では、データ更新をキャプチャーするトリガーがアクティブ化された後の基本表からの文書のバッチを、索引更新が処理します。更新された文書の数 <i>commitcount_number</i> の数に達すると、サーバーは暫定コミットを実行します。処理されていない文書により生成されたログ項目は、ステージング表から除去されます。イニシャル・テキスト索引更新に UPDATEAUTOCOMMIT オプションを使用すると、実行時間が大幅に増加します。</li><li>インクリメンタル更新では、処理されたログ項目は、各暫定コミットでステージング表から除去されます。<i>commitcount_number</i> は更新された文書の数であり、ステージング表項目の数ではありません。</li></ul>

表 14. INDEX CONFIGURATION パラメーターのオプションと値のペア (続き)

オプション	値	データ・タイプ	説明
AUXLOG	<i>auxlog_value</i>	ストリング	トリガーに認識されない変更をキャプチャーするための、追加のログ・インフラストラクチャーの作成を制御します。範囲パーティション表のデフォルト設定は ON です。非範囲パーティション表の場合は <i>AuxLogNorm</i> を設定し、範囲パーティション表の場合は <i>AuxLogPart</i> を設定して、デフォルト表内のデフォルト値を変更できます。  索引の作成後に、テキスト索引の補助ログ・インフラストラクチャー・プロパティを変更することはできません。
CJKSEGMENTATION	<i>CJKSEGMENTATION_method</i>	ストリング	中国語、日本語、および韓国語 (zh_CN、zh_TW、ja_JP、ko_KR ロケール・セット) の文書に関するセグメンテーション方式を適用できます。自動言語検出が使用可能になっている (LANGUAGE AUTO) 場合も行われます。オプションを指定しないと、デフォルト表内の CJKSEGMENTATION の値が適用されます。サポートされる値は、以下のとおりです。  <ul style="list-style-type: none"> <li>• MORPHOLOGICAL</li> <li>• NGRAM</li> </ul> 指定されたセグメンテーション方式は <i>SYSIBMTS.TSCONFIGURATION</i> 管理ビューに追加され、テキスト索引の作成後に変更することはできません。

**要確認:** コメントなどの非数値は、単一引用符で囲む必要があります。ストリング値内の単一引用符の文字は、2 つの連続した単一引用符によって表記される必要があります。

**例:** INDEX CONFIGURATION (COMMENT 'Index on User''s Guide column')

*message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

## message

正常に完了した操作の警告メッセージまたは通知メッセージを指定する、タイプ VARCHAR(32) の出力引数。

## 例

例 1: 以下の例では、myTextIndex と呼ばれるテキスト検索索引を作成するために、SYSTS\_CREATE プロシージャを呼び出します。UPDATE MINIMUM 10 パラメータは、索引に関連付けられたテキスト文書に 10 の変更が加えられてから、索引のインクリメンタル更新が実行されることを指定します。すべてのエラー・メッセージは英語で戻されます。基本となるテキスト検索コマンドが正常に実行された場合は、message 出力パラメーターがコマンド実行の状況を示すように設定されます。

```
CALL SYSPROC.SYSTS_CREATE('db2ts', 'myTextIndex',
    'myUserSchema.myBaseTable (myTextColumn)', 'UPDATE MINIMUM 10',
    'en_US', ?)
```

出力例を以下に示します。

```
Value of output parameters
-----
Parameter Name : MESSAGE
Parameter Value : Operation completed successfully.
Return Status = 0
```

例 2: 以下の例では、myTextIndex2 と呼ばれるテキスト検索索引を作成するために、SYSTS\_CREATE プロシージャを呼び出します。オプションが指定されていません。この例では、索引が存在しており、結果としてエラー・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_CREATE('db2ts', 'myTextIndex2',
    'myUserSchema.myBaseTable (myTextColumn)', '', 'en_US', ?)
```

出力例を以下に示します。

```
SQL20427N An error occurred during a text search administration
procedure or command. The error message is "CIE00201 Text search
index "db2ts"."myTextIndex2" already exists. "
```

## 使用上の注意

- テキスト検索管理プロシージャは、データベースへの既存の接続を使用します。テキスト検索管理操作を実行する前に、直前のトランザクションがコミットされなかった場合、デッドロックになる場合があります。これは、同じデータベース・オブジェクトに影響が及ぶので、直前のステップがコミットされるまで操作待機するためです。例えば、AUTOCOMMIT がオフになっている場合、表を作成してからテキスト索引を作成する際に 2 つのトランザクションの間で明示的なコミットをしないと、このデッドロックが生じる可能性があります。
- COLLECTION DIRECTORY 節を指定しないと、Text Server 構成内で defaultDataDirectory パラメーターを使用して定義したパス内の、システム生成の索引 ID にちなんでも命名されたサブディレクトリ内にコレクションが入れられます。configTool ユーティリティを使用して、このパラメーターを明示的に構成することができます。defaultDataDirectory パラメーターが明示的に構成されていない場合には、コレクションのサブディレクトリは、<configPath>/config/collections ディレクトリ内に配置されます。ホーム・ディレクトリやデータベース・インスタンス・パス以外の場所を

*defaultDataDirectory* や **COLLECTION DIRECTORY** として使用することを強くお勧めします。この場所に適切なストレージ・スペースがあり、しかもこの場所がローカルである (NFS にマウントされていない) ことを確認してください。

- **COLLECTION DIRECTORY** パラメーターの値を構成すると、ストライピングされた RAID デバイスの使用が可能になります。

キーに関連した以下の規則が適用されます。

- 表の主キーを定義する必要があります。
- 主キー列の数は、他の DB2 環境で許可されている主キー列の数より 2 列少ない数に制限されています。
- DB2 Text Search 索引のある表のすべての主キー列の合計長は、他の DB2 環境で許可されている主キーの最大合計長より 15 バイト少ない長さに制限されています。DB2 CREATE INDEX ステートメントに関する制約事項を参照してください。

**LANGUAGE** パラメーターを **AUTO** オプションに設定して索引を作成すると、**CJKSEGMENTATION** 指定がオプションとして許可されます。指定されたセグメンテーション方式は、中国語、日本語、および韓国語の文書に適用されます。いったん索引作成が完了したら、*CJKSEGMENTATION\_method* 値の値セットを変更することはできません。

特定のプロシージャまたはコマンドは、操作が競合するタイミングによってはエラー・メッセージが出される可能性があるため、テキスト検索索引に対して同時に実行することはできません。

- **SYSTS\_DISABLE** プロシージャまたは **db2ts DISABLE DATABASE FOR TEXT** コマンド
- **SYSTS\_CONFIGURE** プロシージャ

**CREATE INDEX** コマンドが正常に実行されると、以下の効果が得られます。

- DB2 Text Search のサーバー・データが更新されます。以下の例のように、*instance\_database-name\_index-identifier\_number* という名前のコレクションが作成されます。

```
tigertail_MYTSDB_TS250517_0000
```

パーティション・データベースの場合、コレクションはパーティションごとに作成されます。**SYSIBMTS.TSCOLLECTIONNAMES** ビューの **COLLECTIONNAME** 列からコレクション名を取得できます。

- 索引のイベント表が、指定された表スペースに作成されます。さらに、索引のステージング表が、適切な DB2 索引のある指定された表スペースに作成されます。**INDEX CONFIGURATION** パラメーターの **AUXLOG ON** オプションを設定した場合には、保全性処理による変更をキャプチャーするために、2 目目のステージング表が作成されます。
- DB2 Text Search のカタログ情報が更新されます。しかし、新しく作成されたテキスト検索索引には、自動的にデータが追加されません。テキスト検索索引にデータを追加するには、**SYSTS\_UPDATE** プロシージャまたは **UPDATE INDEX** コマンドを手動で、または (**UPDATE FREQUENCY** パラメーターを使用して索引の更新スケジュールを定義した結果として) 自動で発行する必要があります。

- DB2 Text Search と DB2 Net Search Extender が共に存在し、表列のアクティブ Net Search Extender 索引が存在している場合、新しい DB2 Text Search 索引は非アクティブに設定されます。
- **LANGUAGE** パラメーターを AUTO に設定し、**CJKSEGMENTATION** を MORPHOLOGICAL に設定して、テキスト検索索引を作成した場合、形態素索引で有効なストリングを検索すると、ゼロ行が戻されます。結果を取得するには、以下の照会のように **CONTAINS** 関数に **QUERYLANGUAGE** オプションを追加します。

```
select bookname from morphobooks
where contains (story, '■■■','QUERYLANGUAGE=zh_CN') = 1
```

---

## SYSTS\_DISABLE プロシージャ - テキスト検索の現行データベースを使用不可にする

このプロシージャは、現行データベースの DB2 テキスト検索を使用不可にします。

テキスト検索機能を使用不可にすると、テキスト検索索引およびコマンドは、データベースで使用できなくなります。

このプロシージャは **DISABLE DATABASE FOR TEXT** テキスト検索管理コマンドをデータベース・サーバー上で発行します。

### 許可

このステートメントの許可 ID には、以下の特権が含まれている必要があります。

- DBADM と DATAACCESS を併せ持つ権限。
- SYSTS\_ADM ロール。

### デフォルトの PUBLIC 特権

なし

### 構文

```
►►SYSTS_DISABLE(—options—,—message_locale—,—message—)◄◄
```

スキーマは SYSPROC です。

### プロシージャ・パラメーター

#### *options*

データベースが使用不可の場合に使用するオプションを指定する、タイプ VARCHAR(128) の入力引数。引数は FORCE に設定できます。この値を指定した場合、すべての索引はドロップされ、テキスト検索機能は強制的に使用不可にされます。テキスト検索索引は保存されず、エラー・メッセージまたは警告は戻されません。引数が NULL または空ストリングである場合、データベースのテキスト検索機能を使用不可にすることが試行されます。

#### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ

VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

#### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 以下の例では、テキスト検索は、SYSTS\_DISABLE プロシージャを使用してデータベースに対して使用不可にされます。FORCE オプションを指定すると、テキスト検索索引が引き続きデータベース内の表に存在しているとしても、この機能を確実に使用不可にできます。エラー・メッセージは英語で戻ることが指定されます。message 出力パラメーターは、情報メッセージ・ストリングに設定されません。

```
CALL SYSPROC.SYSTS_DISABLE('FORCE', 'en_US', ?)
```

以下は、この照会からの出力例です。

```
Value of output parameters
-----
Parameter Name   : MESSAGE
Parameter Value  : Operation completed successfully.

Return Status = 0
```

例 2: 以下の例では、FORCE オプションを指定せずに SYSTS\_DISABLE プロシージャを使用して、テキスト検索を既存のテキスト検索索引があるデータベースに対して使用不可にします。これは結果としてエラー・メッセージを呼び出し元に戻します。テキスト検索機能を使用不可にするか、またはその代わりに options 入力パラメーター値に FORCE オプションを指定する前に、すべての既存のテキスト検索索引をドロップすることをお勧めします。

```
CALL SYSPROC.SYSTS_DISABLE('', 'en_US', ?)
```

以下は、この照会からの出力例です。

```
SQL20427N An error occurred during a text search administration
procedure or command. The error message is "CIE00326 Text search
index active in specified or default database. ". SQLSTATE 38H14
```

## 使用上の注意

- テキスト検索管理プロシージャは、データベースへの既存の接続を使用します。テキスト検索管理プロシージャにおけるコミットまたはロールバックからの予期しない影響を避けるために、そのプロシージャを実行する前に、すべてのトランザクション変更をコミットすることをお勧めします。これを行う 1 つの方法は、AUTOCOMMIT をオンにすることです。
- 複数のプロシージャまたはコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。競合するプロシージャのいくつかは、以下のとおりです。
  - SYSTS\_ALTER プロシージャ
  - SYSTS\_CLEAR\_EVENTS プロシージャ
  - SYSTS\_DISABLE プロシージャ

- SYSTS\_CONFIGURE プロシージャ
- SYSTS\_UPDATE プロシージャ

競合がある場合、プロシージャは SQLCODE -20426 および SQLSTATE 38H13 を戻します。

**注: SYSTS\_DISABLE** プロシージャに関するロックが、データベース・レベルで設定されます。

- このプロシージャが実行されると、
  - DB2 Text Search のカタログ情報が更新されます。索引ログおよびイベント表がドロップされます。ユーザー・テキスト表のトリガーが削除されます。
  - **FORCE** オプションを指定した場合、すべてのテキスト索引情報はデータベースから削除され、すべての関連する集合も削除されます。詳しくは、『db2ts DROP INDEX コマンド』または『SYSTS\_DROP プロシージャ』を参照してください。
- このプロシージャは、データベースの DB2 Net Search Extender 使用可能化状況には影響を与えません。これは、SYSTS\_ENABLE プロシージャまたは **ENABLE FOR TEXT** コマンドにより作成された DB2 テキスト検索カタログ表およびビューを削除します。
- テキスト検索索引定義がある DB2 データベースをドロップする前に、このプロシージャを実行して、テキスト索引および集合が正常に削除されていることを確認してください。
- **FORCE** オプションを使用して、いくつかの索引が削除できなかった場合、コレクション名が **db2diag** ログ・ファイルに書き込まれます。

**注:** 結果がオーファン集合 (つまり残された集合がテキスト検索サーバー上に定義された状態であるが、DB2 には使用されない) となる使用法は推奨されていません。以下は、孤立したコレクションの原因となる可能性のあるいくつかのケースです。

- DISABLE DATABASE FOR TEXT コマンドを実行せずに、DROP DATABASE CLP コマンドが実行される場合。
- **FORCE** オプションを使用して、SYSTS\_DISABLE プロシージャまたは **DISABLE DATABASE FOR TEXT** コマンドが実行される場合。

---

## SYSTS\_DROP プロシージャ - テキスト検索索引のドロップ

SYSTS\_DROP プロシージャは、表列と関連付けられた既存のテキスト検索索引をドロップします。

このプロシージャが正常に実行した後は、テキスト検索索引がドロップされた列に対してテキスト検索照会を実行することはできません。

このプロシージャは **DROP INDEX** テキスト検索管理コマンドをデータベース・サーバー上で発行します。

## 許可

ステートメントの許可 ID によって保持されている特権には、SYSTS\_MGR ロールと、少なくとも以下のいずれかの特権または権限が含まれていなければなりません。

- その索引の定義されている表に対する CONTROL 特権。
- その索引の定義されているスキーマに対する DROPIN 特権。
- テキスト検索索引に既存のスケジュールがある場合、許可 ID は索引の作成者と同じであるか、DBADM 権限が含まれている必要があります。

## デフォルトの PUBLIC 特権

なし

## 構文

```
▶▶ SYSTS_DROP (—index_schema—, —index_name—, —options—)
▶▶ —message_locale—, —message—) ▶▶
```

スキーマは SYSPROC です。

## プロシージャ・パラメーター

### *index\_schema*

テキスト検索索引のスキーマを指定する、タイプ VARCHAR(128) の入力引数。 *index\_schema* は、DB2 スキーマ名の命名上の制約に従う必要があります。引数が NULL または空ストリングである場合、値 CURRENT SCHEMA が使用されます。 *index\_schema* には、大文字小文字の区別があります。

### *index\_name*

索引の名前を指定する、タイプ VARCHAR (128) の入力引数。 *index\_schema* とともに使用すると、データベースのテキスト検索索引を一意的に識別します。 *index\_name* には、大文字小文字の区別があります。

### *options*

使用するオプションを指定する、タイプ VARCHAR(32000) の入力引数。オプションを必要としない場合は、引数を NULL にするか、または空ストリングにすることができます。

### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。



## 例

例 1: 以下の例では、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex' で作成されたテキスト検索索引がドロップされます。すべてのエラー・メッセージは英語で戻ることが要求されます。このプロシージャが成功すると、操作の成功を示す出力パラメーター・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_DROP('db2ts', 'myTextIndex', '', 'en_US', ?)
```

前のリリースと同様に **SYSTS\_DROP** プロシージャは、**options** 引数を指定しなくてもサポートされます。以下に例を示します。

```
CALL SYSPROC.SYSTS_DROP('db2ts', 'myTextIndex', 'en_US', ?)
```

以下はこの照会の出力例です。

```
Value of output parameters
-----
Parameter Name : MESSAGE
Parameter Value : Operation completed successfully.

Return Status = 0
```

例 2: 以下の例では、**SYSTS\_DROP** が呼び出され、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex2' があるテキスト検索索引がドロップされます。この索引は存在せず、結果としてエラーを戻します。

```
CALL SYSPROC.SYSTS_DROP('db2ts', 'myTextIndex2', 'en_US', ?)
```

以下はこの照会の出力例です。

```
SQL20427N An error occurred during a text search administration
procedure or command. The error message is "CIE00316 Text search
index "db2ts"."myTextIndex2" does not exist.". SQLSTATE 38H14
```

## 使用上の注意

- 複数のプロシージャまたはコマンドは、競合する可能性がある場合は、テキスト検索索引に対して同時に実行することはできません。以下に、競合するプロシージャをいくつか示します。

- **SYSTS\_ALTER** プロシージャ
- **SYSTS\_CLEAR\_EVENTS** プロシージャ
- **SYSTS\_DISABLE** プロシージャ
- **SYSTS\_DROP** プロシージャ
- **SYSTS\_UPDATE** プロシージャ
- **SYSTS\_CONFIGURE** プロシージャ

**DROP** 操作と並行して実行する **STOP FOR TEXT** コマンドでは、競合するコマンド・メッセージは発生しませんが、代わりに、**DROP** がコレクションを除去する前にテキスト検索サーバーがシャットダウンされる場合は、テキスト検索サーバーが使用不可であるというエラーが戻されます。

- テキスト検索索引がドロップされた後、テキスト検索は対応するテキスト列で使用できなくなります。同じテキスト列に新規のテキスト検索を作成する予定の場合、まずデータベースから切断し、再接続してから、新規のテキスト検索索引を作成する必要があります。
- db2ts DROP INDEX** コマンドは、データベースに以下の変更を行います。

- DB2 Text Search のカタログ情報を更新します。
- 索引ステージング表およびイベント表をドロップします。
- ユーザー・テキスト表のトリガーを削除します。
- DB2 Text Search 索引定義に関連付けられたコレクションを破棄します。

## SYSTS\_ENABLE プロシージャ - 現在のデータベースをテキスト検索用に使用可能にする

SYSTS\_ENABLE プロシージャは、現行データベースの DB2 Text Search を使用可能にします。

このプロシージャは、データベース内の列に対してテキスト検索索引を作成する前に、正常に実行する必要があります。

このプロシージャは **ENABLE DATABASE FOR TEXT** テキスト検索管理コマンドをデータベース・サーバー上で発行します。

### 許可

プロシージャの許可 ID によって保持されている特権には、SYSTS\_ADM ロールと DBADM 権限が含まれていなければなりません。

### デフォルトの PUBLIC 特権

なし

### 構文

```
»»SYSTS_ENABLE(—options—,—message_locale—,—message—)««
```

オプション:

```
»»┌──────────────────────────────────────────────────────────────────────────────────┐
│ ADMINISTRATION TABLES IN tablespace-name │
└──────────────────────────────────────────────────────────────────────────────────┘««
```

スキーマは SYSPROC です。

### プロシージャ・パラメーター

#### options

使用するオプションを指定する、タイプ VARCHAR(32K) の入力引数。オプションを必要としない場合は、引数を NULL にするか、または空ストリングにすることができます。サポートされる値は以下のとおりです。

#### ADMINISTRATION TABLES IN *table-space-name*

データベースを DB2 Text Search に使用可能にする際に作成される管理表のための、既存の REGULAR 表スペースの名前を指定します。この節が指定されない場合には、SYSTOOLSPACE が表スペースとして使用されます。

SYSTOOLSPACE を使用するのか、明示的に指定された表スペース名を使用するのにかかわらず、表スペースは特定の要件を満たしている必要があります。要件は次のとおりです。

- REGULAR 表スペース
- パーティション・データベースの場合、バッファ・プールおよびページ・サイズが 32KB の表スペースを定義することをお勧めします。

大/小文字の区別のある表スペース名を使用するには、名前を二重引用符で囲んで、区切り ID を使用します。デフォルトでは、通常 ID として扱われ、大文字に変換されます。

#### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

#### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 表スペース内に管理表と英語の出力メッセージを作成して、データベースをテキスト検索に使用できるようにします。

```
CALL SYSPROC.SYSTS_ENABLE ('ADMINISTRATION TABLES IN TSSPACE', 'en_US', ?)
```

表スペース内に管理表とフランス語の出力メッセージを作成して、データベースをテキスト検索に使用できるようにします。

```
CALL SYSPROC.SYSTS_ENABLE ('ADMINISTRATION TABLES IN "tbs32k" ', 'fr_FR', ?)
```

前のリリースと同様に SYSTS\_ENABLE プロシージャは、**options** 引数を指定しなくてもサポートされます。以下に例を示します。

```
CALL SYSPROC.SYSTS_ENABLE ('en_US', ?)
```

この照会の出力例:

```
Value of output parameters
-----
Parameter Name : MESSAGE
Parameter Value : Operation completed successfully.

Return Status = 0
```

例 2: 以下の例では、SYSTS\_ENABLE がテキスト検索用に既に使用可能にされているデータベース上で呼び出されます。これは結果としてエラー・メッセージを呼び出し元に戻します。

```
CALL SYSPROC.SYSTS_ENABLE('en_US', ?)
```

この照会の出力例:

```
SQL20427N An error occurred during a text search administration
procedure or command. The error message from the text search
product is "CIE00322 Specified or default database already
enabled for text. ". SQLSTATE 38H14
```

## 使用上の注意

- テキスト検索管理プロシージャは、データベースへの既存の接続を使用します。テキスト検索管理プロシージャ内のコミットまたはロールバックからの予期しない影響を避けるため、プロシージャを実行する前に、すべてのトランザクション変更をコミットすることをお勧めします。これを行う 1 つの方法は、AUTOCOMMIT をオンにすることです。
- このプロシージャが実行されると、次のイベントが発生します。
  - このプロシージャは、スキーマ SYSIBMTS にテキスト検索管理カタログ表およびビューなどのデータベース・オブジェクトを作成します。
  - テキスト検索索引の設定済みデータベース・デフォルトは、ビュー SYSIBMTS.TSDEFAULTS で入手可能です。
  - コマンドが正常に完了したら、テキスト検索カタログ表およびビューが作成され、使用可能になります。
- このプロシージャを実行する場合、Text Search サーバー構成を取得するための十分な実行およびファイルアクセス特権がなければ、プロシージャはテキスト検索カタログを「不完全な使用可能化」警告とともに作成します。この場合は、Text Search サーバー接続情報を手動で更新する必要があります。

---

## SYSTS\_UPDATE プロシージャ - テキスト検索索引の更新

**SYSTS\_UPDATE** プロシージャは、索引が関連付けられるテキスト列の現在の内容を反映するテキスト検索索引を更新します。

更新が実行されている間も、検索は可能です。更新が完了するまでは、検索は部分的に更新された索引に対して実行されます。

このプロシージャは **UPDATE INDEX** テキスト検索管理コマンドをデータベース・サーバー上で発行します。

### 許可

ステートメントの許可 ID が保持する特権には、SYSTS\_MGR ロールと以下の少なくとも 1 つの権限が含まれている必要があります。

- DATAACCESS 権限
- そのテキスト索引の定義されている表に対する CONTROL 特権。
- テキスト索引が定義されている基本表に対する、INDEX 特権および SELECT 特権。

さらに、イニシャル更新では、許可要件が **CREATE TRIGGER** ステートメントで概説されているように適用されます。

### デフォルトの PUBLIC 特権

なし

### 構文

►►SYSTS\_UPDATE(—*index\_schema*—,—*index\_name*—,——————►

スキーマは SYSPROC です。

## プロシージャ・パラメーター

### *index\_schema*

テキスト検索索引のスキーマを指定する、タイプ VARCHAR(128) の入力引数。 *index\_schema* は、DB2 スキーマ名の命名上の制約に従う必要があります。引数が NULL または空ストリングである場合、値 CURRENT SCHEMA が使用されます。 *index\_schema* には、大文字小文字の区別があります。

### *index\_name*

索引の名前を指定する、タイプ VARCHAR (128) の入力引数。 *index\_schema* とともに使用すると、データベースのテキスト検索索引を一意的に識別します。 *index\_name* には、大文字小文字の区別があります。

### *update\_options*

更新オプションを指定する、タイプ VARCHAR(32K) の入力引数。オプションを必要としない場合は、引数を NULL にするか、または空ストリングにすることができます。可能な値は次のとおりです。

- USING UPDATE MINIMUM: この値は、 **CREATE INDEX** テキスト検索管理コマンドおよび SYSTS\_CREATE プロシージャからの UPDATE MINIMUM 設定を受け入れます。
- FOR DATA REDISTRIBUTION: データ・パーティションが追加または削除され、後続のデータ再配分操作が完了した後に、パーティション・データベース内のテキスト検索索引をリフレッシュする必要があることを指定します。検索結果は、テキスト検索索引が FOR DATA REDISTRIBUTION オプションで更新されるまでは、不整合である場合があります。
- ALLROWS: イニシャル更新を無条件に試行することを指定します。
- NULL または空ストリング (""): プロシージャの呼び出し時に更新が無条件に開始されることを指定します。

### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 以下の例では、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex' で作成されたテキスト検索索引が更新されます。 *update\_options* が NULL 値であることは、ストアード・プロシージャの呼び出し時に更新が無条件で開始されることを意味します。すべてのエラー・メッセージは英語で戻ることが要求されます。このプロシージャが成功すると、操作の成功を示す出力パラメーター・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_UPDATE
('db2ts', 'myTextIndex', '', 'en_US', ?)
```

この照会からの出力例:

```
Value of output parameters
-----
Parameter Name : MESSAGE
Parameter Value : Operation completed successfully.
```

例 2: 基本表に関連付けられているパーティション・グループ上で操作が実行された後、テキスト索引を更新し、エラー・メッセージを英語で戻します。

```
CALL SYSPROC.SYSTS_UPDATE
('db2ts', 'myTextIndex2', 'FOR DATA REDISTRIBUTION', 'en_US', ?)
```

```
Value of output parameters
-----
Parameter Name : MESSAGE
Parameter Value : Operation completed successfully.
```

Return Status = 0

例 3: 以下の例では、SYSTS\_UPDATE が呼び出され、*index\_schema* 'db2ts' および *index\_name* 'myTextIndex3' があるテキスト検索索引が更新されます。この索引は存在せず、結果としてエラーを戻します。

```
CALL SYSPROC.SYSTS_UPDATE('db2ts', 'myTextIndex3', 'USING UPDATE MINIMUM',
'en_US', ?)
```

```
SQL20427N An error occurred during a text search administration
procedure or command. The error message is "CIE00316 Text search
index "db2ts"."myTextIndex3" does not exist. ". SQLSTATE 38H14
```

## 使用上の注意

- テキスト検索管理プロシージャは、データベースへの既存の接続を使用します。現行トランザクションは、プロシージャの完了結果に応じてコミットされるかまたはロールバックされる場合があります。そのようなコミットまたはロールバックからの予期しない影響を避けるために、すべてのトランザクション変更をコミットすることができます。すべてのトランザクション変更をコミットする方法の 1 つとして、AUTOCOMMIT をオンにする方法があります。
- 特定のプロシージャまたはコマンドは、テキスト検索索引に対して同時に実行することはできません。これは、競合する操作のタイミングによってエラーが発生する可能性があるためです。競合するプロシージャおよびコマンドの一部は以下のとおりです。

- SYSTS\_ALTER プロシージャまたは **db2ts ALTER INDEX** コマンド
- SYSTS\_CLEAR\_EVENTS プロシージャまたは **db2ts CLEAR EVENTS FOR INDEX** コマンド
- SYSTS\_DISABLE プロシージャまたは **db2ts DISABLE DATABASE FOR TEXT** コマンド
- SYSTS\_UPDATE プロシージャまたは **db2ts UPDATE INDEX** コマンド

競合がある場合、プロシージャは SQLCODE -20426 および SQLSTATE 38H13 を戻します。

- このプロシージャは、すべての索引更新処理が完了するまで戻されません。この期間は、これから索引付けされる文書の数および既に索引付けされている文書

の数に応じて異なります。索引のコレクション名は、SYSIBMTS.TSCOLLECTIONNAMES ビューの COLLECTIONNAME 列から取られます。

- 個々の文書にエラーがある場合、文書を訂正する必要があります。エラーがある文書の主キーは、索引のイベント表で参照できます。ユーザー表内の対応する行を変更することで、SYSTS\_UPDATE への次の呼び出しはこれらの文書を再処理します。
- SYSTS\_UPDATE プロシージャが実行すると、次のイベントが発生します。
  - 行がイベント表に挿入されます (パーサー・エラー情報を含む)。インクリメンタル更新の場合、情報は索引ステー징表から削除されます。最初の更新の前に、SYSTS\_UPDATE プロシージャはユーザー表にトリガーを作成します。
  - コレクションが更新されます。
    - 新規または変更された文書が構文解析および索引付けされます。
    - 削除された文書が索引から破棄されます。
- シノニム・ディクショナリーがテキスト索引に関連付けられている場合、ALLROWS オプションまたは FOR DATA REDISTRIBUTION オプションを指定して更新を実行すると、テキスト索引のコレクションがドロップおよび再作成されて、関連が除去されます。シノニム・ディクショナリーは、新しいテキスト索引コレクションに関連付ける必要があります。

---

## SYSTS\_UPGRADE\_CATALOG プロシージャ - テキスト検索カタログのアップグレード

このプロシージャは、管理表および管理ビューを含む DB2 Text Search カタログを、製品の最新バージョンにアップグレードします。

このプロシージャは、製品の最新バージョンで使用される新しいカタログ表およびビューを作成するとともに、既存のカタログ表およびビューを更新します。廃止されたカタログ表およびビューは削除されます。

### 許可

このプロシージャの許可 ID が持つ特権に、SYSTS\_ADM ロールおよび DBADM 権限が含まれている必要があります。

### デフォルトの PUBLIC 特権

なし

### 構文

▶—SYSTS\_UPGRADE\_CATALOG—(—message\_locale—,—message—)————▶

スキーマは SYSPROC です。

## プロシージャ・パラメーター

### *message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 次の例では、以前のリリースのテキスト検索がデータベースで使用可能になっています。 **SYSTS\_UPGRADE\_CATALOG** プロシージャを呼び出すと、テキスト検索カタログ表がアップグレードされます。プロシージャが成功すると、操作の成功を示す出力パラメーター・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_UPGRADE_CATALOG('en_US',?)
```

Value of output parameters

```
-----  
Parameter Name : MESSAGE  
Parameter Value : CIE0213W The DB2 Text Search Catalog has been upgraded to  
the current version. You will now need to update the text search index using  
the SYSPROC.SYSTS_UPGRADE_INDEX stored procedure.  
index also needs to be updated by calling "SYSPROC.SYSTS_UPGRADE_INDEX".
```

Return Status = 0

例 2: 次の例では、プロシージャを呼び出す前は、旧リリースの DB2 Text Search がデータベースで使用可能になっていません。 *message\_locale* の位置にある NULL 値は、デフォルト・ロケール 'en\_US' が使用されることを意味します。エラー・メッセージはすべて英語で戻されます。

```
CALL SYSPROC.SYSTS_UPGRADE_CATALOG('',?)
```

```
SQL20427N An error occurred during a text search administration procedure  
or command. The error message is "CIE0323E Specified or default database  
not enabled for text. ".  
SQLSTATE=38H14
```

例 3: 次の例では、テキスト検索カタログとテキスト検索索引が現行バージョンに既にアップグレードされています。

```
CALL SYSPROC.SYSTS_UPGRADE_CATALOG('en_US',?)
```

Value of output parameters

```
-----  
Parameter Name : MESSAGE  
Parameter Value : CIE0002I The DB2 Text Search release level is current  
for the database. The system has not been upgraded.
```

Return Status = 0

## 使用上の注意

テキスト検索カタログ固有のアップグレードを実行するため、**SYSTS\_UPGRADE\_CATALOG** プロシージャが **DB2 UPGRADE DATABASE** コマンドに組み込まれています。データベース・アップグレードがテキスト検索カタログのアップ



グレードに失敗した場合、ユーザーが **SYSTS\_UPGRADE\_CATALOG** プロシージャを別個に実行し、データベース・アップグレードを完了させる必要があります。

テキスト検索索引カタログをアップグレードするには、以下の手順を実行します。

1. DB2 Text Search インスタンス・サービスが停止していることを確認します。
2. **SYSTS\_UPGRADE\_CATALOG** プロシージャを実行します。

**注:** DB2 Text Search 管理プロシージャは、データベースへの既存の接続を使用します。現行トランザクションは、プロシージャの完了結果に応じてコミットされるかまたはロールバックされる場合があります。そのため、そのようなコミットまたはロールバックからの予期しない影響を避けるために、すべてのトランザクション変更をコミットすることができます。これを行う 1 つの方法は、**AUTOCOMMIT** をオンにすることです。

3. **SYSTS\_UPGRADE\_CATALOG** プロシージャが、カタログに Text Search サーバー情報の設定を試行します。SYSIBMTS.TSSERVER の内容を検討して、必要に応じて、Text Search サーバー情報を更新してください。
4. **SYSTS\_UPGRADE\_INDEX** プロシージャについて示されている手順に従って、テキスト検索索引をアップグレードします。**SYSTS\_UPGRADE\_INDEX** プロシージャの実行が正常に完了するまでは、SYSIBMTS.TSDEFAULTS 管理ビュー内のバージョン値が更新されません。

---

## SYSTS\_UPGRADE\_INDEX - テキスト検索索引のアップグレード

このプロシージャは、DB2 カタログ表およびテキスト検索カタログ表での DB2 Text Search 索引情報を更新します。

テキスト検索索引コレクションは、Text Search サーバーによって管理されます。詳しくは、DB2 Text Search のアップグレードを参照してください。

### 許可

このプロシージャの許可 ID が持つ特権に、SYSTS\_ADM ロールおよび DBADM 権限が含まれている必要があります。

### デフォルトの PUBLIC 特権

なし

### 構文

▶▶—SYSTS\_UPGRADE\_INDEX—(—message\_locale—, —message—)▶▶

スキーマは SYSPROC です。

### プロシージャ・パラメーター

*message\_locale*

戻されるすべてのエラー・メッセージに使用されるロケールを指定する、タイプ

VARCHAR(33) の入力引数。引数が NULL または空ストリングであるか、指定したロケールのメッセージ・ファイルがサーバーで使用できない場合には、'en\_US' が使用されます。

#### *message*

正常に完了した操作の警告または通知メッセージを指定する、タイプ VARCHAR(32K) の出力引数。

## 例

例 1: 次の例では、旧リリースのテキスト検索がデータベースで使用可能になりましたが、プロシージャ **SYSTS\_UPGRADE\_CATALOG** が既に正常に完了しています。プロシージャ **SYSTS\_UPGRADE\_INDEX** は、データベース・カタログでのテキスト索引メタデータのアップグレードを完了します。このプロシージャが成功すると、操作の成功を示す出力パラメーター・メッセージが呼び出し元に戻されます。

```
CALL SYSPROC.SYSTS_UPGRADE_INDEX('en_US', ?)
```

```
Parameter Name : MESSAGE
```

```
Parameter Value : CIE00001 Operation completed successfully.
```

```
Return Status = 0
```

例 2: 次の例では、旧リリースの製品のテキスト検索がデータベースで使用可能になっていません。 *message\_locale* に NULL 値を設定した場合、システム・ロケールが使用されることとなります。システム・ロケールが使用不可の場合は、デフォルト・ロケール 'en\_US' が使用されます。

```
CALL SYSPROC.SYSTS_UPGRADE_INDEX('', ?)
```

```
SQL20427N An error occurred during a text search administration procedure  
or command. The error message is "CIE0323E Specified or default database  
not enabled for text."  
SQLSTATE=38H14
```

例 3: 次の例では、DB2 Text Search カタログとテキスト検索の索引が現行バージョンに既にアップグレードされています。

```
CALL SYSPROC.SYSTS_UPGRADE_INDEX('en_US', ?)
```

```
Value of output parameters
```

```
-----
```

```
Parameter Name : MESSAGE
```

```
Parameter Value : CIE0002I The DB2 Text Search release level is current for  
the database. The system has not been upgraded.
```

```
Return Status = 0
```

例 4: 次の例では、DB2 Text Search カタログがアップグレードされていません。

```
CALL SYSPROC.SYSTS_UPGRADE_INDEX('en_US', ?)
```

```
CIE0409E The DB2 Text Search catalog has not been upgraded to the current version.
```

## 使用上の注意

- テキスト検索索引のアップグレードを実行するため、**SYSTS\_UPGRADE\_INDEX** プロシージャが **DB2 UPGRADE DATABASE** コマンドに組み込まれています。 **DB2 UPGRADE DATABASE** コマンドを発行すると、このプロシージャもまた実行されます。テキスト検索索引のアップグレードに失敗した場合、手順を手動で実行する必要があります。

- **SYSTS\_UPGRADE\_INDEX** プロシージャの前に **SYSTS\_UPGRADE\_CATALOG** プロシージャを実行する必要があります。 **SYSTS\_UPGRADE\_INDEX** プロシージャの実行が正常に完了するまでは、**SYSIBMTS.TSDEFAULTS** 管理ビュー内のバージョン値が更新されません。
- **DB2 Text Search** 管理プロシージャは、データベースへの既存の接続を使用します。現行トランザクションは、プロシージャの完了結果に応じてコミットされるかまたはロールバックされる場合があります。そのため、そのようなコミットまたはロールバックからの予期しない影響を避けるために、すべてのトランザクション変更をコミットすることができます。これを行う 1 つの方法は、**AUTOCOMMIT** をオンにすることです。



## 第 12 章 テキスト検索の管理ビュー

DB2 Text Search は、データベース内のテキスト検索索引とそれらのプロパティーについて説明するいくつかの管理ビューを作成および保守します。

特に指示されない限り、これらのビューのいずれも更新しないでください。

以下のビューは、システムの現在の構成を反映します。

- データベース・レベルのビュー:
  - SYSIBMTS.TSDEFAULTS
  - SYSIBMTS.TSLOCKS
  - SYSIBMTS.TSSERVERS
- 索引レベルのビュー:
  - SYSIBMTS.TSINDEXES
  - SYSIBMTS.TSCONFIGURATION
  - SYSIBMTS.TSCOLLECTIONNAMES
  - SYSIBMTS.TSEVENT\_#####
  - SYSIBMTS.TSSTAGING\_#####

### SYSIBMTS.TSDEFAULTS ビュー

SYSIBMTS.TSDEFAULTS は、データベース内のすべてのテキスト検索索引のデフォルト値を表示します。

このビューでは、デフォルト値は属性と値の対として表示されます。

表 15. SYSIBMTS.TSDEFAULTS ビュー

列名	データ・タイプ	NULL 可能	説明
DEFAULTNAME	VARCHAR (30)	いいえ	テキスト検索のデータベース・デフォルト・パラメーター
DEFAULTVALUE	VARCHAR (512)	いいえ	テキスト検索のデータベース・デフォルト・パラメーターの値

以下の値は、db2ts **CREATE INDEX**、**ALTER INDEX**、**UPDATE INDEX**、および **CLEAR EVENTS FOR INDEX** コマンドのデフォルトとして使用されます。

- **AUXLOGNORM**: 明示的な索引構成 **AUXLOG ON** でテキスト検索索引に対してステージング・インフラストラクチャーを有効にすることができます。非パーティション表に対しては、拡張テキスト保持ステージング・インフラストラクチャーをデフォルトで有効にしません。
- **AUXLOGPART**: 明示的な索引構成 **AUXLOG OFF** でテキスト索引に対してステージング・インフラストラクチャーを無効にすることができます。デフォルトでは、範囲パーティション表に対して拡張テキスト保持ステージング・インフラストラクチャーを有効にします。

- **CJKSEGMENTATION**: 中国語、日本語、および韓国語の文書を索引付けする際に使用するセグメンテーションの方法を指定します。サポートされる値には、**MORPHOLOGICAL** および **NGRAM** があります。デフォルト値は **NGRAM** です。
- **CODEPAGE**: 新規索引の初期デフォルトのコード・ページは、データベース・コード・ページです。
- **DOCUMENTRESULTQUEUE SIZE**: この値は、1つのコレクションの更新操作ごとに予約されるデータベース・メモリー量を制限するために使用されます。デフォルト値は 30,000 で、その範囲は 100 から 100,000 です。複数パーティション・セットアップの場合、並列実行用に構成された単一のテキスト索引更新では、更新が必要なコレクションごとにメモリー・スペースが予約されます。
- **FORMAT**: 文書フォーマットの初期デフォルトはプレーン・テキストです。
- **LANGUAGE**: 文書索引付けの初期デフォルトは、**en\_US** です。
- **MAXCONCURRENTUPDATES**: 特定の時点で並行して実行できるコレクション更新の数を制御します。複数パーティション・セットアップの場合、各テキスト索引に対するコレクションの数は、表の配分にしがたって決定されます。ただし、カウントされるのはアクティブなパーティションの更新のみです。デフォルトは 8 です。
- **MAXCONCURRENTCOLLECTIONS**: 作成できるコレクションの数を制御します。単一ノード・データベースの場合、コレクションの数はテキスト索引の数と同じです。複数パーティション・セットアップの場合、テキスト索引ごとのコレクション数は表の配分と一致します。デフォルトは 160 です。
- **MAXDOCUMENTSIZEINMB**: 処理対象として受け入れられる文書のサイズを制御します。制限を超えるテキストがあると、イベント表に警告メッセージが書き込まれます。値は 100 です。
- **UPDATEFREQUENCY**: 新規索引の更新スケジュールの初期デフォルトは、**NONE** です。
- **UPDATEMINIMUM**: 新規索引更新の初期デフォルトは 1 で、インクリメンタル更新が変更ごとに行われます。
- **UPDATEAUTOCOMMIT**: 新規索引更新の初期デフォルトは 0 で、DB2 テキスト列から文書が読み取られるときに、中間コミットは行われません。この値は予約済みで、変更できません。

デフォルト値は、データベース・レベルで **db2ts** コマンドを使用して変更することはできません。

## SYSIBMTS.TSLOCKS ビュー

SYSIBMTS.TSLOCKS を使用して、データベースおよび索引レベルのコマンド・ロック情報を表示できます。

表 16. SYSIBMTS.TSLOCKS ビュー

列名	データ・タイプ	NULL 可能	説明
COMMAND	VARCHAR(30)	いいえ	ロックを作成したコマンドの名前。可能な値は、CREATE INDEX、ALTER INDEX、DROP INDEX、UPDATE INDEX、CLEAR EVENTS、DISABLE DATABASE、CONFIGURE、CLEANUP です。

表 16. SYSIBMTS.TSLOCKS ビュー (続き)

列名	データ・タイプ	NULL 可能	説明
LOCKSCOPE	VARCHAR(30)	いいえ	ロックの有効範囲。可能な値は、DATABASE または INDEX です。
INDSCHEMA	VARCHAR(128)	いいえ	テキスト検索索引のスキーマ名 (LOCKSCOPE = INDEX の場合のみ)
INDNAME	VARCHAR(128)	いいえ	テキスト検索索引の非修飾名 (LOCKSCOPE = INDEX の場合のみ)
PARTITION	INTEGER	いいえ	テキスト検索のロックが作成されているパーティション番号
LOCKCREATETIME	TIMESTAMP	いいえ	ロックが付与されたときのタイム・スタンプ

ロッキング戦略について理解するための 3 つの異なるシナリオがあります。

- 操作が開始され、適用可能なロックが発生しない: プロシージャによりロックが設定され、実行が続行されます。実行が成功しても失敗しても、ロックは解除されます。
- 操作が開始され、適用可能なロックが発生する: 競合するコマンドのメッセージで要求が返されます。
- 操作が開始され、関連付けられている操作が現在実行中でない場合でも適用可能なロックが発生する: 以前の操作でロックを適切に解除できないようにする障害が発生しています。これは、ディスク障害、異常終了の異常な状況で発生する可能性があります。このような場合は、障害の原因の対処を行い、システムの整合性を確認した後に、適宜索引レベルまたはデータベース・レベルで **CLEAR COMMAND LOCKS** 操作を発行してロックを解除する必要があります。

## SYSIBMTS.TSSERVERS ビュー

SYSIBMTS.TSSERVERS ビューの各行の表示は、データベース用に構成された DB2 Text Search サーバーに関する情報を示します。

ビューを照会して、使用するサーバーとしてマーク付けされているテキスト検索サーバーに関する情報を取得できます。

```
db2 "SELECT SERVERID, HOST from SYSIBMTS.TSSERVERS where SERVERSTATUS = 0"
```

表 17. SYSIBMTS.TSSERVERS ビュー

列名	データ・タイプ	NULL 可能	説明
SERVERID	INTEGER	いいえ	テキスト検索サーバー用に生成される固有の ID。
HOST	VARCHAR(256)	いいえ	テキスト検索サーバーのホスト名または IP アドレス。パーティション・データベースでは、スタンドアロン・テキスト検索サーバーをデプロイメントする場合、またはリモート・クライアントから管理操作を実行する場合に、「localhost」ではなく、実際のホスト名または IP アドレスを使用していることを確認します。
PORT	INTEGER	いいえ	テキスト検索サーバーのポート番号。(ADMIN/SEARCH)
TOKEN	VARCHAR(256)	いいえ	テキスト検索サーバーの認証トークン。
KEY	VARCHAR(128)	いいえ	テキスト検索サーバーのサーバー・キー。

表 17. SYSIBMTS.TSSERVERS ビュー (続き)

列名	データ・タイプ	NULL 可能	説明
DEFAULTLOCALE	VARCHAR(33)	いいえ	テキスト検索サーバーからのメッセージに対して想定されるデフォルトのクライアント・ロケール
SERVERTYPE	INTEGER	いいえ	この値は、各テキスト検索サーバーのタイプを示します。 <ul style="list-style-type: none"> <li>• 0 = デフォルト (統合) テキスト検索サーバー</li> <li>• 0 以外の値 = スタンドアロン・テキスト検索サーバー <ul style="list-style-type: none"> <li>- 1 = ローカルのスタンドアロン・テキスト検索サーバー</li> <li>- 2 = リモートのスタンドアロン・テキスト検索サーバー</li> </ul> </li> </ul>
SERVERSTATUS	INTEGER	いいえ	テキスト検索サーバーを使用して、新しいテキスト検索索引を作成するかどうかを示します。デフォルト値は 0 で、サーバーがアクティブで使用可能であることを示します。

## SYSIBMTS.TSINDEXES ビュー

SYSIBMTS.TSINDEXES ビューには、現在のテキスト検索索引プロパティが表示されます。

以下の例では、索引スキーマと索引名を使用します。

```
db2 "SELECT COLNAME from SYSIBMTS.TSINDEXES where INDSHEMA=schema-name
and INDNAME=index-name"
```

以下の表で、SYSIBMTS.TSINDEXES ビューについて説明します。

表 18. SYSIBMTS.TSINDEXES ビュー

列名	データ・タイプ	NULL 可能	説明
INDSCHEMA	VARCHAR(128)	いいえ	テキスト検索索引のスキーマ名。
INDNAME	VARCHAR(128)	いいえ	テキスト検索索引の非修飾名。
TABSCHEMA	VARCHAR(128)	いいえ	基本表のスキーマ名。
TABNAME	VARCHAR(128)	いいえ	基本表の非修飾名。
COLNAME	VARCHAR(128)	いいえ	テキスト検索索引が作成された列。
CODEPAGE	INTEGER	いいえ	テキスト検索索引の文書コード・ページ。
LANGUAGE	VARCHAR(5)	いいえ	テキスト検索索引の文書言語。
FORMAT	VARCHAR(30)	はい	文書フォーマット。
FUNCTIONSCHEMA	VARCHAR(128)	はい	列タイプのスキーマ。
FUNCTIONNAME	VARCHAR(18)	はい	列タイプ変換関数の名前。
COLLECTIONDIRECTORY	VARCHAR(512)	はい	テキスト検索索引ファイルのディレクトリー。
UPDATEFREQUENCY	VARCHAR(300)	いいえ	索引へ更新を適用するトリガーの基準。



表 18. SYSIBMTS.TSINDEXES ビュー (続き)

列名	データ・タイプ	NULL 可能	説明
UPDATEMINIMUM	INTEGER	はい	インクリメンタル更新が実行される前にログ表にある項目の最小数。値が低いほど、表列とテキスト検索索引との間の整合性が高まります。ただし、値が低いと、テキスト検索索引付けに必要なリソースも増加します。
EVENTVIEWSHEMA	VARCHAR(128)	いいえ	テキスト検索索引用に作成されたイベント・ビューのスキーマ (常に SYSIBMTS)。
EVENTVIEWNAME	VARCHAR(128)	いいえ	テキスト検索索引用に作成されたイベント・ビューの名前。
STAGINGVIEWSHEMA	VARCHAR(128)	はい	テキスト検索索引用に作成されたログ・ビューのスキーマ (常に SYSIBMTS)。
STAGINGVIEWNAME	VARCHAR(128)	はい	テキスト検索索引用に作成されたログ・ビューの名前。
REORGAUTOMATIC	INTEGER	はい	予約済み (このリリースではサポートされません)。値は常に 1 です。
RECREATEONUPDATE	INTEGER	いいえ	予約済み (このリリースではサポートされません)。値は常に 0 です。
ATTRIBUTES	VARCHAR(32000)	はい	予約済み (このリリースではサポートされません)。
COLLECTIONNAMEPREFIX	VARCHAR(128)	いいえ	テキスト検索サーバーのコレクション名の接頭部。
COMMENT	VARCHAR(512)	はい	<b>CREATE INDEX</b> コマンドの索引プロパティに関連したパラメーターに指定されたコメント。
AUXSTAGINGSHEMA	VARCHAR(48)	はい	テキスト保持ステージング表のスキーマ。
AUXSTAGINGNAME	VARCHAR(48)	はい	テキスト保持ステージング表の名前。
INDSTATUS	VARCHAR(10)	いいえ	索引状況: <ul style="list-style-type: none"> <li>• <b>ACTIVE</b> はアクティブな索引を示します。</li> <li>• <b>INACTIVE</b> は非アクティブな索引を示します。(この値は、DB2 Text Search では使用しません。)</li> <li>• <b>INVALID</b> は、無効にされた索引を示します。通常は DB2 操作の副次作用です。</li> </ul>

---

## SYSIBMTS.TSCONFIGURATION ビュー

索引構成パラメーターに関する情報は、SYSIBMTS.TSCONFIGURATION ビューにあります。

各行は、テキスト検索索引の構成パラメーターを表します。

以下に、索引名を使用したビューに対する照会の例を示します。

```
db2 "SELECT VALUE from SYSIBMTS.TSCONFIGURATION where INDSHEMA=schema-name
and INDNAME=ind-name and PARAMETER ='parameter'"
```

表 19. SYSIBMTS.TSCONFIGURATION ビュー

列名	データ・タイプ	NULL 可能	説明
INDSCHEMA	VARCHAR(128)	いいえ	テキスト検索索引のスキーマ名
INDNAME	VARCHAR(128)	いいえ	テキスト検索索引の非修飾名
PARAMETER	VARCHAR(30)	いいえ	構成パラメーターの名前
VALUE	VARCHAR(512)	いいえ	パラメーターの値

PARAMETER 列には、テキスト検索索引構成パラメーターの名前 (CREATE INDEX ステートメントで指定されている) および SYSIBMTS.TSDEFAULTS ビューからのいくつかのパラメーターの名前が含まれます。

---

## SYSIBMTS.TSCOLLECTIONNAMES ビュー

SYSIBMTS.TSCOLLECTIONNAMES ビューは、コレクションの名前を表示します。

各行は、テキスト検索索引のコレクションを表します。

表 20. SYSIBMTS.TSCOLLECTIONNAMES ビュー

列名	データ・タイプ	NULL 可能	説明
INDSCHEMA	VARCHAR(128)	いいえ	テキスト検索索引のスキーマ名
INDNAME	VARCHAR(128)	いいえ	テキスト検索索引の非修飾名
COLLECTIONNAME	VARCHAR(132)	いいえ	テキスト検索サーバー上の関連コレクションの名前。パーティション・データベース・システムでは、各テキスト索引パーティションがコレクションとして表されます。コレクション名にはパーティション番号が接尾部として含まれます。

---

## SYSIBMTS.TSEVENT ビュー

このイベント・ビューは、索引付けの状況およびエラー・イベントに関する情報を示します。

データベースには、接頭部 SYSIBMTS.TSEVENT のビューが複数ある場合があります。各ビューは、ビューが関連付けられている対応するテキスト索引を指す内部 ID、nnnnnn 値によって区別されます。特定のビューに関連付けられているテキスト検索索引を判別するには、ビュー SYSIBMTS.TSINDEXES を照会して、列

EVENTVIEWSHEMA および EVENTVIEWNAME 内でスキーマ名およびビュー名を検索します。照会は、問題のテキスト検索索引およびユーザー表を記述する単一の行を戻します。

このビューの列の数は、ユーザー表内の主キー列の数によって異なります。列 PK1..PKnn は、ユーザー表の主キー列と一致し、対応するデータ・タイプと長さの定義が含まれます。ビュー内の各列のデータ・タイプは、対応する主キー列のデータ・タイプに厳密に対応します。

このビューの各行は、テキスト検索索引に発行された **UPDATE INDEX** コマンドからのメッセージを表します。例えば、ある行は **UPDATE INDEX** コマンドが開始済みである、または完了したことを示す場合があります。また、ある行はテキスト文書に索引が付けられたときに生じた問題を説明する場合があります。そのテキスト文書は、このビューの行から主キー列値を取得して、それらをユーザー表で調べることにより、見つけることができます。

イベントは、db2ts **CLEAR EVENTS FOR INDEX** コマンドを使用して消去できます。

表 21. イベント・ビュー

列名	データ・タイプ	NULL 可能	説明
OPERATION	INTEGER	はい	テキスト検索索引に反映される基本表に対する操作 (挿入、更新、または削除)
TIME	TIMESTAMP	はい	イベント項目作成のタイム・スタンプ
SEVERITY	INTEGER	はい	メッセージが単一の文書に対応する場合、以下の値のうちいずれか。 <ul style="list-style-type: none"> <li>• 1 = 情報</li> <li>• 4 = 文書の一部は索引を付けられたが、メッセージで示されたように警告がある</li> <li>• 8 = メッセージで示されたように、文書には索引が付けられなかった</li> <li>• 0= その他</li> </ul>
SQLCODE	INTEGER	はい	関連したエラーがあった場合、その SQLCODE
MESSAGE	VARCHAR(1024)	はい	特定のエラーに関するテキスト情報
PARTITION	INTEGER	はい	IBM が内部で使用するために予約済みです。
PK01	基本表の最初の主キー列のデータ・タイプ	はい	イベントが生じたときに処理されていた行の、テキスト検索索引の基本表の最初の主キー列の値
...	...	...	...
PKnn	基本表の最後の主キー列のデータ・タイプ	はい	イベントが生じたときに処理されていた行の、テキスト検索索引の基本表の最後の主キー列の値

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も、このビューで使用可能です。この場合、PK01、PKnn、および OPERATION には、すべて NULL 値が指定されています。MESSAGE のコード・ページおよびロケールは、データベースの設定に対応します。

## SYSIBMTS.TSSTAGING ビュー

このステージング表は、テキスト検索索引との同期を必要とするユーザー表に対する変更操作を保管します。

変更情報をステージング表に送るためにトリガーがユーザー表上に作成されます。さらに、補助ログ・オプションを使用して、整合性処理でユーザー表に対する変更を検出します。 **UPDATE INDEX** コマンドはそれらの項目を読み取り、正常な同期の後、削除します。

データベースには、接頭部 **SYSIBMTS.TSSTAGING\_** のビューが複数ある場合があります。各ビューは、ビューが関連付けられている対応するテキスト索引を指す内部 ID、*nnnnnn* 値によって区別されます。特定のビューと関連付けられているテキスト検索索引を判別するには、ビュー **SYSIBMTS.TSINDEXES** を照会して、列 **STAGINGVIEWSHEMA** および **STAGINGVIEWNAME** 内でスキーマ名およびビュー名を検索します。照会は、問題のテキスト検索索引およびユーザー表を記述する単一の行を戻します。

このビューの列の数は、ユーザー表内の主キー列の数によって異なります。列 **PK1..PK*n*** は、ユーザー表の主キー列と一致し、対応するデータ・タイプと長さの定義が含まれます。ビュー内の各列のデータ・タイプは、対応する主キー列のデータ・タイプに厳密に対応します。

このビューの各行は、ユーザー表行またはテキスト文書に対する 1 回の挿入、削除、または更新操作を表します。そのテキスト文書は、このビューの行から主キー列値を取得して、それらをユーザー表で調べることにより、見つけることができます。

ビューに関する情報は、以下の照会を使用して取得できます。

```
db2 "SELECT STAGINGVIEWSHEMA, STAGINGVIEWNAME from SYSIBMTS.TSINDEXES
where INDSHEMA=schema-name and INDNAME=index-name"
```

表 22. **SYSIBMTS.TSSTAGING** ビュー

列名	データ・タイプ	NULL 可能	説明
OPERATION	INTEGER	いいえ	テキスト検索索引に反映される基本表に対する操作 (挿入、更新、または削除)
TIME	TIMESTAMP	いいえ	行のシーケンス ID (挿入、更新、または削除トリガーの起動時)。これはタイム・スタンプですが、操作の時刻を厳密に表すものではありません。
STATUS	INTEGER	いいえ	行の処理状況: -1 未処理
PK01	基本表の最初の主キー列のデータ・タイプ	はい	基本表の最初の主キー列。
...	...	...	...
PK <i>n</i>	基本表の最後の主キー列のデータ・タイプ	はい	基本表の最後の主キー列。

## 付録 A. DB2 Text Search および Net Search Extender の比較

Net Search Extender (NSE) から DB2 Text Search へのマイグレーションを行う前に、フルテキスト検索照会について、両方の解決策は似ているものの、構文、意味、および結果セットは異なるということを認識しておく必要があります。

表 23 および 230 ページの表 24 は、NSE から DB2 Text Search への移植が可能かどうかを判断する際に役立ちます。

DB2 Text Search は、NSE がサポートされているすべてのオペレーティング・システム (Linux on System z<sup>®</sup> (64 ビット) オペレーティング・システムを除く) でサポートされています。以下の表は、NSE および DB2 Text Search で使用可能なインストール機能のリストを示しています。

表 23. NSE および DB2 Text Search で使用可能なインストール機能

機能	NSE	DB2 Text Search	コメント、および追加情報へのリンク
テキスト・エンジンのローカル・インストール	はい	はい	
テキスト・エンジンのリモート・インストール	いいえ	はい	『DB2 Text Search サーバーのデプロイメント・シナリオ』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0058598.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0058598.html</a> )
データベース・パーティション化	はい	はい	『パーティション・データベース環境の DB2 Text Search』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0058524.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0058524.html</a> )
パーティション化されていない基本表への索引付け	はい	はい	テキスト検索索引の作成、更新、およびプロパティの変更 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c_textindexcreation.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c_textindexcreation.html</a> )

表 23. NSE および DB2 Text Search で使用可能なインストール機能 (続き)

機能	NSE	DB2 Text Search	コメント、および追加情報へのリンク
パーティション化された (範囲パーティション化された) 基本表への索引付け	はい	はい	テキスト検索索引インクリメンタル更新のための拡張テキスト保守ステージング・インフラストラクチャー ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0057426.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0057426.html</a> )
ニックネームへの索引付け (レプリケーションによる)	非推奨	いいえ	バージョン 9.7 で非推奨になりました。
ビューへの索引付け	はい	いいえ	

DB2 Text Search には、NSE の機能と類似する機能があります。以下の表に、NSE および DB2 Text Search で使用可能な機能を示します。

表 24. NSE および DB2 Text Search で使用可能な機能

機能項目	NSE	DB2 Text Search	コメント、および追加情報へのリンク
更新時の再作成	はい	はい	
カスタム変換関数	はい	はい	
キャッシング	いいえ	いいえ	
複数索引	はい	いいえ	
索引の事前ソート	いいえ	いいえ	
シノニム・ディクショナリー	はい	はい	『DB2 Text Search のシノニム・ディクショナリー』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0052652.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0052652.html</a> )
シソーラス (連想、階層、ユーザー定義)	いいえ	いいえ	
テキスト、HTML、XML	はい	はい	『DB2 Text Search でサポートされる文書フォーマット』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r0053096.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r0053096.html</a> )

表 24. NSE および DB2 Text Search で使用可能な機能 (続き)

機能項目	NSE	DB2 Text Search	コメント、および追加情報へのリンク
INSO	はい	はい	DB2 Text Search では、DB2 Accessories Suite パッケージを使用する INSO がサポートされます。詳しくは、『リッチ・テキスト文書のサポート』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0054766.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0054766.html</a> ) を参照してください。
GPP	はい	いいえ	DB2 Text Search で関数を作成することで、GPP をサポートできます。
文書モデル	はい	いいえ	
言語学的処理	はい	はい <sup>+</sup>	NSE 言語学的処理は、簡単なステミング (英語のみ) に制限されます。  DB2 Text Search では、形態素セグメンテーションと、中国語、日本語、および韓国語のための N-gram セグメンテーションの両方のサポートを含む、20 カ国語の言語学的処理がサポートされます。詳しくは、『DB2 Text Search の言語学的処理』を参照してください。
CONTAINS 関数	はい	はい	『CONTAINS 関数』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r_contains.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r_contains.html</a> )
SCORE 関数	はい	はい	DB2 Text Search では、異なる結果を戻す可能性のある異なるアルゴリズムを使用します。詳しくは、『SCORE 関数』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r_score.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r_score.html</a> ) を参照してください。
一致する数	いいえ	いいえ	
強調表示	いいえ	いいえ	

表 24. NSE および DB2 Text Search で使用可能な機能 (続き)

機能項目	NSE	DB2 Text Search	コメント、および追加情報へのリンク
ストップワード処理	はい	はい	『DB2 Text Search 構文のストップワード・ツール』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r0058492.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r0058492.html</a> )
結果限度	はい	はい	CONTAINS 関数と SCORE 関数には、戻される結果の最大数を示す <b>RESULTLIMIT</b> パラメーターがあります。
文字の正規化	はい	はい	
エスケープ文字	はい	はい	DB2 Text Search ではカスタマイズを使用できません。
Boolean 検索	はい	はい	『テキスト検索索引数の構文』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r0052651.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r0052651.html</a> )
ワイルドカード文字	はい	はい	『テキスト検索索引数の構文』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r0052651.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/r0052651.html</a> )
語幹検索	はい	はい	語幹検索は DB2 Text Search のデフォルトです。
厳密検索	はい	はい	DB2 Text Search では大/小文字が区別されません。詳しくは、『厳密検索』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/t_searchingwiththetextindex.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/t_searchingwiththetextindex.html</a> ) を参照してください。
ファジー検索	はい	はい	『ファジー検索』 ( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0058557.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0058557.html</a> )
近接検索	はい	はい	『近接検索』 ( <a href="http://www.ibm.com/support.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0058673.html">http://www.ibm.com/support.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0058673.html</a> )



表 24. NSE および DB2 Text Search で使用可能な機能 (続き)

機能項目	NSE	DB2 Text Search	コメント、および追加情報へのリンク
範囲検索	はい	はい (XML の場合)	DB2 Text Search は、範囲検索を行う場合は XML の XPath 式に依存します。Net Search Extender は、文書モデルを介した範囲検索をサポートしています。
フリー・テキスト検索	はい	いいえ	
フィールド化された検索	はい	はい (XML の場合)	DB2 Text Search サポートでは XML の XPath 式が使用されます。NSE サポートでは文書モデルが使用されます。詳しくは、『DB2 Text Search の XML 検索構成』( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0052709.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0052709.html</a> ) および 『DB2 Text Search を使用した XML 文書の検索』( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0052708.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/c0052708.html</a> ) を参照してください。
属性検索	はい	いいえ	
重み/ランキング調整	はい	はい	DB2 Text Search および NSE にはさまざまなアルゴリズムがあります。詳しくは、『SCORE を使用したテキスト検索索引の検索』( <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/t_searchingandreturningscore.html">http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ts.doc/doc/t_searchingandreturningscore.html</a> ) を参照してください。



## 付録 B. DB2 Text Search でサポートされるロケール

以下の表には、DB2 Text Search による文書処理でサポートされるロケールがリストされています。

表 25. サポートされるロケール

ロケール・コード	言語	地域
ar_AA	アラビア語	アラブ諸国または諸地域
cs_CZ	チェコ語	チェコ共和国
da_DK	デンマーク語	デンマーク
de_CH	ドイツ語	スイス
de_DE	ドイツ語	ドイツ
el_GR	ギリシャ語	ギリシャ
en_AU	英語	オーストラリア
en_GB	英語	英国
en_US	英語	米国
es_ES	スペイン語	スペイン
fi_FI	フィンランド語	フィンランド
fr_CA	フランス語	カナダ
fr_FR	フランス語	フランス
it_IT	イタリア語	イタリア
ja_JP	日本語	日本
ko_KR	韓国語	韓国
nb_NO	ノルウェー語ブークモール	ノルウェー
nl_NL	オランダ語	オランダ
nn_NO	ノルウェー語ニーノシュク	ノルウェー
pl_PL	ポーランド語	ポーランド
pt_BR	ポルトガル語	ブラジル
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
zh_CN	中国語	中国
zh_TW	中国語	台湾



## 付録 C. DB2 コマンド

### db2iupgrade - インスタンスのアップグレード・

インスタンスを前のリリースの DB2 コピーから、現行リリースの DB2 コピーにアップグレードします。 **db2iupgrade** コマンドの実行元の DB2 コピーは、アップグレードする DB2 コピーからのインスタンス・サポートをアップグレードしていません。

Linux および UNIX オペレーティング・システムでは、このコマンドは *DB2DIR/instance* ディレクトリにあります。 *DB2DIR* は、DB2 データベース・システムの新規リリースがインストールされている場所です。このコマンドは、非 root インストールのインスタンス・アップグレードはサポートしません。

Windows オペレーティング・システムでは、このコマンドは *DB2PATH\bin* ディレクトリにあります。 *DB2PATH* は、DB2 コピーがインストールされている場所です。インスタンス・プロファイルを現在の場所から別の場所に移動するには、 */p* オプションを使用し、インスタンス・プロファイル・パスを指定します。こうしない場合、アップグレード後にインスタンス・プロファイルが元の場所に残ります。

#### 許可

Linux および UNIX オペレーティング・システム上の root ユーザー権限、または Windows オペレーティング・システム上のローカル管理者

#### コマンド構文

Linux および UNIX オペレーティング・システムの場合

```
db2iupgrade [-d] [-k]
             [-j "TEXT_SEARCH" [,servicename] [,portnumber]]
             [-a AuthType] [-u FencedID] InstName
```

Windows オペレーティング・システムの場合

```
db2iupgrade InstName /u:username,password
             [/p:instance-profile-path] [/q] [/a:authType]
             [/j "TEXT_SEARCH" [,servicename] [,portnumber] [/?]
```

## コマンド・パラメーター

### Linux および UNIX オペレーティング・システムの場合

**-d** デバッグ・モードをオンにします。このオプションは、DB2 データベース・サポートからの指示があった場合にのみ使用してください。

**-k db2iupgrade** コマンドの実行元の DB2 コピーでサポートされている場合は、アップグレード前のインスタンス・タイプを保持します。このパラメーターが指定されていない場合、インスタンス・タイプは、サポートされるデフォルトのインスタンス・タイプにアップグレードされます。

#### **-j "TEXT\_SEARCH"**

DB2 Text Search サーバーを、サービス名と TCP/IP ポート番号に対する生成済みのデフォルト値を使用して構成します。インスタンス・タイプがクライアントの場合、このパラメーターは使用できません。

#### **-j "TEXT\_SEARCH, servicename"**

DB2 Text Search サーバーを、提供されたサービス名と自動的に生成されたポート番号を使用して構成します。サービス名に `services` ファイル内で割り当てられたポート番号がある場合、それはその割り当てられたポート番号を使用します。

#### **-j "TEXT\_SEARCH, servicename, portnumber"**

DB2 Text Search サーバーを、提供されたサービス名とポート番号を使用して構成します。

#### **-j "TEXT\_SEARCH, portnumber"**

DB2 Text Search サーバーを、デフォルトのサービス名と提供されたポート番号を使用して構成します。有効なポート番号は 1024 - 65535 の範囲になければなりません。

#### **-a AuthType**

インスタンスの認証タイプ (SERVER、CLIENT、または SERVER\_ENCRYPT) を指定します。デフォルトは SERVER です。

#### **-u FencedID**

fenced ユーザー定義関数および fenced ストアード・プロシージャを実行するユーザー ID の名前を指定します。このオプションは、DB2 クライアント・インスタンスを DB2 サーバー・インスタンスにアップグレードする際に必要です。

#### *InstName*

インスタンスの名前を指定します。

### Windows オペレーティング・システムの場合

#### *InstName*

インスタンスの名前を指定します。

**/u:username,password**

DB2 サービスのアカウント名およびパスワードを指定します。パーティション・インスタンスをアップグレードする場合には、このオプションは必須です。

**/p:instance-profile-path**

アップグレードされるインスタンス用の新しいインスタンス・プロファイル・パスを指定します。

**/q db2iupgrade** コマンドを静止モードで発行します。

**/a:authType**

インスタンスの認証タイプ (SERVER、CLIENT、または SERVER\_ENCRYPT) を指定します。

**/j "TEXT\_SEARCH"**

DB2 Text Search サーバーを、サービス名と TCP/IP ポート番号に対する生成済みのデフォルト値を使用して構成します。インスタンス・タイプがクライアントの場合、このパラメーターは使用できません。

**/j "TEXT\_SEARCH, servicename"**

DB2 Text Search サーバーを、提供されたサービス名と自動的に生成されたポート番号を使用して構成します。サービス名に `services` ファイル内で割り当てられたポート番号がある場合、それはその割り当てられたポート番号を使用します。

**/j "TEXT\_SEARCH, servicename, portnumber"**

DB2 Text Search サーバーを、提供されたサービス名とポート番号を使用して構成します。

**/j "TEXT\_SEARCH, portnumber"**

DB2 Text Search サーバーを、デフォルトのサービス名と提供されたポート番号を使用して構成します。有効なポート番号は 1024 - 65535 の範囲になければなりません。

**/? db2iupgrade** コマンドの使用方法を表示します。

## 使用上の注意

**db2iupgrade** コマンドを使用してアップグレードできるのは、DB2 Enterprise Server Edition のインスタンス (インスタンス・タイプ `ese`) のみです。

**db2iupgrade** コマンドは、**db2ckupgrade** コマンドを **-not1** パラメーターを指定して呼び出し、**db2ckupgrade** のログ・ファイルとして `update.log` を指定します。

**db2iupgrade** 用に作成されるデフォルトのログ・ファイルは `/tmp/db2ckupgrade.log.processID` です。インスタンスをアップグレードする前に、ローカル・データベースがアップグレードの準備ができていることを確認してください。**-not1** パラメーターは、タイプ 1 索引の検査を使用不可にします。ログ・ファイルは、Linux および UNIX オペレーティング・システムの場合はインスタンスのホーム・ディレクトリー、Windows オペレーティング・システムの場合は現行ディ

レクトリに作成されます。 **db2ckupgrade** コマンドが何らかのエラーを戻した場合、インスタンスのアップグレードは続行されません。

パーティション・データベース環境では、**db2iupgrade** コマンドを実行する前に、**db2ckupgrade** コマンドを実行します。 **db2ckupgrade** コマンドは、すべてのパーティションを検査して、パーティションで検出されたエラーを返します。アップグレードの準備ができていないかすべてのデータベース・パーティションをチェックしない場合、インスタンスのアップグレードが成功したとしても、それに続くデータベースのアップグレードで失敗する可能性があります。詳しくは、**db2ckupgrade** を参照してください。

#### Linux および UNIX オペレーティング・システムの場合

- **db2iupgrade** コマンドを使用することによって旧バージョンの DB2 インスタンスを DB2 データベース・システムの現行バージョンにアップグレードする場合、旧 DB2 データベース・インストール・パス内で定義されている DB2 グローバル・プロファイル変数は、新しいインストール・ロケーションにアップグレードされません。インスタンス固有の DB2 インスタンス・プロファイル変数のアップグレードは、インスタンスのアップグレード後に実行されます。
- root ユーザーになるために **login** コマンドではなく **su** コマンドを使用している場合は、**su** コマンドを **-** オプションを指定して発行し、あたかも **login** コマンドを使用してシステムにログインしているかのように、プロセス環境を設定することを指示する必要があります。
- root ユーザー用の DB2 インスタンス環境を入手しないでください。DB2 インスタンス環境を入手している場合に、**db2iupgrade** コマンドを実行することはサポートされていません。
- AIX 6.1 (またはそれ以上) では、このコマンドは、システム・ワークロード・パーティション (WPAR) グローバル環境内の共有 DB2 コピーから実行する場合、root ユーザーとして実行されなければなりません。WPAR は DB2 pureScale 環境ではサポートされていません。

---

## db2icrt - インスタンスの作成

DB2 pureScale インスタンスを含めて DB2 インスタンスを作成します。また、このコマンドを使用して、DB2 pureScale インスタンスの作成の一環として、初期 DB2 メンバーおよびクラスター・キャッシング・ファシリティーを作成することもできます。

Linux および UNIX オペレーティング・システムでは、**db2icrt** は **DB2DIR/instance** にあります。 **DB2DIR** は、DB2 データベース・システムがインストールされているインストール・ディレクトリを表します。 Windows オペレーティング・システムでは、**db2icrt** は **DB2PATH\bin** にあります。 **DB2PATH** は DB2 コピーがインストールされているディレクトリです。

**db2icrt** コマンドは、インスタンス所有者のホーム・ディレクトリに DB2 インスタンスを作成します。 DB2 pureScale インスタンスは、DB2 pureScale 環境ごとに 1 つのみ作成できます。



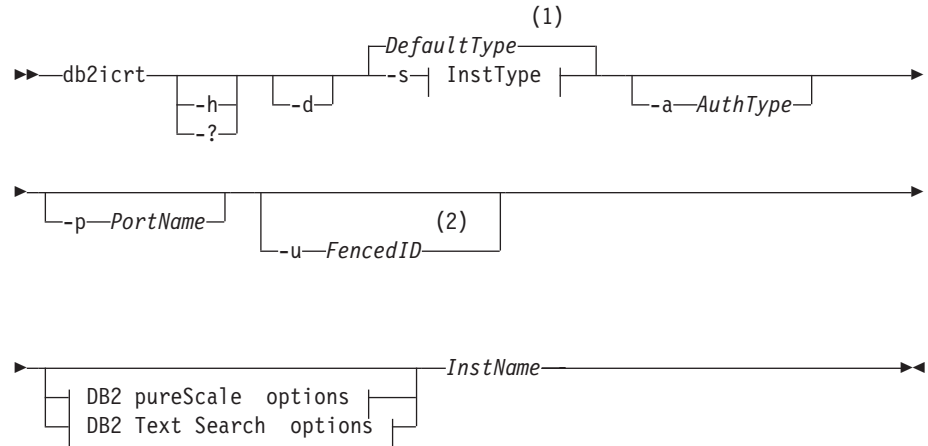
注: このコマンドは、DB2 データベース製品の非 root インストールには利用できません。

## 許可

Linux および UNIX オペレーティング・システムでは root ユーザー権限、Windows オペレーティング・システムではローカル管理者権限が必要です。

## コマンド構文

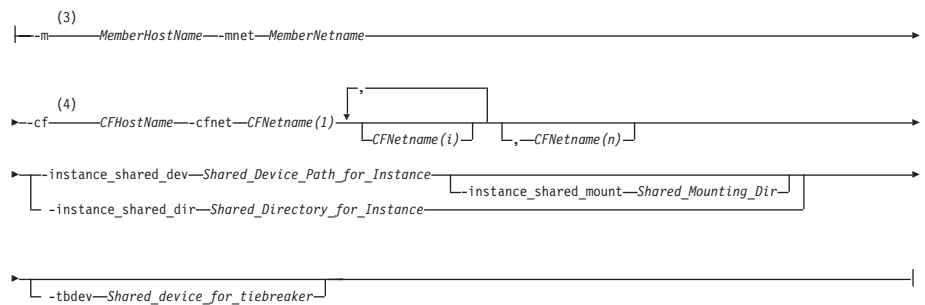
Linux および UNIX オペレーティング・システムの場合



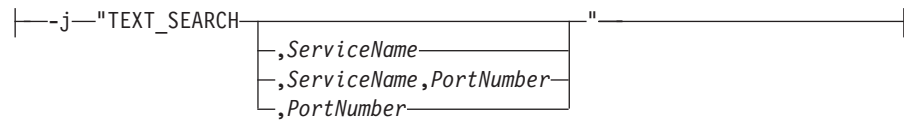
### InstType:



### DB2 pureScale options:



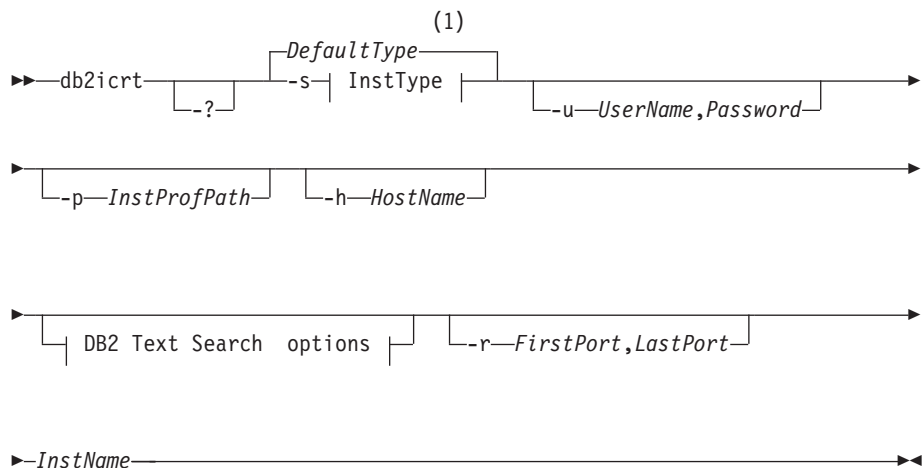
## DB2 Text Search options:



### 注:

- 1 インスタンス・タイプを `-s` で指定しなかった場合には、作成されるデフォルトのインスタンス・タイプは、DB2 コピーのライセンスで作成が許可されているタイプのうち、*InstType* にリストされているタイプの中で最も高位のタイプです。
- 2 クライアント・インスタンスを作成する場合、`-u FencedID` は有効なオプションではありません。
- 3 *MemberHostName:MemberNetname* 形式は、`-m` オプションでは推奨されなくなりました。今後廃止される可能性があります。DB2 pureScale Feature での IPv6 サポートには、`-m` および `-mnet` オプションの両方を使用する新しい形式が必要です。
- 4 *CFHostName:CFNetames* 形式は、`-cf` オプションでは推奨されなくなりました。今後廃止される可能性があります。DB2 pureScale Feature での IPv6 サポートには、`-cf` および `-cfnet` オプションの両方を使用する新しい形式が必要です。

## Windows オペレーティング・システムの場合



### InstType:



## DB2 Text Search options:

```
|---j---"TEXT_SEARCH"-----|
|                               |
|---,ServiceName-----|
|---,ServiceName,PortNumber---|
|---,PortNumber-----|
```

### 注:

- 1 インスタンス・タイプを `-s` で指定しなかった場合には、作成されるデフォルトのインスタンス・タイプは、DB2 コピーのライセンスで作成が許可されているタイプのうち、*InstType* にリストされているタイプの中で最も高位のタイプです。

## コマンド・パラメーター

### Linux および UNIX オペレーティング・システムの場合

- ? 使用情報を表示します。
- h 使用情報を表示します。
- d デバッグ・モードをオンにします。db2icrt.trc.ProcessID として、/tmp 内にデフォルト名でトレース・ファイルを保存します。このオプションは、DB2 データベース・サポートからの指示があった場合にのみ使用してください。
- a *AuthType*  
インスタンスの認証タイプ (SERVER、CLIENT、または SERVER\_ENCRYPT) を指定します。デフォルトは SERVER です。
- j "TEXT\_SEARCH"  
DB2 Text Search サーバーを、生成されたサービス名と TCP/IP ポート番号のデフォルト値を使用して構成します。インスタンス・タイプがクライアント の場合、このパラメーターは使用できません。
- j "TEXT\_SEARCH, servicename"  
DB2 Text Search サーバーを、提供されたサービス名と自動的に生成されたポート番号を使用して構成します。サービス名に services ファイル内で割り当てられたポート番号がある場合、それはその割り当てられたポート番号を使用します。
- j "TEXT\_SEARCH, servicename, portnumber"  
DB2 Text Search サーバーを、提供されたサービス名とポート番号を使用して構成します。
- j "TEXT\_SEARCH, portnumber"  
DB2 Text Search サーバーを、デフォルトのサービス名と提供されたポート番号を使用して構成します。有効なポート番号は 1024 - 65535 の範囲になければなりません。
- p <TCP/IP PortName>  
インスタンスが使用する TCP/IP ポート名または番号を指定しま

す。さらに、このオプションによって、データベース・マネージャ構成パラメーター **SVCENAME** を DB2 インスタンスに対して構成します。

**-m** *MemberHostName:NetName1*

インスタンスの作成時に DB2 メンバーとしてセットアップするホストを指定します。このパラメーターは、DB2 pureScale 環境で必須です。 **db2icrt** コマンドでセットアップできる DB2 メンバーは 1 つだけです。 **db2iupdt -add** コマンドを使用して、さらに DB2 メンバーを追加することができます。 *NetName1* 構文は推奨されなくなりました。今後のリリースで廃止される可能性があります。代わりに **-mnet** パラメーターを使用してください。

*MemberHostName* は正規のホスト名にしてください (例えば、'hostname' コマンドの出力はローカル・ホスト上で実行します)。ここに指定される *NetName1* 値は、**-cf** パラメーターで指定されているのと同じサブネットに属している必要があります。

**-mnet** *MemberNetName*

このパラメーターは、推奨されなくなった **-m** *MemberHostName:NetName1* パラメーターの *:NetName1* 構文に取って代わるものです。クラスター相互接続ネット名を指定します。これは、DB2 pureScale インスタンスで、メンバーとクラスター・キャッシング・ファシリティー (CF と呼ばれる) との高速通信に使用される相互接続のホスト名です。

**-cf** *CFHostName:NetName2*

インスタンス作成時にクラスター・キャッシング・ファシリティー (CF ともいう) としてセットアップするホストを指定します。このパラメーターは、DB2 pureScale 環境で必須です。 **db2icrt** コマンドでセットアップできる CF は 1 つだけです。 **db2iupdt -add** コマンドを使用して、さらに CF を追加することができます。 *NetName2* 構文は推奨されなくなりました。今後のリリースで廃止される可能性があります。代わりに **-cfnet** パラメーターを使用してください。

**-cfnet** *CFNetName*

このパラメーターは、推奨されなくなった **-cf** *CFHostName:NetName2* パラメーターの *:NetName2* 構文に取って代わるものです。クラスター相互接続ネット名を指定します。これは、DB2 pureScale インスタンスで、メンバーと CF との高速通信に使用される相互接続のホスト名です。

*CFNetName* は、**-m** パラメーターで指定されたサブネットと同じサブネットに属していて、クラスター相互接続ネット名に対応している必要があります (例えば *db2\_<hostname\_ib0>*)。

**-instance\_shared\_dev** *Shared\_Device\_Path\_for\_Instance*

インスタンス共有ファイルとデフォルトのデータベース・パスを保持する DB2 pureScale インスタンスをセットアップするのに必要な共有ディスク装置パスを指定します。例えば、*dev/hdisk1* と指定します。DB2 pureScale インスタンスについて、共有ディレクトリー

は、すべてのホストでアクセス可能でなければなりません。このオプションの値は、**-tbdev** オプションと同じ値であってはなりません。

**-instance\_shared\_dev** パラメーターを指定すると、DB2 インストーラーによって、DB2 クラスタ・ファイル・システムが作成されます。

**-instance\_shared\_dev** パラメーターと **-instance\_shared\_dir** パラメーターは、相互に排他的です。

**-instance\_shared\_mount** *Shared\_Mounting\_Dir*

新規 IBM General Parallel File System (GPFS™) ファイル・システム用のマウント・ポイントを指定します。指定するパスは、新規で空のパスでなければならず、既存の GPFS ファイル・システムの内部にネストされたパスであってはなりません。

**-instance\_shared\_dir** *Shared\_Directory\_for\_Instance*

インスタンス共有ファイルとデフォルトのデータベース・パスを保持する DB2 pureScale インスタンスをセットアップするのに必要な、共有ファイル・システム (GPFS) 内のディレクトリーを指定します。例えば、/sharedfs となります。DB2 pureScale インスタンスについて、そのディスクはすべてのホストでアクセス可能でなければなりません。このオプションの値は、**-tbdev** オプションと同じ値であったり、インストール・パスであってはなりません。

**-instance\_shared\_dir** パラメーターを指定すると、DB2 インストーラーによって、ユーザー管理ファイル・システムが作成されます。ユーザー管理のファイル・システムはすべてのホストで使用可能でなければならず、GPFS ファイル・システムでなければなりません。

**-instance\_shared\_dir** パラメーターと **-instance\_shared\_dev** パラメーターは、相互に排他的です。

**-tbdev** *Shared\_device\_for\_tiebreaker*

データの保全性を確実に維持するために、DB2 pureScale 環境でタイブレーカーとして機能するデバイスの共有装置パスを指定します。このオプションの値は、**-instance\_shared\_dev** オプションまたは **-instance\_shared\_dir** オプションと同じ値であってはなりません。DB2 クラスタ・サービスのタイブレーカーが初めて作成されるときには、このオプションは必須です。ディスク装置には、それに関連付けられたファイル・システムがあってはなりません。DB2 クラスタ・サービスのピア・ドメインが既に存在する場合は、このオプションは無効です。

**-s** *InstType*

作成するインスタンスのタイプを指定します。**-s** オプションは、**db2icrt** を実行しているインストール済みの製品に関連したデフォルト以外のインスタンスを作成する場合にのみ指定してください。有効な値は以下のとおりです。

**dsf** ローカルおよびリモート・クライアントで DB2 データベース・サーバーの DB2 pureScale インスタンスを作成するた

めに使用します。このオプションは、IBM DB2 pureScale Feature のデフォルトのインスタンス・タイプです。

**ese** ローカルおよびリモート・クライアントでデータベース・サーバーのインスタンスを作成するために使用します。このオプションは、DB2 Enterprise Server Edition または DB2 Advanced Enterprise Server Edition のデフォルトのインスタンス・タイプです。

**wse** ローカルおよびリモート・クライアントでデータベース・サーバーのインスタンスを作成するために使用します。このオプションは DB2 Workgroup Server Edition、DB2 Express Edition または DB2 Express-C、 および DB2 Connect™ Enterprise Edition のデフォルトのインスタンス・タイプです。

**standalone**

ローカル・クライアントでデータベース・サーバーのインスタンスを作成するために使用します。これは、DB2 Personal Edition のデフォルトのインスタンス・タイプです。

**client** クライアントのインスタンスを作成するために使用します。このオプションは IBM Data Server Client、IBM Data Server Runtime Client、 および DB2 Connect Personal Edition のデフォルトのインスタンス・タイプです。

DB2 データベース製品は、デフォルトのインスタンス・タイプと、デフォルトより下位のインスタンス・タイプをサポートしています。例えば、DB2 Enterprise Server Edition は、ese、wse、standalone、 および client のインスタンス・タイプをサポートしています。

**-u Fenced ID**

fenced ユーザー定義関数および fenced ストアード・プロシージャを実行するユーザー ID の名前を指定します。クライアント・インスタンスを作成しない場合は、**-u** オプションは必須です。

*InstName*

オペレーティング・システムの既存ユーザー名であるインスタンス名を指定します。インスタンス名は、**db2icrt** コマンドの最後の引数でなければなりません。

**Windows オペレーティング・システムの場合**

*InstName*

インスタンスの名前を指定します。

**-s InstType**

作成するインスタンスのタイプを指定します。現在 4 種類の DB2 インスタンス・タイプがあります。有効な値は以下のとおりです。

**client** クライアントのインスタンスを作成するために使用します。このオプションは IBM Data Server Client、IBM Data Server Runtime Client、 および DB2 Connect Personal Edition のデフォルトのインスタンス・タイプです。

### **standalone**

ローカル・クライアントでデータベース・サーバーのインスタンスを作成するために使用します。これは、DB2 Personal Edition のデフォルトのインスタンス・タイプです。

**ese** パーティション・データベース環境サポートのあるローカルおよびリモート・クライアントでデータベース・サーバーのインスタンスを作成するために使用します。

*-s ese -u Username, Password*

ESE インスタンス・タイプおよびパーティション・データベース環境インスタンスを作成するには、**db2icrt** でオプションを使用する必要があります。

**wse** ローカルおよびリモート・クライアントでデータベース・サーバーのインスタンスを作成するために使用します。このオプションは DB2 Workgroup Server Edition、DB2 Express Edition または DB2 Express-C、 および DB2 Connect Enterprise Edition のデフォルトのインスタンス・タイプです。

DB2 データベース製品は、デフォルトのインスタンス・タイプと、デフォルトより下位のインスタンス・タイプをサポートしています。例えば、DB2 Enterprise Server Edition は、ese、wse、standalone、および client のインスタンス・タイプをサポートしています。

### **-u Username, Password**

DB2 サービスのアカウント名およびパスワードを指定します。このオプションはパーティション・データベース・インスタンスを作成する時に必要です。

### **-p InstProfPath**

インスタンス・プロファイル・パスを指定します。

### **-h HostName**

現行のマシンに対して複数のデフォルトの TCP/IP ホスト名がある場合、それらをオーバーライドします。TCP/IP ホスト名は、デフォルト・データベース・パーティション (データベース・パーティション 0) を作成する際に使用されます。このオプションは、パーティション・データベース・インスタンスに対してのみ有効です。

### **-r PortRange**

MPP モードで実行する場合に、パーティション・データベース・インスタンスによって使用される TCP/IP ポートの範囲を指定します。例えば、**-r 50000,50007**。このオプションを指定した場合、ローカル・マシンの **services** ファイルは、以下の項目で更新されます。

DB2_InstName	baseport/tcp
DB2_InstName_END	endport/tcp

### **/j "TEXT\_SEARCH"**

DB2 Text Search サーバーを、生成されたサービス名と TCP/IP ポ

ート番号のデフォルト値を使用して構成します。インスタンス・タイプがクライアント の場合、このパラメーターは使用できません。

**/j "TEXT\_SEARCH, servicename"**

DB2 Text Search サーバーを、提供されたサービス名と自動的に生成されたポート番号を使用して構成します。サービス名に `services` ファイル内で割り当てられたポート番号がある場合、それはその割り当てられたポート番号を使用します。

**/j "TEXT\_SEARCH, servicename, portnumber"**

DB2 Text Search サーバーを、提供されたサービス名とポート番号を使用して構成します。

**/j "TEXT\_SEARCH, portnumber"**

DB2 Text Search サーバーを、デフォルトのサービス名と提供されたポート番号を使用して構成します。有効なポート番号は 1024 - 65535 の範囲になければなりません。

**-?** 使用法情報を表示します。

## 例

1. インスタンス所有者 `db2sdin1`、fenced ユーザー `db2sdfel` に対して DB2 pureScale インスタンスを作成するには、次のコマンドを実行します。

```
DB2DIR/instance/db2icrt
-cf host1.domain.com -cfnet host1.domain.com-ib0
-m host2.domain.com -mnet host2.domain.com-ib0
-instance_shared_dev /dev/hdisk1
-tbdev /dev/hdisk2
-u db2sdfel
db2sdin1
```

ここで、*DB2DIR* は、DB2 のコピーのインストール・ロケーションを示します。DB2 pureScale インスタンスである `db2sdin1` は、`host1` 上に CF を持ち、`host2` 上にメンバーを持つことになります。さらにこのコマンドは `/dev/hdisk1` を使用して、インスタンス共有ファイルを保管するための共有ファイル・システムを作成し、タイブレーカーの共有装置パスとして `/dev/hdisk2` をセットアップします。

2. ユーザー ID `db2inst1` に対して DB2 Enterprise Server Edition インスタンスを作成するには、次のコマンドを実行します。

```
DB2DIR/instance/db2icrt -s ese -u db2fenc1 db2inst1
```

*DB2DIR* は、DB2 コピーのインストール場所を表します。

3. インスタンス所有者 `db2sdin1` および fenced ユーザー `db2sdfel` で、DB2 製品によって管理される既存のファイル・システム (GPFS) を使用する DB2 pureScale インスタンスを作成するには、以下のコマンドを実行します。

```
DB2DIR/instance/db2icrt
-cf host1.domain.com -cfnet host1.domain.com-ib0
-m host2.domain.com -mnet host2.domain.com-ib0
-tbdev /dev/hdisk2
-u db2sdfel
db2sdin1
```



*DB2DIR* は、DB2 コピーのインストール場所を表します。

4. インスタンス所有者 *db2sdin1* および *fenced* ユーザー *db2sdfel* で、ユーザー管理の既存の GPFS ファイル・システム (*/gpfs\_shared\_dir*) を使用して DB2 pureScale インスタンスを作成するには、次のコマンドを実行します。

```
DB2DIR/instance/db2icrt
-cf host1.domain.com -cfnet host1.domain.com-ib0
-m host2.domain.com -mnet host2.domain.com-ib0
-instance_shared_dir /gpfs_shared_dir
-tbdev /dev/hdisk2
-u db2sdfel
db2sdin1
```

ここで、*DB2DIR* は、DB2 のコピーのインストール・ロケーションを示します。

5. AIX マシン上でユーザー ID *db2inst1* のインスタンスを作成するには、以下のコマンドを発行します。

クライアント・マシンの場合:

```
DB2DIR/instance/db2icrt db2inst1
```

サーバー・マシンの場合:

```
DB2DIR/instance/db2icrt -u db2fenc1 db2inst1
```

ここで、*db2fenc1* は、*fenced* ユーザー定義関数および *fenced* ストアード・プロシージャを実行するユーザー ID です。

## 使用上の注意

- インスタンス・ユーザーはすべてのホスト上に存在していて、それらは UID、GID、グループ名、およびホーム・ディレクトリー・パスが同じでなければなりません。同じ規則が *fenced* ユーザーに適用されます。 **db2icrt** コマンドが正常に実行された後、DB2 インストーラーは、全ホストにわたってインスタンス・ユーザーのための SSH をセットアップします。
- **db2icrt** コマンドを使用するときは、インスタンスの名前が既存ユーザーの名前に一致している必要があります。
- DB2 pureScale環境ごとに、存在できるインスタンスは 1 つだけです。
- DB2 インスタンスを作成する際には、以下の制限事項を考慮してください。
  - 既存の ID を使用して DB2 インスタンスを作成する場合は、ID がロックされておらず、パスワードが期限切れでないことを確認してください。
  - グループ ID (GID) は、100-65535 の範囲になければなりません。
- また、**db2isetup** コマンドを使用すると、グラフィカル・インターフェースで、DB2 インスタンスの作成および更新、複数ホストの追加を実行することができます。
- root ユーザーになるために **login** コマンドではなく **su** コマンドを使用している場合は、**su** コマンドを - オプションを指定して発行して、あたかも **login** コマンドを使用してシステムにログインしているかのように、プロセス環境を設定することを指示する必要があります。

- root ユーザー用の DB2 インスタンス環境を入手しないでください。DB2 インスタンス環境を入手している場合に、**db2icrt** を実行することはサポートされていません。
- 以前に DB2 pureScale インスタンスを作成してドロップしたことがある場合、**-instance\_shared\_dev** パラメーター指定を使ってそれを再作成することはできません。それは、DB2 クラスタ・ファイル・システムが既に作成されている可能性があるからです。以前に作成された共有ファイル・システムを指定するには、以下のようにします。
  - 既存の GPFS 共有ファイル・システムが DB2 pureScale Feature によって作成および管理されている場合には、**-instance\_shared\_dev** パラメーターと **-instance\_shared\_dir** パラメーターはいずれも使用しないでください。
  - 既存の GPFS 共有ファイル・システムが DB2 pureScale Feature によって作成および管理されているのではない場合には、**-instance\_shared\_dir** パラメーターを使用してください。
- AIX 6.1 (またはそれ以上) では、このコマンドは、システム・ワークロード・パーティション (WPAR) グローバル環境内の共有 DB2 コピーから実行する場合、root ユーザーとして実行されなければなりません。WPAR は DB2 pureScale 環境ではサポートされていません。

## db2idrop - インスタンスの除去

**db2icrt** によって作成された DB2 インスタンスを除去します。

ドロップできるインスタンスは、**db2idrop** コマンドを発行している発行元と同じ DB2 コピーに関する **db2ilist** コマンドによってリスト表示されるインスタンスだけです。**db2idrop** コマンドを使用して DB2 pureScale インスタンスをドロップすることもできます。

Linux および UNIX オペレーティング・システムでは、このユーティリティーは **DB2DIR/instance** ディレクトリーにあります。**DB2DIR** は、DB2 データベース・システムの現行バージョンがインストールされている場所です。Windows オペレーティング・システムでは、このユーティリティーは **DB2PATH%bin** ディレクトリーにあります。**DB2PATH** は、DB2 コピーがインストールされている場所です。

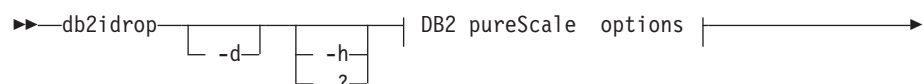
注: このコマンドを使用して、Linux および UNIX オペレーティング・システムに非 root インストールされた DB2 インスタンスをドロップすることはできません。非ルート DB2 コピーをアンインストールするオプションのみあります。詳しくは、後述の『使用上の注意』というセクションを参照してください。

### 許可

Linux および UNIX オペレーティング・システムでは root ユーザー権限、Windows オペレーティング・システムではローカル管理者でアクセスします。

### コマンド構文

Linux および UNIX オペレーティング・システムの場合



▶ | Outside Of DB2 pureScale options | *InstName* →

### DB2 pureScale options:

|  -g |

### DB2 pureScale 外のオプション:

|  -f |

### Windows オペレーティング・システムの場合

▶▶ db2idrop  -f  -h *InstName* →

## コマンド・パラメーター

### Linux および UNIX オペレーティング・システムの場合

**-d** DB2 データベース・サポートによって使用されるデバッグ・モードに入ります。

**-f** このパラメーターは、推奨されていません。

強制アプリケーション・フラグを指定します。このフラグを指定すると、このインスタンスを使用しているすべてのアプリケーションが強制的に終了させられます。このパラメーターは DB2 pureScale 環境ではサポートされません。

**-g** **db2idrop** が DB2 pureScale インスタンスで使用される場合には、このパラメーターは必須です。すべてのホスト上で DB2 pureScale インスタンスをドロップすることを指定します。このパラメーターは、DB2 pureScale インスタンス内のすべてのホストで、すべての DB2 メンバーと、すべてのクラスター・キャッシング・ファシリティーが停止されていることを必要とします。他のすべてのインスタンス・タイプのドロップについては、このオプションは無視されます。

**-h | -?**

使用情報を表示します。

*InstName*

インスタンスの名前を指定します。

### Windows オペレーティング・システムの場合

*InstName*

インスタンスの名前を指定します。

**-f** 強制アプリケーション・フラグを指定します。このフラグを指定すると、このインスタンスを使用しているすべてのアプリケーションが強制的に終了させられます。

**-h** 使用法情報を表示します。

## 例

Linux または UNIX オペレーティング・システム上で、以下のコマンドを発行することによって `db2inst1` を作成した場合、

```
/opt/IBM/db2/copy1/instance/db2icrt -u db2fenc1 db2inst1
```

`db2inst1` をドロップするには、以下のコマンドを実行する必要があります。

```
/opt/IBM/db2/copy1/instance/db2idrop db2inst1
```

## 使用上の注意

- インスタンスをドロップする前に、すべてのホストで DB2 データベース・マネージャが停止していること、およびそのインスタンスにアクセスする DB2 データベース・アプリケーションが切断され、終了していることを確認してください。必要なら、そのインスタンスに関連する DB2 データベースをバックアップし、構成データを将来参照できるように保存することができます。
- **db2idrop** コマンドではデータベースは除去されません。データベースがもはや不要であれば、まずそれを除去してください。データベースを除去しない場合、それらはいつでも同じリリースの別の DB2 コピーの下でカタログして、使用を継続することができます。
- DB2 Text Search の構成を保存して、インスタンス・データベースを再利用する予定がある場合、**db2idrop** コマンドを発行する前に、`config` ディレクトリー (UNIX では `instance_home/sqllib/db2tss/config`、Windows では `instance_profile_path¥instance_name¥db2tss¥config`) または `config` ディレクトリー内容を保存するという、追加のステップを実行する必要があります。新しいインスタンスが作成された後に、`config` ディレクトリーをリストアできます。ただし、`config` ディレクトリーのリストアが有効になるのは、作成される新しいインスタンスが同じリリースおよびフィックスパック・レベルのものである場合だけです。
- Linux および UNIX オペレーティング・システム上で非 `root` インストールされたインスタンスをドロップすることはできません。ユーザーがこの DB2 インスタンスを除去するには、**db2\_deinstall -a** を実行して DB2 の非ルート・コピーをアンインストールするオプションのみ使用できます。
- Linux および UNIX オペレーティング・システムの場合、`root` ユーザーになるために `login` コマンドではなく `su` コマンドを使用している場合は、`su` コマンドを `-` オプションを指定して発行し、あたかも `login` コマンドを使用してシステムにログインしているかのように、プロセス環境を設定することを指示する必要があります。
- Linux および UNIX オペレーティング・システムの場合、`root` ユーザー用の DB2 インスタンス環境を入手しないでください。DB2 インスタンス環境を入手している場合に、**db2idrop** を実行することはサポートされていません。
- DB2 pureScale環境では、**-g** パラメーターは必須です。この場合、すべてのホストでインスタンスがドロップされます。ただし、インストール開始ホスト (IH) 上の IBM General Parallel File System (GPFS) については削除されず、GPFS ファイル・システムについても同様です。ファイル・システムを手動で削除し、GPFS をアンインストールする必要があります。

- Windows オペレーティング・システムでは、インスタンスが Microsoft Cluster Service (MSCS) でクラスター化されている場合、**db2mcs** または **db2iclus** コマンドを発行してそのインスタンスをクラスター解除してからインスタンスを廃棄できます。
- AIX 6.1 (またはそれ以上) では、このコマンドは、システム・ワークロード・パーティション (WPAR) グローバル環境内の共有 DB2 コピーから実行する場合、root ユーザーとして実行されなければなりません。WPAR は DB2 pureScale 環境ではサポートされていません。

## db2iupdt - インスタンスの更新

**db2iupdt** コマンドを使用して、リリース内のさらに高いレベルにインスタンスを更新したり、DB2 pureScaleインスタンス以外のインスタンスを DB2 pureScaleインスタンスへ更新したり、DB2 pureScaleインスタンスのスケールを変えたりできます。

**db2iupdt** コマンドを実行する前に、インスタンスを停止し、そのインスタンスについて実行されているすべてのプロセスを停止する必要があります。このコマンドを使用することにより DB2 pureScaleインスタンス以外のインスタンスを更新してメンバーまたはクラスター・キャッシング・ファシリティを追加する場合は、前提条件とプリインストール・チェックリストを検討して、インスタンスおよびホストが準拠していることを確認してください。

**注:** DB2 pureScale インスタンスでは、構成クォーラム に達していないと、つまり絶対多数のノードがオンラインになっていないとリソース・モデルに変更を加えることができません。2 つのホストで構成されるセットアップの場合、いずれかがオフラインになっていると **db2iupdt** コマンドを使用できません。

### 許可

UNIX および Linux オペレーティング・システムでは root ユーザー権限、Windows オペレーティング・システムではローカル管理者。

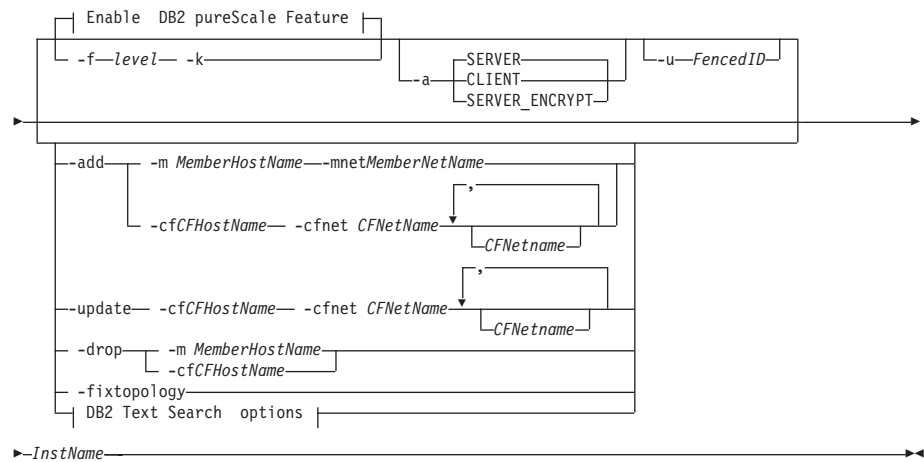
### コマンド構文

UNIX および Linux オペレーティング・システムの場合

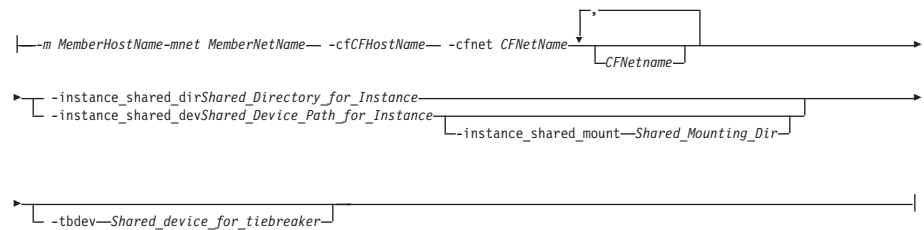
```

▶▶ db2iupdt [-h] [-d] [-D]
             [-?]

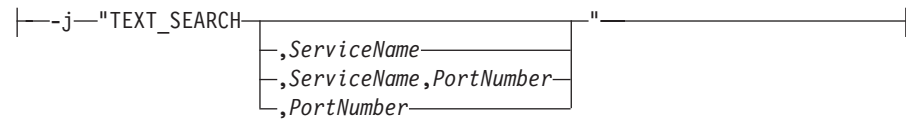
```



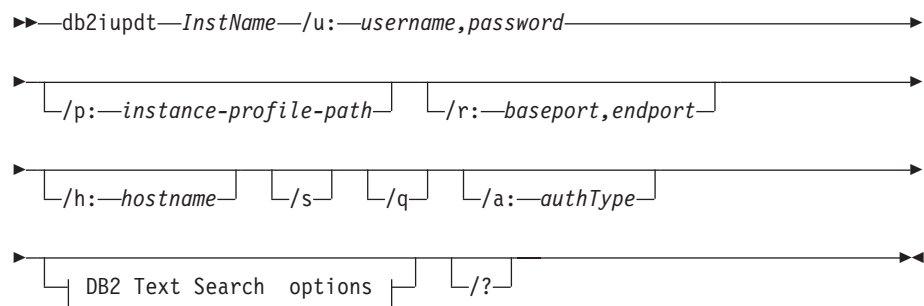
### Enable DB2 pureScale Feature:



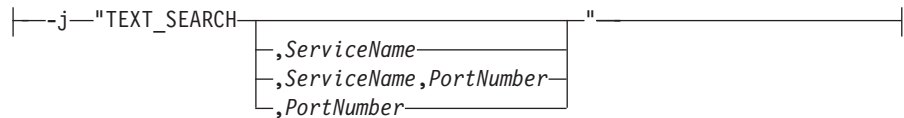
### DB2 Text Search options:



### Windows オペレーティング・システムの場合



### DB2 Text Search options:



## コマンド・パラメーター

### UNIX および Linux オペレーティング・システムの場合

**-h | -?**

使用情報を表示します。

**-a AuthType**

インスタンスの認証タイプ (SERVER、SERVER\_ENCRYPT、または CLIENT) を指定します。デフォルトは SERVER です。

**-d** デバッグ・モードをオンにします。

**-k** 更新時に現行のインスタンス・タイプが変更されないようにします。

**-D** このパラメーターは推奨されなくなりました。今後のリリースで除去される可能性があります。このパラメーターは **-f level** パラメーターに置き換えられます。あるパスにインストールされた高いコード・レベルのインスタンスから別のパスにインストールされた低いコード・レベルのインスタンスに更新します。

**-f level**

互換性のために、高い DB2 バージョンのインスタンス・タイプから、低い DB2 バージョンのインスタンス・タイプへ、インスタンスを更新します。

**-add -m MemberHostName:Netname1 | -cf CFHostName:Netname2**

DB2 pureScale インスタンスに追加するホスト (メンバーまたはクラスター・キャッシング・ファシリティー) を指定します。

追加するホストのタイプを指定するには、メンバーの場合は **-m** を、クラスター・キャッシング・ファシリティーの場合は **-cf** を使用します。両方ではなく、一方を選択する必要があります。

**db2iupdt -add** コマンドは、既に DB2 pureScale インスタンスの一部になっているホストから実行する必要があります。

**-m MemberHostName:NetName1** および **-cf CFHostName:NetName2** 構文は推奨されなくなりました。今後のリリースでは除去される可能性があります。代わりに、構文 **-add -m MemberHostName -mnet MemberNetName** または **-add -cf CFHostName -cfnet CFNetName** を使用してください。

**-add -m MemberHostName -mnet MemberNetName | -add -cf CFHostName -cfnet CFNetName**

クラスター相互接続ネット名を指定します。これは、DB2 メンバーとクラスター・キャッシング・ファシリティーとの高速通信に使用される相互接続のホスト名です。追加する各ホストのネット名は、インスタンス内のすべての既存のメンバーまたは CF と同じサブネット上にある必要があります。

追加するホストのタイプを指定するには、メンバーの場合は **-m MemberHostName -mnet MemberNetName** を、クラスター・キャッシング・ファシリティーの場合は **-cf CFHostName -cfnet CFNetName** を使用します。両方ではなく、一方を選択する必要があります。

**db2iupdt -add** コマンドは、既に DB2 pureScale インスタンスの一部になっているホストから実行する必要があります。

**-add** DB2 pureScale Feature インスタンスに追加するホストのホスト名およびクラスター相互接続ネット名を指定します。 **db2iupdt -add** コマンドは、既に DB2 pureScale インスタンスの一部になっているホストから実行する必要があります。

**-m MemberHostName -mnet MemberNetName**

クラスター相互接続ネット名 *MemberNetName* を指定して、ホスト名 *MemberHostName* のホストを、DB2 pureScale Feature インスタンスに追加します。

**-cf CFHostName -cfnet CFNetName**

クラスター相互接続ネット名 *CFNetName* を指定して、ホスト名 *CFHostName* のホストを、DB2 pureScale Feature インスタンスにクラスター・キャッシング・ファシリティーとして追加します。 *CFHostName* に複数のクラスター相互接続ネットワーク・アダプター・ポートがある場合は、各クラスター相互接続ネット名をコンマで区切ったリストを *CFNetName* に指定することもできます。

**-drop -m MemberHostName | -cf CFHostName**

DB2 pureScale インスタンスからドロップするホスト (メンバーまたはクラスター・キャッシング・ファシリティー) を指定します。ドロップするホストのタイプを指定するには、メンバーの場合は **-m** オプションを、クラスター・キャッシング・ファシリティーの場合は **-cf** オプションを使用します。両方ではなく、一方を選択する必要があります。このパラメーターを使用して、最後のメンバーおよび最後の CF を DB2 pureScale インスタンスからドロップすることはできません。このパラメーターは、**-add** パラメーターと組み合わせて使用しないでください。

**-instance\_shared\_dev Shared\_Device\_Path\_for\_Instance**

インスタンス共有ファイルとデフォルトのデータベース・パスを保持する DB2 pureScale インスタンスをセットアップするのに必要な共有ディスク装置パスを指定します。例えば、装置パス *dev/hdisk1* を指定します。DB2 pureScale インスタンスについて、共有ディレクトリーは、すべてのホストでアクセス可能でなければなりません。このパラメーターの値は、**-tbdev** パラメーターと同じ値ではありません。このパラメーターと **-instance\_shared\_dir** は、相互に排他的です。

DB2 pureScale インスタンス以外のインスタンスを DB2 pureScale インスタンスに更新する場合にのみ、このパラメーターが必要です。



**-instance\_shared\_mount** *Shared\_Mounting\_Dir*

新規 IBM General Parallel File System (GPFS) ファイル・システム用のマウント・ポイントを指定します。指定するパスは、新規で空のパスでなければならず、既存の GPFS ファイル・システムの内部にネストされたパスであってはなりません。

**-instance\_shared\_dir** *Shared\_Directory\_for\_Instance*

インスタンス共有ファイルとデフォルトのデータベース・パスを保持する DB2 pureScale インスタンスをセットアップするのに必要な、共有ファイル・システム (GPFS) 内のディレクトリーを指定します。例えば、/sharedfs となります。DB2 pureScale インスタンスについて、そのディスクはすべてのホストでアクセス可能でなければなりません。このパラメーターの値は、**-tbdev** パラメーターと同じ値であってはなりません。このパラメーターと **-instance\_shared\_dev** は、相互に排他的です。

DB2 pureScale インスタンス以外のインスタンスを DB2 pureScale インスタンスに更新する場合にのみ、このパラメーターが必要です。

**-tbdev** *Shared\_device\_for\_tiebreaker*

データの保全性を維持するために、DB2 pureScale 環境でタイブレーカーとして機能する共有装置パスを指定します。このパラメーターの値は、**-instance\_shared\_dev** パラメーターや **-instance\_shared\_dir** パラメーターと同じ値であってはなりません。このパラメーターが必要になるのは、DB2 クラスタ・サービスのタイブレーカーが初めて作成されるときや、DB2 pureScale インスタンス以外のインスタンスを DB2 pureScale インスタンスに更新する場合です。DB2 クラスタ・サービスのピア・ドメインが存在する場合は、このパラメーターは無効です。

**-j "TEXT\_SEARCH"**

DB2 Text Search サーバーを、生成されたサービス名と TCP/IP ポート番号のデフォルト値を使用して構成します。インスタンス・タイプがクライアントまたは dsf の場合、このパラメーターは使用できません。

**-j "TEXT\_SEARCH, servicename"**

DB2 Text Search サーバーを、提供されたサービス名と自動的に生成されたポート番号を使用して構成します。サービス名に *services* ファイル内で割り当てられたポート番号がある場合、それはその割り当てられたポート番号を使用します。

**-j "TEXT\_SEARCH, servicename, portnumber"**

DB2 Text Search サーバーを、提供されたサービス名とポート番号を使用して構成します。

**-j "TEXT\_SEARCH, portnumber"**

DB2 Text Search サーバーを、デフォルトのサービス名と提供されたポート番号を使用して構成します。有効なポート番号は 1024 - 65535 の範囲になければなりません。

#### **-u Fenced ID**

**fenced** ユーザー定義関数および **fenced** ストアード・プロシージャを実行するユーザー ID の名前を指定します。このパラメーターが必要になるのは、インスタンスのタイプをクライアント・インスタンスからクライアント以外のインスタンスに変換する場合だけです。現在のインスタンス・タイプを判別するには、**GET DBM CFG** コマンドからの出力のノード・タイプ・パラメーターを参照してください。インスタンスがすでにクライアント以外のインスタンスである場合や、インスタンスがクライアント・インスタンスであり、なおかつクライアント・インスタンスとして保たれる場合 (例えば、**-k** パラメーターを使用する場合) は、**-u** パラメーターは必要ありません。**-u** パラメーターによって、既存のインスタンスの **fenced** ユーザーを変更できます。

#### **-fixtopology**

失敗した追加操作またはドロップ操作を手動で修正する場合に使用します。追加操作の場合、このパラメーターはすべての変更をロールバックして、以前のトポロジーに戻します。ドロップ操作の場合、このパラメーターはドロップ操作を完了させます。このパラメーターは、**-d** を除く他のパラメーターと組み合わせて使用することはできません。

#### *InstName*

インスタンスの名前を指定します。

### **Windows オペレーティング・システムの場合**

#### *InstName*

インスタンスの名前を指定します。

#### **/u:username,password**

DB2 サービスのアカウント名およびパスワードを指定します。

#### **/p:instance-profile-path**

更新されたインスタンス用の新しいインスタンス・プロファイル・パスを指定します。

#### **/r:baseport,endport**

**MPP** モードで実行する場合に、パーティション・データベース・インスタンスによって使用される **TCP/IP** ポートの範囲を指定します。このオプションを指定した場合、ローカル・マシンのサービス・ファイルは、以下の項目で更新されます。

```
DB2_InstName      baseport/tcp
DB2_InstName_END  endport/tcp
```

#### **/h:hostname**

現行のマシンに対して複数のデフォルトの **TCP/IP** ホスト名がある場合、それらをオーバーライドします。

**/s** インスタンスを更新してパーティション・インスタンスにします。

**/q db2iupdt** コマンドを静止モードで発行します。

**/a:authType**

インスタンスの認証タイプ *authType* (SERVER、CLIENT、または SERVER\_ENCRYPT) を指定します。

**/j "TEXT\_SEARCH"**

DB2 Text Search サーバーを、生成されたサービス名と TCP/IP ポート番号のデフォルト値を使用して構成します。インスタンス・タイプがクライアントの場合、このパラメーターは使用できません。

**/j "TEXT\_SEARCH, servicename"**

DB2 Text Search サーバーを、提供されたサービス名と自動的に生成されたポート番号を使用して構成します。サービス名に *services* ファイル内で割り当てられたポート番号がある場合、それはその割り当てられたポート番号を使用します。

**/j "TEXT\_SEARCH, servicename, portnumber"**

DB2 Text Search サーバーを、提供されたサービス名とポート番号を使用して構成します。

**/j "TEXT\_SEARCH, portnumber"**

DB2 Text Search サーバーを、デフォルトのサービス名と提供されたポート番号を使用して構成します。有効なポート番号は 1024 - 65535 の範囲になければなりません。

**/? db2iupdt** コマンドの使用方法を表示します。

## 例

### UNIX および Linux オペレーティング・システムの場合

インスタンス *db2inst2* は、*DB2DIR1* にインストールされている DB2 データベース製品の DB2 コピーに関連しています。同じコンピューター上の *DB2DIR2* に、*DB2DIR1* にインストールされている DB2 データベース製品と同じバージョンの DB2 データベース製品の別のコピーがあります。インスタンスを更新して、*DB2DIR1* にインストールされている DB2 コピーから、*DB2DIR2* にインストールされている DB2 コピーに対して実行するには、以下のコマンドを発行します。

```
DB2DIR2/instance/db2iupdt db2inst2
```

*DB2DIR2* にインストールされている DB2 コピーが *DB2DIR1* でインストールされている DB2 コピーより低いレベルのものであるなら、発行するコマンドは次のようになります。

```
DB2DIR2/instance/db2iupdt -D db2inst2
```

### リリース内のさらに高いレベルに、インスタンスを更新する

DB2 インスタンスをさらに高いレベルに更新する場合や、1 つの DB2 インストール・パスから別のインストール・パスに更新する場合には、次のようなコマンドを入力します。

```
DB2DIR/instance/db2iupdt db2inst1
```

ここで、*DB2DIR* は、DB2 のコピーのインストール・ロケーションを示します。このコマンドが DB2 pureScale Feature コピーから実行される場合、既存の *db2inst1* のインスタンス・タイプは *dsf* でなければなりません。*db2inst1* インスタンスが DB2 pureScale インスタンスである場合、この例により、それを 1 つのレベルから、DB2 pureScale Feature を持つ DB2 ESE の異なるレベルに更新することができます。この例は、*ese* タイプのインスタンスを DB2 pureScale インスタンスに更新することには適用されません。次の例は、この手順を略述しています。

### DB2 pureScale インスタンス以外のインスタンスを DB2 pureScale インスタンスに更新する

あるインスタンスを DB2 pureScale インスタンスに更新するには、以下のようになります。

```
DB2DIR/instance/db2iupdt
-cf host2
-cfnet host2-ib0
-m host1
-mnet host1-ib0
-instance_shared_dev /dev/hdisk1
-tbdev /dev/hdisk2
-u db2fenc1
db2inst1
```

*DB2DIR* は、DB2 コピーのインストール場所を表します。

さらにこのコマンドは */dev/hdisk1* を使用して、インスタンス共有ファイルを保管するための共有ファイル・システムを作成し、タイプレーカーとして機能する共有装置パスとして */dev/hdisk2* をセットアップします。

**-tbdev** パラメーターの値は、**-instance\_shared\_dev** パラメーターの値とは異なってなければなりません。

### DB2 pureScale インスタンスの拡張と縮小 (*db2iupdt -add* または *db2iupdt -drop* を使用)

以下の例は DB2 pureScale 環境に適用されます。

- DB2 pureScale インスタンスを更新して、メンバーを追加します。 *host1* というメンバーを、ネット名 *host1-ib0* で、*db2sdin1* という DB2 pureScale インスタンスに追加するには、次のようなコマンドを入力します。

```
DB2DIR/instance/db2iupdt -d -add -m host1 -mnet host1-ib0 db2sdin1
```

*DB2DIR* は、DB2 コピーのインストール場所を表します。

- DB2 pureScale インスタンスを更新して、第 2 のクラスター・キャッシング・ファシリティーを追加します。 *host2* というクラスター・キャッシング・ファシリティーを、ネット名 *host2-ib0* で、*db2sdin1* という DB2 pureScale インスタンスに追加するには、次のようなコマンドを入力します。

```
DB2DIR/instance/db2iupdt -d -add -cf host2 -cfnet host2-ib0 db2sdin1
```

*DB2DIR* は、DB2 コピーのインストール場所を表します。

- DB2 pureScale インスタンスから、メンバーをドロップします。 host1 というメンバーを、db2sdin1 という DB2 pureScale インスタンスからドロップするには、次のようなコマンドを入力します。

```
DB2DIR/instance/db2iupdt -d -drop -m host1 db2sdin1
```

ここで、*DB2DIR* は、DB2 のコピーのインストール・ロケーションを示します。 host1 が同じインスタンス内で CF ロールを持っていない場合には、host1 以外のホストからコマンドを実行する必要があります。

## 使用上の注意

### サポートされているすべてのオペレーティング・システムの場合

- **db2iupdt** コマンドを使用して、DB2 データベース製品の同じバージョンの、ある DB2 コピーから別の DB2 コピーに DB2 インスタンスを更新する場合、旧 DB2 コピーのインストール・パス内で定義されている DB2 グローバル・プロファイル変数は、新しいインストール・ロケーションに更新されません。インスタンス固有の DB2 インスタンス・プロファイル変数は、インスタンスの更新後に実行されます。

### UNIX および Linux オペレーティング・システムの場合

- DB2DB2 Enterprise Server Edition のみ、**db2iupdt** コマンドを使用して更新できます。
- メンバーの追加やドロップなど、メンバー・トポロジーを変更する場合、データベースにアクセスする前にオフライン・バックアップをとる必要があります。オフライン・バックアップをとる前にデータベースにアクセスしようとする、データベースはバックアップ・ペンディング状態に入ります。

複数のメンバーを追加したり、複数のメンバーをドロップすることができ、変更ごとに後でバックアップをとらなければならないわけではありません。例えば、メンバーを 3 つ追加する場合、バックアップをとる必要があるのは、すべての追加操作が完了した後だけです。ただし、2 つのメンバーを追加した後に 1 つのメンバーをドロップしたり、2 つのメンバーをドロップした後に 1 つのメンバーを追加したりする場合には、何らかの追加のトポロジー変更を実行する前に、バックアップをとる必要があります。

- **db2iupdt** コマンドは *DB2DIR/instance* ディレクトリーにあります。*DB2DIR* は、DB2 データベース製品の現行バージョンがインストールされている場所です。
- 非ルート・インスタンスを更新しない場合は、**db2nrupdt** 非 root インストール・インスタンス更新コマンドを参照してください。**db2iupdt** は、非ルート・インスタンスの更新をサポートしません。
- root ユーザーになるために **login** コマンドではなく **su** コマンドを使用している場合は、**su** コマンドを - オプションを指定して発行して、あたかも **login** コマンドを使用してシステムにログインしているかのように、プロセス環境を設定することを指示する必要があります。
- root ユーザー用の DB2 インスタンス環境を入手しないでください。DB2 インスタンス環境を入手している場合に、**db2iupdt** を実行することはサポートされていません。

- AIX 6.1 (またはそれ以上) では、このコマンドは、システム・ワークロード・パーティション (WPAR) グローバル環境内の共有 DB2 コピーから実行する場合、root ユーザーとして実行されなければなりません。WPAR は DB2 pureScale 環境ではサポートされていません。
- **db2iupdt** を実行して、リリース内のさらに高いレベルへインスタンスを更新する場合には、ルーチンとライブラリーが各メンバーから共有ロケーションにコピーされます。名前が同じものの、各ホストで内容が異なるライブラリーがある場合には、共有ロケーション内のライブラリー・コンテンツは、**db2iupdt** コマンドを最後に実行したホストのライブラリー・コンテンツになります。

#### Windows オペレーティング・システムの場合

- **db2iupdt** コマンドは **DB2PATH\bin** ディレクトリーにあります。**DB2PATH** は、DB2 データベース製品の現行バージョンがインストールされている場所です。
- インスタンスは、**db2iupdt** コマンドの発行元から DB2 コピーに更新されます。ただし、インスタンス・プロファイルを現在の場所から別の場所に移動するには、**/p** オプションを使用し、インスタンス・プロファイル・パスを指定します。こうしない場合、インスタンスの更新後にインスタンス・プロファイルが元の場所に残ります。代わりに **db2iupgrade** コマンドを使用して、前のリリースから現行リリースにアップグレードします。

---

## 付録 D. DB2 技術情報の概説

DB2 技術情報は、さまざまな方法でアクセスすることが可能な、各種形式で入手できます。

DB2 技術情報は、以下のツールと方法を介して利用できます。

- DB2インフォメーション・センター
  - トピック (タスク、概念、およびリファレンス・トピック)
  - サンプル・プログラム
  - チュートリアル
- DB2 資料
  - PDF ファイル (ダウンロード可能)
  - PDF ファイル (DB2 PDF DVD に含まれる)
  - 印刷資料
- コマンド行ヘルプ
  - コマンド・ヘルプ
  - メッセージ・ヘルプ

**注:** DB2 インフォメーション・センターのトピックは、PDF やハードコピー資料よりも頻繁に更新されます。最新の情報を入手するには、資料の更新が発行されたときにそれをインストールするか、[ibm.com](http://ibm.com) にある DB2 インフォメーション・センターを参照してください。

技術資料、ホワイト・ペーパー、IBM Redbooks® 資料などのその他の DB2 技術情報には、オンライン ([ibm.com](http://ibm.com)) でアクセスできます。DB2 Information Management ソフトウェア・ライブラリー・サイト (<http://www.ibm.com/software/data/sw-library/>) にアクセスしてください。

### 資料についてのフィードバック

DB2 の資料についてのお客様からの貴重なご意見をお待ちしています。DB2 の資料を改善するための提案については、[db2docs@ca.ibm.com](mailto:db2docs@ca.ibm.com) まで E メールを送信してください。DB2 の資料チームは、お客様からのフィードバックすべてに目を通しますが、直接お客様に返答することはありません。お客様が関心をお持ちの内容について、可能な限り具体的な例を提供してください。特定のトピックまたはヘルプ・ファイルについてのフィードバックを提供する場合は、そのトピック・タイトルおよび URL を含めてください。

DB2 お客様サポートに連絡する場合には、この E メール・アドレスを使用しないでください。資料を参照しても、DB2 の技術的な問題が解決しない場合は、お近くの IBM サービス・センターにお問い合わせください。

## DB2 テクニカル・ライブラリー (ハードコピーまたは PDF 形式)

以下の表は、IBM Publications Center ([www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss](http://www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss)) から利用できる DB2 ライブラリーについて説明しています。英語および翻訳された DB2 バージョン 10.1 のマニュアル (PDF 形式) は、[www.ibm.com/support/docview.wss?rs=71&uid=swg2700947](http://www.ibm.com/support/docview.wss?rs=71&uid=swg2700947) からダウンロードできます。

この表には印刷資料が入手可能かどうかを示されていますが、国または地域によっては入手できない場合があります。

資料番号は、資料が更新される度に大きくなります。資料を参照する際は、以下にリストされている最新版であることを確認してください。

注: DB2 インフォメーション・センターは、PDF やハードコピー資料よりも頻繁に更新されます。

表 26. DB2 の技術情報

資料名	資料番号	印刷資料が入手可能かどうか	最終更新
管理 API リファレンス	SA88-4671-00	入手可能	2012 年 4 月
管理ルーチンおよびビュー	SA88-4672-00	入手不可	2012 年 4 月
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	SA88-4676-00	入手可能	2012 年 4 月
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	SA88-4677-00	入手可能	2012 年 4 月
コマンド・リファレン ス	SA88-4673-00	入手可能	2012 年 4 月
データベース: 管理の 概念および構成リファ レンス	SA88-4662-00	入手可能	2012 年 4 月
データ移動キューティ リティー ガイドおよびリ ファレンス	SA88-4693-00	入手可能	2012 年 4 月
データベースのモニタ リング ガイドおよびリ ファレンス	SA88-4663-00	入手可能	2012 年 4 月
データ・リカバリーと 高可用性 ガイドおよび リファレンス	SA88-4694-00	入手可能	2012 年 4 月
データベース・セキュ リティー・ガイド	SA88-4695-00	入手可能	2012 年 4 月



表 26. DB2 の技術情報 (続き)

資料名	資料番号	印刷資料が入手可能かどうか	最終更新
DB2 ワークロード管理ガイドおよびリファレンス	SA88-4685-00	入手可能	2012 年 4 月
ADO.NET および OLE DB アプリケーションの開発	SA88-4665-00	入手可能	2012 年 4 月
組み込み SQL アプリケーションの開発	SA88-4666-00	入手可能	2012 年 4 月
Java アプリケーションの開発	SA88-4669-00	入手可能	2012 年 4 月
Perl、PHP、Python および Ruby on Rails アプリケーションの開発	SA88-4670-00	入手不可	2012 年 4 月
SQL および外部ルーチンの開発	SA88-4667-00	入手可能	2012 年 4 月
データベース・アプリケーション開発の基礎	GI88-4279-00	入手可能	2012 年 4 月
DB2 インストールおよび管理 概説 (Linux および Windows 版)	GI88-4280-00	入手可能	2012 年 4 月
グローバリゼーション・ガイド	SA88-4696-00	入手可能	2012 年 4 月
DB2 サーバー機能 インストール	GA88-4679-00	入手可能	2012 年 4 月
IBM データ・サーバー・クライアント機能インストール	GA88-4680-00	入手不可	2012 年 4 月
メッセージ・リファレンス 第 1 巻	SA88-4688-00	入手不可	2012 年 4 月
メッセージ・リファレンス 第 2 巻	SA88-4689-00	入手不可	2012 年 4 月
Net Search Extender 管理およびユーザズ・ガイド	SA88-4691-00	入手不可	2012 年 4 月
パーティションおよびクラスタリングのガイド	SA88-4697-00	入手可能	2012 年 4 月
pureXML ガイド	SA88-4686-00	入手可能	2012 年 4 月
Spatial Extender ユーザズ・ガイドおよびリファレンス	SA88-4690-00	入手不可	2012 年 4 月

表 26. DB2 の技術情報 (続き)

資料名	資料番号	印刷資料が入手可能 かどうか	最終更新
SQL プロシージャ言語: アプリケーション のイネーブルメントお よびサポート	SA88-4668-00	入手可能	2012 年 4 月
SQL リファレンス 第 1 巻	SA88-4674-00	入手可能	2012 年 4 月
SQL リファレンス 第 2 巻	SA88-4675-00	入手可能	2012 年 4 月
Text Search ガイド	SA88-4692-00	入手可能	2012 年 4 月
問題判別およびデータ ベース・パフォーマンス のチューニング	SA88-4664-00	入手可能	2012 年 4 月
DB2 バージョン 10.1 へのアップグレード	SA88-4678-00	入手可能	2012 年 4 月
DB2 バージョン 10.1 の新機能	SA88-4684-00	入手可能	2012 年 4 月
XQuery リファレンス	SA88-4687-00	入手不可	2012 年 4 月

表 27. DB2 Connect 固有の技術情報

資料名	資料番号	印刷資料が入手可能 かどうか	最終更新
DB2 Connect Personal Edition インストールお よび構成	SA88-4681-00	入手可能	2012 年 4 月
DB2 Connect サーバー 機能 インストールおよ び構成	SA88-4682-00	入手可能	2012 年 4 月
DB2 Connect ユーザー ズ・ガイド	SA88-4683-00	入手可能	2012 年 4 月

## コマンド行プロセッサから SQL 状態ヘルプを表示する

DB2 製品は、SQL ステートメントの結果の原因になったと考えられる条件の SQLSTATE 値を戻します。SQLSTATE ヘルプは、SQL 状態および SQL 状態クラス・コードの意味を説明します。

### 手順

SQL 状態ヘルプを開始するには、コマンド行プロセッサを開いて以下のように入力します。

```
? sqlstate または ? class code
```

ここで、*sqlstate* は有効な 5 桁の SQL 状態を、*class code* は SQL 状態の最初の 2 桁を表します。

例えば、? 08003 を指定すると SQL 状態 08003 のヘルプが表示され、? 08 を指定するとクラス・コード 08 のヘルプが表示されます。

---

## 異なるバージョンの DB2 インフォメーション・センターへのアクセス

他のバージョンの DB2 製品の資料は、ibm.com® のそれぞれのインフォメーション・センターにあります。

### このタスクについて

DB2 バージョン 10.1 のトピックを扱っている DB2 インフォメーション・センターの URL は、<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1> です。

DB2 バージョン 9.8 のトピックを扱っている DB2 インフォメーション・センターの URL は、<http://publib.boulder.ibm.com/infocenter/db2luw/v9r8/> です。

DB2 バージョン 9.7 のトピックを扱っている DB2 インフォメーション・センターの URL は、<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/> です。

DB2 バージョン 9.5 のトピックを扱っている DB2 インフォメーション・センターの URL は、<http://publib.boulder.ibm.com/infocenter/db2luw/v9r5> です。

DB2 バージョン 9.1 のトピックを扱っている DB2 インフォメーション・センターの URL は、<http://publib.boulder.ibm.com/infocenter/db2luw/v9/> です。

DB2 バージョン 8 のトピックについては、DB2 インフォメーション・センターの URL (<http://publib.boulder.ibm.com/infocenter/db2luw/v8/>) を参照してください。

---

## コンピューターまたはイントラネット・サーバーにインストールされた DB2 インフォメーション・センターの更新

ローカルにインストールした DB2 インフォメーション・センターは、定期的に更新する必要があります。

### 始める前に

DB2 バージョン 10.1 インフォメーション・センターが既にインストール済みである必要があります。詳しくは、「DB2 サーバー機能 インストール」の『DB2 セットアップ・ウィザードによる DB2 インフォメーション・センターのインストール』のトピックを参照してください。インフォメーション・センターのインストールに適用されるすべての前提条件と制約事項は、インフォメーション・センターの更新にも適用されます。

### このタスクについて

既存の DB2 インフォメーション・センターは、自動で更新することも手動で更新することもできます。

- 自動更新は、既存のインフォメーション・センターのフィーチャーと言語を更新します。自動更新を使用すると、手動更新と比べて、更新中にインフォメーション

ン・センターが使用できなくなる時間が短くなるというメリットがあります。さらに、自動更新は、定期的に行う他のバッチ・ジョブの一部として実行されるように設定することができます。

- 手動更新は、既存のインフォメーション・センターのフィーチャーと言語の更新に使用できます。自動更新は更新処理中のダウン時間を減らすことができますが、フィーチャーまたは言語を追加する場合は手動処理を使用する必要があります。例えば、ローカルのインフォメーション・センターが最初は英語とフランス語でインストールされており、その後ドイツ語もインストールすることにした場合、手動更新でドイツ語をインストールし、同時に、既存のインフォメーション・センターのフィーチャーおよび言語を更新できます。しかし、手動更新ではインフォメーション・センターを手動で停止、更新、再始動する必要があります。更新処理の間はずっと、インフォメーション・センターは使用できなくなります。自動更新処理では、インフォメーション・センターは、更新を行った後に、インフォメーション・センターを再始動するための停止が発生するだけで済みます。

このトピックでは、自動更新のプロセスを詳しく説明しています。手動更新の手順については、『コンピューターまたはイントラネット・サーバーにインストールされた DB2 インフォメーション・センターの手動更新』のトピックを参照してください。

## 手順

コンピューターまたはイントラネット・サーバーにインストールされている DB2 インフォメーション・センターを自動更新する手順を以下に示します。

1. Linux オペレーティング・システムの場合、次のようにします。
  - a. インフォメーション・センターがインストールされているパスにナビゲートします。デフォルトでは、DB2 インフォメーション・センターは、`/opt/ibm/db2ic/V10.1` ディレクトリーにインストールされています。
  - b. インストール・ディレクトリーから `doc/bin` ディレクトリーにナビゲートします。
  - c. 次のように `update-ic` スクリプトを実行します。

```
update-ic
```
2. Windows オペレーティング・システムの場合、次のようにします。
  - a. コマンド・ウィンドウを開きます。
  - b. インフォメーション・センターがインストールされているパスにナビゲートします。デフォルトでは、DB2 インフォメーション・センターは、`<Program Files>¥IBM¥DB2 Information Center¥バージョン 10.1` ディレクトリーにインストールされています (`<Program Files>` は「Program Files」ディレクトリーのロケーション)。
  - c. インストール・ディレクトリーから `doc¥bin` ディレクトリーにナビゲートします。
  - d. 次のように `update-ic.bat` ファイルを実行します。

```
update-ic.bat
```

## タスクの結果

DB2 インフォメーション・センターが自動的に再始動します。更新が入手可能な場合、インフォメーション・センターに、更新された新しいトピックが表示されます。インフォメーション・センターの更新が入手可能でなかった場合、メッセージがログに追加されます。ログ・ファイルは、`doc\%eclipse%configuration` ディレクトリにあります。ログ・ファイル名はランダムに生成された名前です。例えば、`1239053440785.log` のようになります。

---

## コンピューターまたはイントラネット・サーバーにインストールされた DB2 インフォメーション・センターの手動更新

DB2 インフォメーション・センターをローカルにインストールしている場合は、IBM から資料の更新を入手してインストールすることができます。

### このタスクについて

ローカルにインストールされた *DB2* インフォメーション・センター を手動で更新するには、以下のことを行う必要があります。

1. コンピューター上の *DB2* インフォメーション・センター を停止し、インフォメーション・センターをスタンドアロン・モードで再始動します。インフォメーション・センターをスタンドアロン・モードで実行すると、ネットワーク上の他のユーザーがそのインフォメーション・センターにアクセスできなくなります。これで、更新を適用できるようになります。*DB2* インフォメーション・センターのワークステーション・バージョンは、常にスタンドアロン・モードで実行されます。を参照してください。
2. 「更新」機能を使用することにより、どんな更新が利用できるかを確認します。インストールしなければならない更新がある場合は、「更新」機能を使用してそれを入手およびインストールできます。

**注:** ご使用の環境において、インターネットに接続されていないマシンに *DB2* インフォメーション・センター の更新をインストールする必要がある場合、インターネットに接続されていて *DB2* インフォメーション・センター がインストールされているマシンを使用して、更新サイトをローカル・ファイル・システムにミラーリングしてください。ネットワーク上の多数のユーザーが資料の更新をインストールする場合にも、更新サイトをローカルにミラーリングして、更新サイト用のプロキシを作成することにより、個々のユーザーが更新を実行するのに要する時間を短縮できます。

更新パッケージが入手可能な場合、「更新」機能を使用してパッケージを入手します。ただし、「更新」機能は、スタンドアロン・モードでのみ使用できます。

3. スタンドアロンのインフォメーション・センターを停止し、コンピューター上の *DB2* インフォメーション・センター を再開します。

**注:** Windows 2008、Windows Vista (およびそれ以上) では、このセクションの後の部分でリストされているコマンドは管理者として実行する必要があります。完全な管理者特権でコマンド・プロンプトまたはグラフィカル・ツールを開くには、ショートカットを右クリックしてから、「管理者として実行」を選択します。

## 手順

コンピューターまたはイントラネット・サーバーにインストール済みの DB2 インフォメーション・センターを更新するには、以下のようにします。

1. DB2 インフォメーション・センターを停止します。
    - Windows では、「スタート」 > 「コントロール パネル」 > 「管理ツール」 > 「サービス」をクリックします。次に、「DB2 インフォメーション・センター」サービスを右クリックして「停止」を選択します。
    - Linux では、以下のコマンドを入力します。

```
/etc/init.d/db2icdv10 stop
```
  2. インフォメーション・センターをスタンドアロン・モードで開始します。
    - Windows の場合:
      - a. コマンド・ウィンドウを開きます。
      - b. インフォメーション・センターがインストールされているパスにナビゲートします。デフォルトでは、DB2 インフォメーション・センターは、`Program_Files\IBM\DB2 Information Center\バージョン 10.1` ディレクトリーにインストールされています (`Program_Files` は Program Files ディレクトリーのロケーション)。
      - c. インストール・ディレクトリーから `doc\bin` ディレクトリーにナビゲートします。
      - d. 次のように `help_start.bat` ファイルを実行します。

```
help_start.bat
```
    - Linux の場合:
      - a. インフォメーション・センターがインストールされているパスにナビゲートします。デフォルトでは、DB2 インフォメーション・センターは、`/opt/ibm/db2ic/V10.1` ディレクトリーにインストールされています。
      - b. インストール・ディレクトリーから `doc/bin` ディレクトリーにナビゲートします。
      - c. 次のように `help_start` スクリプトを実行します。

```
help_start
```
- システムのデフォルト Web ブラウザーが開き、スタンドアロンのインフォメーション・センターが表示されます。
3. 「更新」ボタン (🔄) をクリックします。(ブラウザーで JavaScript が有効になっている必要があります。) インフォメーション・センターの右側のパネルで、「更新の検索」をクリックします。既存の文書に対する更新のリストが表示されます。
  4. インストール・プロセスを開始するには、インストールする更新をチェックして選択し、「更新のインストール」をクリックします。
  5. インストール・プロセスが完了したら、「完了」をクリックします。
  6. 次のようにして、スタンドアロンのインフォメーション・センターを停止します。
    - Windows の場合は、インストール・ディレクトリーの `doc\bin` ディレクトリーにナビゲートしてから、次のように `help_end.bat` ファイルを実行します。

help\_end.bat

注: help\_end バッチ・ファイルには、help\_start バッチ・ファイルを使用して開始したプロセスを安全に停止するのに必要なコマンドが含まれています。help\_start.bat は、Ctrl-C や他の方法を使用して停止しないでください。

- Linux の場合は、インストール・ディレクトリーの doc/bin ディレクトリーにナビゲートしてから、次のように help\_end スクリプトを実行します。

help\_end

注: help\_end スクリプトには、help\_start スクリプトを使用して開始したプロセスを安全に停止するのに必要なコマンドが含まれています。他の方法を使用して、help\_start スクリプトを停止しないでください。

#### 7. DB2 インフォメーション・センター を再開します。

- Windows では、「スタート」 > 「コントロール パネル」 > 「管理ツール」 > 「サービス」をクリックします。次に、「DB2 インフォメーション・センター」サービスを右クリックして「開始」を選択します。
- Linux では、以下のコマンドを入力します。

```
/etc/init.d/db2icdv10 start
```

## タスクの結果

更新された DB2 インフォメーション・センター に、更新された新しいトピックが表示されます。

---

## DB2 チュートリアル

DB2 チュートリアルは、DB2 データベース製品のさまざまな機能について学習するための支援となります。この演習をとおして段階的に学習することができます。

### はじめに

インフォメーション・センター (<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/>) から、このチュートリアルの XHTML 版を表示できます。

演習の中で、サンプル・データまたはサンプル・コードを使用する場合があります。個々のタスクの前提条件については、チュートリアルを参照してください。

### DB2 チュートリアル

チュートリアルを表示するには、タイトルをクリックします。

「*pureXML* ガイド」の『**pureXML**』

XML データを保管し、ネイティブ XML データ・ストアに対して基本的な操作を実行できるように、DB2 データベースをセットアップします。

---

## DB2 トラブルシューティング情報

DB2 データベース製品を使用する際に役立つ、トラブルシューティングおよび問題判別に関する広範囲な情報を利用できます。

## DB2 の資料

トラブルシューティング情報は、「問題判別およびデータベース・パフォーマンスのチューニング」または *DB2* インフォメーション・センターの『データベースの基本』セクションにあります。ここでは、以下の情報が記載されています。

- *DB2* 診断ツールおよびユーティリティーを使用した、問題の切り分け方法および識別方法に関する情報。
- 最も一般的な問題のうち、いくつかの解決方法。
- *DB2* データベース製品で発生する可能性のある、その他の問題の解決に役立つアドバイス。

## IBM サポート・ポータル

現在問題が発生していて、考えられる原因とソリューションを見つけるには、IBM サポート・ポータルを参照してください。Technical Support サイトには、最新の *DB2* 資料、TechNotes、プログラム診断依頼書 (APAR またはバグ修正)、フィックスパック、およびその他のリソースへのリンクが用意されています。この知識ベースを活用して、問題に対する有効なソリューションを探し出すことができます。

IBM サポート・ポータル ([http://www.ibm.com/support/entry/portal/Overview/Software/Information\\_Management/DB2\\_for\\_Linux,\\_UNIX\\_and\\_Windows](http://www.ibm.com/support/entry/portal/Overview/Software/Information_Management/DB2_for_Linux,_UNIX_and_Windows)) にアクセスしてください。

---

## ご利用条件

これらの資料は、以下の条件に同意していただける場合に限りご使用いただけます。

**適用度:** これらのご利用条件は、IBM Web サイトのあらゆるご利用条件に追加で適用されるものです。

**個人使用:** これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布 (頒布、送信を含む) または表示 (上映を含む) することはできません。

**商業的使用:** これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

**権利:** ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。



お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、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なしで提供されます。

**IBM の商標:** IBM、IBM ロゴおよび `ibm.com` は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。



---

## 付録 E. 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。IBM 以外の製品に関する情報は、本書の最初の発行時点で入手可能な情報に基づいており、変更される場合があります。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒103-8510  
東京都中央区日本橋箱崎町19番21号  
日本アイ・ビー・エム株式会社  
法務・知的財産  
知的財産権ライセンス渉外

**以下の保証は、国または地域の法律に沿わない場合は、適用されません。** IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Canada Limited  
U59/3600  
3600 Steeles Avenue East  
Markham, Ontario L3R 9Z7  
CANADA

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができませんが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性がありますが、その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確証できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、

利便性もしくは機能性があることをほのめかしたり、保証することはできません。サンプル・プログラムは、現存するままの状態を提供されるものであり、いかなる種類の保証も提供されません。IBM は、これらのサンプル・プログラムの使用から生ずるいかなる損害に対しても責任を負いません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生した創作物には、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. \_年を入れる\_. All rights reserved.

## 商標

IBM、IBM ロゴおよび [ibm.com](http://ibm.com) は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。

以下は、それぞれ各社の商標または登録商標です。

- Linux は、Linus Torvalds 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 Oracle やその関連会社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 の米国およびその他の国における登録商標です。
- インテル、Intel、Intel ロゴ、Intel Inside、Intel Inside ロゴ、Celeron、Intel SpeedStep、Itanium、Pentium は、Intel Corporation または子会社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。
- 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。



# 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]

- インスタンスのアップグレード・コマンド 237
- インスタンスの更新コマンド 253
- インスタンスの作成コマンド 240
- インスタンスの除去コマンド 250
- インストール
  - サイレント
    - Linux 60
    - UNIX 60
    - Windows 60
- 応答ファイル
  - インストール
    - Linux 60
    - UNIX 60
    - Windows 60

## [カ行]

- カタログ
  - TCP/IP ノード 72
- 関数
  - ストアド・プロシージャ
    - SYSTS\_ALTER 181
    - SYSTS\_CLEANUP 187
    - SYSTS\_CLEAR\_COMMANDLOCKS 189
    - SYSTS\_CLEAR\_EVENTS 191
    - SYSTS\_DROP 207
    - SYSTS\_ENABLE 210
    - SYSTS\_UPDATE 212
    - SYSTS\_UPGRADE\_CATALOG 215
    - SYSTS\_UPGRADE\_INDEX 217
- 更新
  - DB2 インフォメーション・センター 267, 269
- コマンド
  - プロシージャからの呼び出し 179
  - db2icrt 240
  - db2idrop 250
  - db2iupdt 253
  - db2iupgrade
    - 詳細情報 237
  - db2ts ALTER INDEX 146
  - db2ts CLEANUP FOR TEXT 151
  - db2ts CLEAR COMMAND LOCKS 152
  - db2ts CLEAR EVENTS FOR INDEX 153
  - db2ts CREATE INDEX 155

- コマンド (続き)
  - db2ts DISABLE DATABASE FOR TEXT 163
  - db2ts DROP INDEX 164
  - db2ts ENABLE DATABASE FOR TEXT 166
  - db2ts HELP 168
  - db2ts RESET PENDING 170
  - db2ts START FOR TEXT 171
  - db2ts STOP FOR TEXT 173
  - db2ts UPDATE INDEX 174
- ご利用条件
  - 資料 272

## [サ行]

- サービス・ファイル
  - TCP/IP 通信のための更新 72
- サイレント・インストール
  - Linux 60
  - UNIX 60
  - Windows 60
- 照会用語に隣接する特殊文字 121
- 資料
  - 印刷 264
  - 概要 263
  - 使用に関するご利用条件 272
  - PDF ファイル 264
- スタンドアロン・インストール 62
- ソフトウェア要件
  - DB2 Text Search 48

## [タ行]

- チュートリアル
  - トラブルシューティング 272
  - 問題判別 272
  - リスト 271
  - pureXML 271
- チューニング 28
- テキスト検索 86
  - 検索関数 115
  - CONTAINS 関数 115
  - SCORE 関数 115
  - xmlcolumn-contains 関数 115
- テキスト索引
  - 近接検索 118
- 同義語ディクショナリー
  - 概要 93
  - 除去 94
  - 追加 93
- 特殊文字の検索 119
- 特記事項 275

トラブルシューティング  
オンライン情報 272  
チュートリアル 272

## [ハ行]

ビュー  
DB2 Text Search  
索引レベルの情報 224  
SYSIBMTS.TSINDEXES 224  
プロシージャ  
SYSTS\_ADMIN\_CMD 179  
SYSTS\_CREATE 196  
SYSTS\_START 193  
ヘルプ  
SQL ステートメント 266

## [マ行]

問題判別  
チュートリアル 272  
利用できる情報 272

## A

ALTER INDEX Text Search コマンド 146

## C

CLEANUP FOR TEXT Text Search コマンド 151  
CLEAR COMMAND LOCKS Text Search コマンド 152  
CLEAR EVENTS FOR INDEX Text Search コマンド 153  
CREATE INDEX Text Search コマンド 155

## D

DB2 Text Search 63  
アップグレード 77  
アンインストール 63  
イベント表  
概要 94  
メッセージの削除 108  
インクリメンタル索引更新 106  
インスタンス・サービスの停止 88  
インストール  
応答ファイル 50  
概要 45  
セットアップ・ウィザード 50  
DB2 Accessories Suite のフィルター・ライブラリー 73  
db2\_install コマンド 51  
インストールのディスク・スペース所要量 61  
開始 88  
概要 1, 21, 229  
関数 115

DB2 Text Search (続き)

管理 102, 145  
管理コマンド 102, 145  
要約 102, 145  
管理ビュー  
イベント表 226  
索引レベル 221, 224, 226, 228  
ステージング表 228  
データベース・レベル 221, 222  
ログ表 228  
SYSIBMTS.TSCOLLECTIONNAMES 226  
SYSIBMTS.TSCONFIGURATION 226  
SYSIBMTS.TSDEFAULTS 221  
SYSIBMTS.TSEVENT 226  
SYSIBMTS.TSINDEXES 224  
SYSIBMTS.TSLOCKS 222  
SYSIBMTS.TSSERVERS 223  
SYSIBMTS.TSSTAGING 228  
管理ルーチン 104, 179  
キー・フィーチャーと概念 3  
基本検索 117  
キャパシティー・プランニング  
TCP/IP ポートの要件 37  
キャパシティー・プランニングおよび最適化 27  
キュー・メモリー・サイズ 31  
近接検索 118  
権限  
インスタンス所有者 25  
データベース管理者 25  
テキスト検索照会を実行するユーザー 26  
ロール 25  
言語学的処理 14  
N-gram セグメンテーションの使用 14  
検索パフォーマンスの改善 134  
検索回数  
構文 125  
パフォーマンスへの影響 39  
更新特性の変更 108  
更新に関する索引固有のパラメーター 36  
更新のインストール 83  
構成  
応答ファイル 50  
概要 45  
構成ツール 67  
セットアップ・ウィザード 50  
方法 65  
構成ツール 67  
構成のチューニング 28  
コマンド  
ALTER INDEX 146  
CLEANUP FOR TEXT 151  
CLEAR COMMAND LOCKS 152  
CLEAR EVENTS FOR INDEX 153  
CREATE INDEX 155  
DISABLE DATABASE FOR TEXT 163  
DROP INDEX 164



## DB2 Text Search (続き)

### コマンド (続き)

- ENABLE DATABASE FOR TEXT 166
- HELP 168
- RESET PENDING 170
- START FOR TEXT 171
- STOP FOR TEXT 173
- UPDATE INDEX 174

### コマンド行ツール 85

### コマンドの実行 85

### コレクションのロケーションの変更 109

### サーバー構成 28

### サーバー情報の更新 69

### 再構成 65, 67

### 最大ヒープ・サイズ 29

### 索引

#### インクリメンタル更新 13

#### 検索 116

#### 更新 7

#### 作成 7, 94, 95

#### 作成 (サポートされていないデータ・タイプ) 97

#### 作成 (バイナリー・データ・タイプ) 97

#### バイナリー・データ・タイプ 97

#### 変更 7

#### 保守 102

### 索引状況の表示 109

### 索引ソース 32

### 索引ソースの特性 32

### 索引付けスレッド 30

### 索引の計画および最適化 32

### 索引の検索 115

### 索引の最適化 33

### 索引のドロップ 111

### 索引の変更 108

### 索引のロケーション 35

### 索引マネージャー 25

### サポートされていないデータ・タイプ 97

### サポートされる言語 22

### サポートされるコード・ページ 22

### システムのチューニング 37

### シナリオ 16

### 照会の計画 39

### 照会用語に隣接する特殊文字 121

### スタンドアロン・インストール 51

### スタンドアロン・サーバー 5

#### 構成 70

### スタンドアロン・サーバーのインストールおよび構成 62

### スタンドアロン・サーバーの構成 70

### セキュリティの概要 23, 26

### タスクのスケジューリング 112

### データベースでテキストを使用可能にする 89

### データベースでテキストを使用不可にする 90

### データベースの使用可能化 89

### データベースを使用不可にする 90

### データ・タイプ

#### サポートされている 21

## DB2 Text Search (続き)

### データ・タイプ (続き)

#### サポートされないものの変換 21

### ディスク使用量 34

### テキスト検索コレクション

#### 孤立したものを削除 91

#### 孤立したものを特定 91

### テキスト検索索引イベントの消去 108

### テキスト検索索引固有のパラメーター 36

### テキスト検索索引のドロップ 111

### テキスト検索索引の変更 108

### テキスト索引の更新 104

### テキスト索引の最適化 33

### 同義語ディクショナリー

#### 概要 93

### 同義語ディクショナリーの除去 94

### 同義語ディクショナリーの追加 93

### 統合サーバー 5

### 特殊文字のエスケープ 119

### 特殊文字の検索 119

### 特殊文字の索引付け 121

### トリガー 33, 94

### パーサー構成 42

### パーティション・データベース環境 10

### ハードウェアおよびソフトウェア 48

### ハードウェア要件 48

### バックアップ 110

### パフォーマンス 39

### 非 root のアップグレード 81

### ヒープ・メモリー使用量 29

### 非同期の索引付け 33

### ファイル記述子 39

### ファジー検索 117

### フィルター・ライブラリー 73

### 複数の述部 40

### 複数パーティション・インスタンスのアップグレード 83

### プランニングおよび最適化 32

### プレーン・テキストでの形態素方式の索引付け 98, 100

### 文書形式

#### サポートされている 21

#### サポートされないものの変換 21

### 文書の切り捨て 23

### ユーザー・ロール 25

### リストア 110

### リッチ・テキスト 86

#### 概要 19

### リッチ・テキスト文書のサポート 19, 73

### リッチ・テキスト・サポートの無効化 87

### リッチ・テキスト・サポートの有効化 86

### ロール

#### インスタンス所有者 25

#### 検索を実行するユーザー 26

#### データベース管理者 25

### ログ表 94

### ロケールおよび言語 40

### Accessories Suite のインストール 73

## DB2 Text Search (続き)

- ALTER INDEX コマンド 146
- CJK 言語における特殊文字 122
- CLEAR COMMAND LOCKS コマンド 152
- CLEAR EVENTS FOR INDEX コマンド 153
- CONTAINS 関数 135
- CREATE INDEX コマンド 155
- DB2 Accessories Suite のアンインストール 74
- DISABLE DATABASE FOR TEXT コマンド 163
- DROP INDEX コマンド 164
- ENABLE DATABASE FOR TEXT コマンド 166
- HELP コマンド 168
- Net Search Extender の比較 229
- RESET PENDING コマンド 170
- RESULTLIMIT 関数 42
- SCORE 関数 41, 138
- SCORE 関数を使用して検索 124
- SQL 116
- SQL 検索関数 135
- START FOR TEXT コマンド 171
- STOP FOR TEXT コマンド 173
- UPDATE INDEX コマンド 174
- XML 検索関数 135
- XML 名前空間 43
- XML 文書 122, 129
- XML 列 140
- XQuery 116
  - xmlcolumn-contains 140

## DB2 Text Search 用のビュー

- 索引レベルの情報
  - 概要 221
  - SYSIBMTS.TSCOLLECTIONNAMES 226
  - SYSIBMTS.TSCONFIGURATION 226
  - SYSIBMTS.TSEVENT 226
  - SYSIBMTS.TSSTAGING 228
- データベース・レベルの情報
  - 概要 221
  - SYSIBMTS.TSDEFAULTS 221
  - SYSIBMTS.TSLOCKS 222

## DB2 インフォメーション・センター

- 更新 267, 269
- バージョン 267

## DB2 サーバー

- インストール
  - Windows 52

## DB2 セットアップ・ウィザード

- インストール
  - DB2 サーバー (Linux)/DB2 サーバー (UNIX) 55

## db2icrt コマンド

- 詳細情報 240

## db2idrop コマンド

- 詳細情報 250

## db2iupdt コマンド

- 詳細情報 253

## db2iupgrade コマンド

- 詳細情報 237

## db2ts コマンド

- ALTER INDEX 146
- CLEANUP FOR TEXT 151
- CLEAR COMMAND LOCKS 152
- CLEAR EVENTS FOR INDEX 153
- CREATE INDEX 155
- DISABLE DATABASE FOR TEXT 163
- DROP INDEX 164
- ENABLE DATABASE FOR TEXT 166
- HELP 168
- RESET PENDING 170
- START FOR TEXT 171
- STOP FOR TEXT 173
- UPDATE INDEX 174

## DISABLE DATABASE FOR TEXT Text Search コマンド 163

## DROP INDEX Text Search コマンド 164

## E

## ENABLE DATABASE FOR TEXT Text Search コマンド 166

## H

## HELP コマンド

- Text Search 168

## L

## Linux

- インストール
  - 応答ファイル 60
  - DB2 サーバー 55

## N

## Net Search Extender

- Text Search との比較 229

## R

## RESET PENDING DB2 Text Search コマンド 170

## S

## SCORE 関数

- テキスト検索索引の検索 138

## SQL ステートメント

- ヘルプ
  - 表示 266

## START FOR TEXT Text Search コマンド 171

## STOP FOR TEXT Text Search コマンド 173

## SYSIBMTS.TSINDEXES ビュー 224

## SYSIBMTS.TSSERVERS 223

## SYSTS\_ADMIN\_CMD プロシージャ 179

SYSTS\_ALTER ストアード・プロシージャー 181  
SYSTS\_CLEANUP ストアード・プロシージャー 187  
SYSTS\_CLEAR\_COMMANDLOCKS ストアード・プロシージャ  
ー 189  
SYSTS\_CLEAR\_EVENTS ストアード・プロシージャー 191  
SYSTS\_CREATE プロシージャー 196  
SYSTS\_DISABLE プロシージャー 205  
SYSTS\_DROP ストアード・プロシージャー 207  
SYSTS\_ENABLE ストアード・プロシージャー 210  
SYSTS\_START プロシージャー 193  
SYSTS\_UPDATE ストアード・プロシージャー 212  
SYSTS\_UPGRADE\_CATALOG ストアード・プロシージャー  
215  
SYSTS\_UPGRADE\_INDEX ストアード・プロシージャー 217

## T

TCP/IP

サービス・ファイルの更新 72

Text Search 28

## U

UNIX

インストール

DB2 セットアップ・ウィザード 55

応答ファイル・インストール 60

UPDATE INDEX Text Search コマンド 174

## W

Windows

インストール

DB2 サーバー (DB2 セットアップ・ウィザードを使用  
した) 52

応答ファイル

使用したインストール 60

Windows サービス 63

## X

XML

DB2 Text Search

検索構文 129

EBNF グラマー 122

XML 名前空間 43

XML 列

テキスト検索 140

xmlcolumn-contains 関数 140

XQuery 関数

xmlcolumn-contains 140







Printed in Japan

SA88-4692-00



**日本アイ・ビー・エム株式会社**  
〒103-8510 東京都中央区日本橋箱崎町19-21

Spine information:

**IBM DB2 10.1 for Linux, UNIX, and Windows**

**Text Search ガイド**

