

IBM

@server

iSeries

DB2 Universal Database Extenders for iSeries
テキスト・エクステンダー
管理およびプログラミングの手引き

SH88-8521-01
(英文原典：SH12-6720-01)





@server

iSeries

DB2 Universal Database Extenders for iSeries

テキスト・エクステンダー

管理およびプログラミングの手引き

SH88-8521-01

(英文原典：SH12-6720-01)

ご注意

本書の情報およびそれによってサポートされる製品を使用する前に、259ページの『特記事項』に記載する一般情報をお読みください。

注: このPDFは同じ資料番号のドキュメントを日本語検索可能にしたものです。以前のものと内容の違いはありません。

この版は、DB2 Universal Database Extenders for iSeries バージョン 7 リリース 2 の、DB2 Universal Database テキスト・エクステンダー・オプション (5722-DE1)、OS/400 バージョン 5 リリース 2 のフィーチャー (5722-SS1)、および新版で特に指示があるまではこれ以降のすべてのリリースおよび変更に適用されます。

本書は、SH12-6720-00 の改訂版です。

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

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

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

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

(URL は、変更になる場合があります)

原典： SH12-6720-01
iSeries
DB2 Universal Database Extenders for iSeries
Text Extender
Administration and Programming

発行： 日本アイ・ビー・エム株式会社

担当： ナショナル・ランゲージ・サポート

第1刷 2002.8

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

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

© Copyright International Business Machines Corporation 2001, 2002. All rights reserved.

© Copyright IBM Japan 2002

目次

本書について	vii
本書の対象読者	vii
本書の使用方法	vii
構文図の読み方	viii
関連情報	ix
ご意見をお寄せいただく場合	ix

第 1 部 ガイド 1

第 1 章 DB2 テキスト・エクステンダーの概説 3

DB2 クライアント/サーバー環境における DB2 テキスト・エクステンダー	4
DB2 テキスト・エクステンダー・プログラムの呼び出し方法	6
DB2 テキスト・エクステンダーによる IASP データベースのサポート方法	7

第 2 章 インストール、セットアップ、および保守 9

DB2 テキスト・エクステンダー環境の設定	9
インストール検査のためにサンプル・データベースを準備する	9
構成	10
テキスト構成設定値	10
テキスト構成の変更	11
DB2 テキスト・エクステンダー・サーバーの設定および保守	11
DB2 テキスト・エクステンダー・インスタンスの作成	11
DB2 テキスト・エクステンダー・サーバーの開始および停止	12
索引と使用可能なサーバーのバックアップおよび復元	12
障害のトレース	13
DB2 テキスト・エクステンダーの SQL 表を準備する	14
ユーザーに権限を提供する	14
マイグレーション	16

第 3 章 はじめに 19

テキストを検索可能にする簡単な例	19
------------------	----

第 4 章 必要に応じた検索の計画 21

テキスト文書を索引付けする理由	21
サポートされる文書形式	23
HTML 文書に関する特別な考慮事項	23
XML 文書に関する特別な考慮事項	24
サポートされていない文書形式の使用	24
言語	26

CCSID	26
EBCDIC	26
ASCII	27
DBCS	27
テキストの保管および使用可能化の際に起きるコード・ページに関する問題の回避	28

検索のタイプ	31
基本語形検索	31
一致語句検索	33
ファジー検索または DBCS 文書の検索	33
索引タイプの変更	34
表に対する 1 つまたは複数のテキスト索引の作成	34
索引サイズの計算	35
索引の更新	35
構造化文書 (セクション・サポート) での作業	37
フラット・ファイルおよび HTML 文書	37
XML 文書	38
辞書、停止ワード・リスト、省略後リスト、および言語パラメーター	40
停止ワードと省略語ファイルの変更	41

第 5 章 テキストを検索可能にする 43

テキストを検索可能にする前の準備	43
DB2 テキスト・エクステンダー・コマンド行プロセッサの開始	44
コマンド行プロセッサのヘルプ	45
データベースの使用可能化	45
テキスト表の使用可能化 (オプション)	46
例	47
テキスト列の使用可能化	49
ハンドル列の追加	49
文書情報の設定	50
ログ表の作成	50
索引の作成	50
例	51
ラージ表でのテキスト列の使用可能化	51
サポートされないデータ・タイプのテキスト列の使用可能化	52
外部テキスト・ファイルの使用可能化	53
例	54
セッションの終了	55

第 6 章 検索方法 57

検索関数の構文例の情報源	57
DB2TX.SAMPLE サンプル表	57
外部ファイルのハンドル	61
現在の関数パスの設定	61
テキストの検索	61
照会の作成	62
検出された一致数の検索と入手	63
検出テキスト文書のランクの検索と入手	63

検索引き数の指定	63
複数の用語の検索	63
ブール演算子 AND および OR を使用した検索	64
用語の変化形の検索	64
用語の一部 (マスキング文字) の検索	65
マスキング文字をすでに含む用語の検索	65
任意の順序での用語の検索	66
同じ文または段落中の用語の検索	66
構造化文書のセクションでの用語の検索	66
用語の同義語の検索	67
基本語形検索の実行	67
ブール演算子 NOT を使用した検索	68
ファジー検索	68
語句の境界の考慮	69
発音の似た語の検索	69
シソーラス検索	69
フリー・テキストと混成検索	70
直前の検索の詳細化	70
ハンドル内の情報の設定と抽出	72
ハンドルからの情報の抽出	73
ハンドル内の情報の変更	74
検索パフォーマンスの向上	75

第 7 章 管理 77

AS/400 オペレーション・ナビゲーターを使用する 管理コマンドの実行	77
テキスト索引の保守	77
索引の更新	78
外部ファイルの索引の更新	78
索引の設定の変更	79
索引状況のリセット	79
索引イベントの削除	80
索引の再編成	80
有用な情報の入手	81
使用可能状況情報の表示	81
テキスト構成設定値の表示	82
索引状況の表示	82
エラー・イベントの表示	84
索引設定の表示	85
列のテキスト設定の表示	85

DB2 テキスト・エクステンダー・カタログ・ビュー の処理	86
テキスト準備処理の取り消し	88
テキスト列を使用不可にする	88
テキスト・ファイルを使用不可にする	89
テキスト表を使用不可にする	89
サーバーを使用不可にする	90

第 8 章 検索およびブラウズのための

API 関数の使用法 91	
アプリケーションのセットアップ	91
アプリケーションのリンク	91
API 関数の概説	91
テキストの検索	93
検索結果表の入手 (DesGetSearchResultTable)	93
テキストのブラウズ	94

ブラウズ情報の入手 (DesGetBrowseInfo)	94
ブラウズ・セッションの開始 (DesStartBrowseSession)	94
文書のオープン (DesOpenDocument)	94
一致情報の入手 (DesGetMatches)	95
文書のクローズ (DesCloseDocument)	95
ブラウズ・セッションの終了 (DesEndBrowseSession)	96
ブラウズ情報の解放 (DesFreeBrowseInfo)	96

第 2 部 参照情報 97

第 9 章 クライアント用のテキスト準備コ マンドと管理コマンド 99

CHANGE INDEX SETTINGS	101
CHANGE TEXT CONFIGURATION	103
CONNECT	106
DELETE INDEX EVENTS	107
DISABLE SERVER FOR DB2TEXT	108
DISABLE TEXT COLUMN	109
DISABLE TEXT FILES	110
DISABLE TEXT TABLE	111
ENABLE SERVER FOR DB2TEXT	112
ENABLE TEXT COLUMN	113
ENABLE TEXT FILES	120
ENABLE TEXT TABLE	123
GET INDEX SETTINGS	127
GET INDEX STATUS	129
GET STATUS	130
GET TEXT CONFIGURATION	131
GET TEXT INFO	132
QUIT	133
REORGANIZE INDEX	134
RESET INDEX STATUS	135
UPDATE INDEX	136

第 10 章 サーバー用の管理コマンド 137

TXICRT	138
TXIDROP	139
TXSAMPLE	140
TXSTART	141
TXSTATUS	142
TXSTOP	143
IMOTHESC	144
IMOTHESN	146
IMOTRACE	148
TXVERIFY	152

第 11 章 検索関数 153

DB2 テキスト・エクステンダー特殊タイプ	153
DB2 テキスト・エクステンダー関数の要約	154
CCSID	155
CONTAINS	156
FILE	157
FORMAT	158

LANGUAGE	159
NO_OF_DOCUMENTS	160
NO_OF_MATCHES	161
RANK	162
REFINE	163
SEARCH_RESULT	164

第 12 章 検索引き数の構文 165

検索引き数	166
-----------------	-----

第 13 章 検索およびブラウズのための

API 関数 175

DesCloseDocument	176
DesEndBrowseSession	177
DesFreeBrowseInfo	178
DesGetBrowseInfo	179
DesGetMatches	182
DesGetSearchResultTable	187
DesOpenDocument	191
DesStartBrowseSession	193

第 14 章 サンプル API プログラム . . . 195

第 15 章 基本語形索引と一致語句索引

の言語的な処理 197

索引作成時の言語的な処理	197
基本テキスト分析	198
用語の基本形への変形 (lemmatization)	202
停止ワードのフィルター操作	202
検索のための言語的な処理	202
同義語	203
シソーラスの拡張	203
発音の拡張	204
文字と語のマスキング	204

ブラウズのための言語的な処理	204
ステージ 1: 正規化と用語拡張	204
ステージ 2: 拡張マッチング	205
シソーラスの概念	206
用語	207
関係	207
Ngram シソーラスの関係	209
シソーラスの作成	210
Ngram シソーラスの作成	213

第 16 章 構成ファイル 217

クライアント構成ファイル	217
サーバー構成ファイル	218

第 17 章 戻りコード 221

第 18 章 メッセージ 227

DB2 テキスト・エクステンダーの関数により戻される SQL の状態	227
DB2 テキスト・エクステンダーからのメッセージ	230

第 19 章 検索エンジンの理由コード 241

第 20 章 エラー・イベント理由コード 243

第 3 部 付録 257

特記事項 259

商標	261
--------------	-----

用語集 263

索引 267

本書について

本書は、テキスト・データを検索するために DB2[®] UDB server for iSeries[™] を準備および保守するにあたって、DB2 テキスト・エクステンダーを使用する方法を説明したものです。また、DB2 テキスト・エクステンダーが提供している SQL 関数およびアプリケーション・プログラミング・インターフェース (API) を使用して、それらのタイプのデータにアクセスし、操作する方法も説明しています。DB2 テキスト・エクステンダーの機能をプログラムの SQL ステートメントに組み込むことにより、また API を組み込むことにより、強力で融通のきくテキスト検索プログラムを作成することができます。

本書で「DB2」と表記する場合は、DB2 UDB を指しています。

本書の対象読者

本書は、DB2 管理の概念、ツール、および技法に精通している DB2 データベース管理者を対象としています。

また、SQL と、DB2 アプリケーション・プログラムに使用できる 1 つまたは複数のプログラミング言語に精通した、DB2 アプリケーション・プログラマーも対象としています。

本書の使用方法

本書は以下のように構成されています。

『第 1 部: ガイド』

この部分では、DB2 テキスト・エクステンダーについて概説すると共に、インストール後の設定方法、計画を立てる際の考慮事項について説明します。さらに、テキストを検索できるように、DB2 データベースを準備して、保守する方法についても説明します。

DB2 テキスト・エクステンダーについてあまり詳しくなく、テキストを検索するために DB2 テキスト・エクステンダー関数および API の使用方法を知りたい場合は、この部分をお読みください。

『第 2 部: 参照』

この部分では、DB2 テキスト・エクステンダーの関数、API、コマンド、およびメッセージやコードなどの診断情報についての参照情報を示します。

DB2 テキスト・エクステンダーの概念およびタスクは理解しているものの、特定の DB2 テキスト・エクステンダー関数、API、コマンド、メッセージ、またはコードについての情報が必要な場合は、この部分をお読みください。

構文図の読み方

本書では、構文は次のように定義された構造を用いて記述されています。

- 構文図は、左から右、上から下に、線に沿って読みます。▶▶— 記号は、ステートメントの開始を示します。

—▶▶ 記号は、構文図が次の行に続くことを示します。

▶▶— 記号は、ステートメントが前の行から続いていることを示します。

—▶▶ 記号は、ステートメントの終わりを示します。

- 必須項目は、水平線 (メインパス) の線上に示されます。

▶▶—必須項目—▶▶

- オプション項目は、メインパスの下に示されます。

▶▶—
└オプション項目┘—▶▶

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

項目から 1 つを選択しなければならない場合、スタック内の項目の 1 つがメインパスの線上に示されます。

▶▶—必須選択項目 1┘
└必須選択項目 2┘—▶▶

項目から選択しなくてもよい場合には、併記された項目全体が主線の下に記載されます。

▶▶—
└オプション項目 1┘
└オプション項目 2┘—▶▶

併記された項目の上を矢印が折り返している場合、併記された項目から 2 つ以上を選択することが可能であることを示します。

▶▶—
└オプション項目 1┘
└オプション項目 2┘—▶▶

- キーワードは大文字で表記されています。キーワードの記載は、表記どおりでなければなりません。変数は小文字で表記されています (たとえば, srcpath)。変数は、構文中でのユーザー指定の名前または値を表します。
- 句読記号、括弧、算術演算子、または同様の他の記号が示されている場合、それらを構文の一部として入力しなければなりません。

関連情報

- データベースのパフォーマンスおよび照会の最適化
- データベース・プログラミング (SD88-5035)
- DB2 マルチ・システム
- DDS 解説書
- 分散データ管理
- 分散データベース・プログラミング
- File Management
- Query for iSeries ご使用の手引き (SD88-5025)
- Query 管理機能プログラミング (SC88-4018)
- Query Manager ご使用の手引き (SC88-4015)
- SQL 呼び出しレベル・インターフェース (ODBC)
- SQL プログラミング
- ホスト言語を使用した SQL プログラミング
- SQL 解説書

注: 資料番号に記述されていないものは、Information Center CD に収録されています。

ご意見をお寄せいただく場合

提供する情報の正確性と品質を高めるためには、お客様からのご意見のフィードバックは重要です。本書やその他の DB2 Extenders™ 資料に対するご意見をお持ちの際は、本書の中表紙の裏に記載されているアドレスにご意見をご送付ください。

- 資料名
- 資料番号
- ご意見をお寄せいただくページの番号またはトピック

本書について

第 1 部 ガイド

第 1 章 DB2 テキスト・エクステンダーの概説

プログラマーは DB2 テキスト・エクステンダーを使って、アプリケーションにテキスト文書用の SQL 照会を組み込むことができます。

DB2 テキスト・エクステンダーは、テキスト文書をデータベースへ保管可能にする DB2 UDB for iSeries の機能を利用して、SQL 照会に全テキスト検索の機能を追加します。

DB2 UDB for iSeries ユーザーおよびアプリケーション・プログラマーは、DB2 テキスト・エクステンダーを使用して、このような文書を短時間に、多様かつ高機能な方法で検索することができます。DB2 テキスト・エクステンダーの利点は、何千もの大量のテキスト文書を検索し、直接要求したものだけでなくその変形語や同義語も検出できるということです。

ユーザーは、DB2 UDB for iSeries データベース内に格納されているテキスト文書だけでなく、ファイルに格納されたテキスト文書も検索することができます。

DB2 テキスト・エクステンダーの中心となるのが、197 ページの『第 15 章 基本語形索引と一致語句索引の言語的な処理』で説明されている、高性能な言語検索技術です。これにより、アプリケーションは、テキスト文書にアクセスし、さまざまな方法で検索を行うことができます。以下をアプリケーションで実行できます。

- 特定のテキストまたはワードの同義語が含まれている文書の検索、あるいは同じ文や段落中などに近接する検索用語が含まれている文書の検索。
- 語および文字の前、真ん中、および後の部分をマスクしたワイルドカード検索。
- さまざまな文書形式の多言語の文書検索。
- 検索語と似たつづりの語の「ファジー」検索。これは、つづりを誤った語でも検索できるので便利です。
- 検索引き数が自然言語で表されるフリー・テキスト検索。
- 検索項目と類似した発音の語の検索。

テキスト検索と業務データ照会を結合できます。たとえば、アプリケーション中の SQL 照会をコーディングして、ある期間内に特定の作成者により作成された、特定の語または句を含むテキスト文書を検索することができます。また、DB2 テキスト・エクステンダー・プログラミング・インターフェースを使用すれば、アプリケーション・ユーザーが文書をブラウズすることもできます。

DB2 UDB for iSeries の SELECT 照会に全テキスト検索を組み込むことによって、高機能の検索を行うことができます。次の SQL ステートメントはこの一例です。

```
SELECT * FROM MyTextTable
WHERE version = '2'
AND DB2TX.CONTAINS (
    DB2BOOKS_HANDLE,
    "authorization"
    IN SAME PARAGRAPH AS "table"
    AND SYNONYM FORM OF "delete") = 1
```

DB2TX.CONTAINS は、いくつかある DB2 テキスト・エクステンダー検索関数のうちの 1 つです。DB2BOOKS_HANDLE は、検索するテキスト文書を含んだ DB2BOOKS 列を参照するハンドル列の名前です。上記ステートメントのそれ以外の部分は、table、ならびに delete または delete の任意の同義語と同じ段落に現れる authorization を探す検索引き数の例です。

DB2 クライアント/サーバー環境における DB2 テキスト・エクステンダー

図 1 は、DB2 テキスト・エクステンダーを DB2 クライアント/サーバー環境にどのように組み込むのかを示しています。

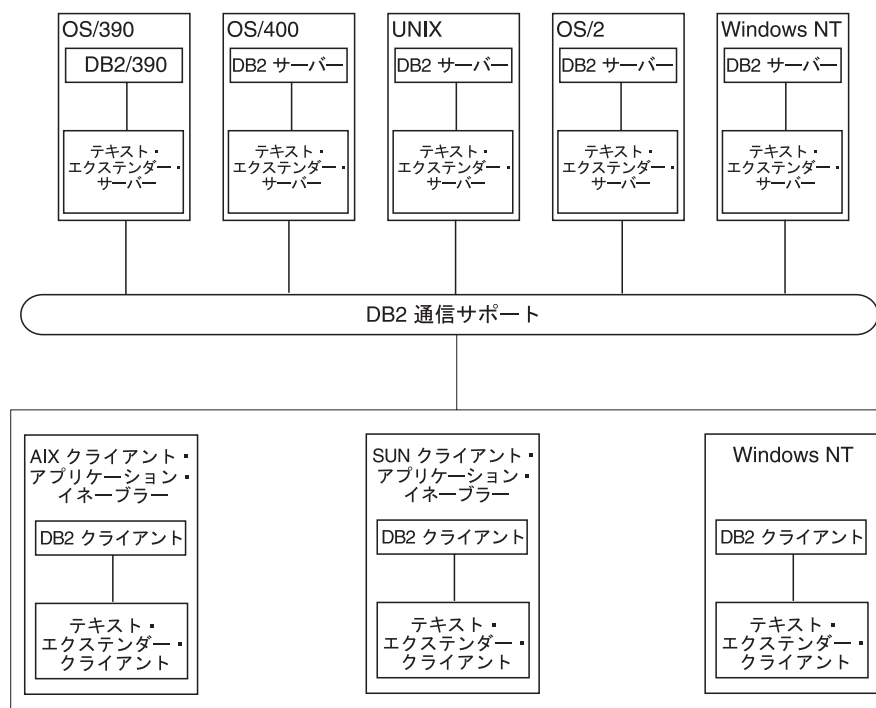


図 1. DB2 クライアント/サーバー環境への DB2 テキスト・エクステンダーの統合

クライアント用の DB2 通信サポート・プロトコル (TCP/IP または NETBIOS など) のリストについては、該当するプラットフォーム用の「DB2 概説およびインストール」を参照してください。

DB2 テキスト・エクステンダーの主なソフトウェア部品は、DB2 サーバーと同じマシンにインストールされます。1 つの DB2 サーバー・インスタンスと共にインストールできる DB2 テキスト・エクステンダー・サーバーは 1 つだけです。

DB2 テキスト・エクステンダーのインストール内容は必要に応じて変更ことができ、以下のものを組み込むことができます。

- 図 1 に示されているオペレーティング・システム上での、1 つまたは複数の DB2 テキスト・エクステンダー・サーバー (UNIX[®] には AIX[®]、SUN-Solaris、および HP-UX ワークステーションが含まれる)
- 1 つまたは複数のリモート DB2 テキスト・エクステンダー・サーバーへのアクセス権を持つ、AIX、SUN-Solaris、Windows NT[®]、および Windows 2000 クライアント

- ローカルなサーバーを含み、リモート・サーバーへのアクセス権を持つ AIX クライアント

図 2 は、典型的な DB2 テキスト・エクステンダー構成を示しています。クライアントから DB2 テキスト・エクステンダーを実行するには、まず 1 台の DB2 クライアントといくつかの DB2 テキスト・エクステンダー・ユーティリティをインストールしておかなければなりません。これらのユーティリティは、厳密にはクライアントではありませんが、DB2 テキスト・エクステンダー「クライアント」を構成するものです。クライアントは、DB2 クライアント接続を介してサーバーと通信します。

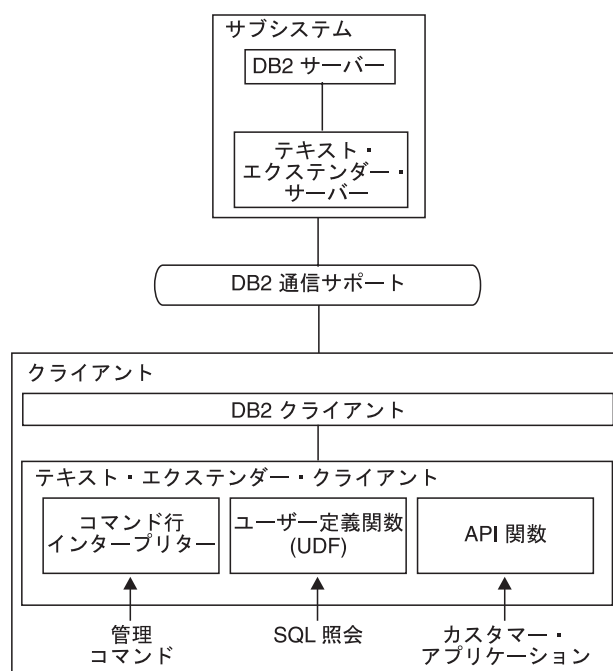


図 2. DB2 テキスト・エクステンダー構成

DB2 テキスト・エクステンダーの主なコンポーネントは以下のとおりです。

- コマンド行インタープリター**。コマンドを使用して、検索されるテキストを列中に作成したり、テキスト索引を保守したりすることができます。
- SQL 関数**。これを SQL 照会に組み込んで、テキストの検索を行ったり、テキスト中に検索項目が出てくる回数などを検出したりすることができます。分かりやすくするために、この図では SQL 関数をクライアント上に示しています。それらは SQL 照会の一部として使用できるからです。実際には、SQL 関数はサーバー・インストールの一部であり、サーバーで実行されます。しかし、これらの関数は、DB2 テキスト・エクステンダー・クライアントをインストールしないでも任意の DB2 クライアントから使用できます。
- アプリケーション・プログラミング・インターフェース (API)**。これは、テキスト検索を行ったり検索結果を表示するために C プログラムで呼び出すことのできる関数から成ります。

ヒント

DB2 テキスト・エクステンダー・クライアント・ユーティリティには、テキスト準備関数、管理関数、および API が備わっています。(これらの関数はサーバー上で使用できます。) これらの関数を使用するには、DB2 テキスト・エクステンダー・クライアントをインストールしなければなりません。DB2 テキスト・エクステンダー・サーバーだけでテキスト準備および管理を行う場合は、DB2 テキスト・エクステンダー・クライアント・ユーティリティは、API 関数を使用するクライアントだけにインストールすれば十分です。

クライアント側で、DB2 UDB for iSeries SQL ステートメントを使用する検索機能しか必要でない場合は、DB2 テキスト・エクステンダー・クライアントをインストールする必要はありません。すべての通信は DB2 UDB for iSeries によって処理され、DB2 テキスト・エクステンダー検索エンジンは、サーバーでのみ稼働します。

DB2 テキスト・エクステンダー・プログラムの呼び出し方法

本書で説明されている DB2 テキスト・エクステンダー・プログラムはすべて、AS/400® 固有の呼び出し規則を使用して呼び出すことができます。

```
CALL PGM(QDB2TX/DB2TX) PARM('ENABLE SERVER FOR DB2TEXT')
```

CALL ステートメントでライブラリー名を指定しない場合は、DB2 テキスト・エクステンダーの製品ライブラリー QDB2TX を、ライブラリー・リスト ("ADDLIB LIB(QDB2TX)") に追加することもできます。

また別の選択肢として、以下のコマンドを使用して、DB2 テキスト・エクステンダーのコマンドが QSHELL 環境で開始されるように、AS/400 システムを変更することもできます。

- DB2TX プログラム用の記号リンクを追加するコマンド

```
QSYS/ADDLNK OBJ('/QSYS.LIB/QDB2TX.LIB/DB2TX.PGM')
NEWLNK('/usr/bin/db2tx')
```

- 記号リンクの所有権を変更するコマンド

```
QSYS/CHGOWN OBJ('/usr/bin/db2tx') NEWOWN(QDESUSR) SYMLNK(*YES)
```

これで、管理コマンドを QSHELL で実行することができます。たとえば次のようになります。db2tx enable text column db2tx.sample comment handle h_p

その他の DB2 テキスト・エクステンダー・プログラム (たとえば TXSTART) を、それに対応する DB2 テキスト・エクステンダー・プログラムで上記のコマンドを実行して、QSHELL 環境で実行することもできます。

注

QShell を使用するには、「AS/400 - QShell インタープリター」がシステムにインストールされていなければなりません。

DB2 テキスト・エクステンダーによる IASP データベースのサポート方法

OS/400 バージョン 5 リリース 2 では、独立 ASP データベースと、それに対応するネーム・スペースという新しい概念が提供されています。DB2 テキスト・エクステンダー機能は、システム/ユーザー ASP 上のデータベース、または独立 ASP 上の 1 つあるいは複数のデータベースと共に使用できますが、これら両方と共に使用できません。

DB2 テキスト・エクステンダーは、現行のアクティブなネーム・スペースと関連付けられているデータベースに対しては暗黙接続を使用します。以下の方法をお勧めします。

- SETASPGRP を使用して、作業を行うデータベースにネーム・スペースを設定する
- DB2 テキスト・エクステンダー・コマンドで暗黙接続を使用する

これにより、正しいネーム・スペースが常に使用できるようになります。

第 2 章 インストール、セットアップ、および保守

DB2 テキスト・エクステンダー環境の設定

DB2 テキスト・エクステンダーを使用する前に、まず、必要な環境変数を設定しなければなりません。プログラム TXPROFILE を実行してください。このプログラムは、DB2 テキスト・エクステンダーの製品ライブラリー QDB2TX に保管されています。この制御言語プログラムのソースは、SAMPLES ファイルに保管されています。

ヒント

TXPROFILE プログラムを、対応するユーザー・プロファイルの初期プログラムに指定しておくこと、ログオン・プロセス中はいつでもこの環境を設定することができます。

インストール検査のためにサンプル・データベースを準備する

DB2 テキスト・エクステンダーは、サンプル・データベースを作成するユーティリティを備えています。これらは、インストール直後に DB2 テキスト・エクステンダーの検索機能をテストするため、またインストール後、随時に汎用テストを行うために迅速にテキストを作成するのに便利です。

サンプル・データベースを準備する前に、以下のステップを完了しておく必要があります。

- ユーザーにテキスト・エクステンダーの管理者権限があることを確認する。テキスト・エクステンダーのインスタンス作成にはこの権限が必要です。詳細は、14 ページの『ユーザーに権限を提供する』を参照してください。
- DB2 テキスト・エクステンダー環境を設定する。『DB2 テキスト・エクステンダー環境の設定』を参照してください。
- DB2 テキスト・エクステンダー・インスタンスを作成する。11 ページの『DB2 テキスト・エクステンダー・インスタンスの作成』を参照してください。
- DB2 テキスト・エクステンダー・サーバーを始動する。12 ページの『DB2 テキスト・エクステンダー・サーバーの開始および停止』を参照してください。

サンプル・データベースを作成するためには、次のようにします。

- オペレーティング・システムのプロンプトに応じて、以下のコマンドを実行する。

```
CALL PGM(QDB2TX/TXSAMPLE)
```

このコマンドは、以下の処理を行います。

指定されたデータベースに接続する。

表 db2tx.sample を作成する。

サンプルの英語文書をインポートして、表を埋める。

次の索引タイプを指定して、表内のテキスト列を使用可能にする。

一致語句
正規化された一致語句
基本語形
Ngram
Ngram (大文字小文字の区別あり)
テキスト索引が作成されるのを待つ。

構成

このセクションでは、DB2 テキスト・エクステンダーの構成情報について説明します。これによって、DB2 テキスト・エクステンダー・パラメーターのデフォルト値を指定することができます。

テキスト構成設定値

各データベースには、以下を含むテキスト構成設定値があります。

- テキスト特性
- 索引特性
- 処理特性

データベースを DB2 テキスト・エクステンダーに対して使用可能にするとき、これらの設定値が設定されます。ENABLE SERVER コマンドは、コマンドで指定する設定値、またはここで説明する初期設定値のいずれかを使用します。これらのデフォルト設定値を表示または変更することができます。82 ページの『テキスト構成設定値の表示』と 11 ページの『テキスト構成の変更』を参照してください。

テキスト特性

21 ページの『第 4 章 必要に応じた検索の計画』では、DB2 テキスト・エクステンダーでサポートされる文書形式、言語、および CCSID について説明します。これらのデフォルト値は、さまざまなコマンドで必要となります。

FORMAT 初期設定: TDS
LANGUAGE 初期設定: データベースに対して設定される言語
CCSID 初期設定: サブシステムに対して設定される CCSID

索引特性

DIRECTORY 索引を保管するためのディレクトリー。
INDEXTYPE 使用される索引タイプ。詳しくは 31 ページの『検索のタイプ』を参照してください。

処理特性

UPDATEINDEX

最初の索引更新の時期を、使用可能化ステップ中にすぐに行う、更新頻度の設定に基づいて後に行う (NOUPDATE)、または明示的な UPDATE INDEX コマンドの結果として行うの、いずれかに決定する設定。

初期設定: UPDATE

COMMITCOUNT

いくつの挿入ステートメントまたは更新ステートメントが出されると、DB2 テキスト・エクステンダーが DB2 UDB for iSeries コミット用ステートメントを出すかを、決めるための設定です。51 ページの『ラージ表でのテキスト列の使用可能化』を参照してください。

初期設定: 0

テキスト構成の変更

DB2 テキスト・エクステンダーが初めてインストールされるときに、デフォルト値がテキスト構成に設定されます。現行のテキスト構成値を表示するには、82 ページの『テキスト構成設定値の表示』を参照してください。

索引の作成時にデフォルト値として使用されるテキスト構成を変更するには、次のように入力します。

```
db2tx CHANGE TEXT CFG USING settings
```

例:

今後の索引のためのデフォルト索引タイプとデフォルト・ディレクトリーを変更するには次のようにします。

```
CALL PGM(QDB2TX/DB2TX) PARM('CHANGE TEXT CONFIGURATION USING
INDEXTYPE precise
INDEXOPTION normalized
DIRECTORY /myfs/indexes')
```

DB2 テキスト・エクステンダー・サーバーの設定および保守

新規にインストールした DB2 テキスト・エクステンダーに関する次のステップは、DB2 テキスト・エクステンダー・サーバーを DB2 テキスト・エクステンダー・クライアントが使用できるように設定することです。通常これは、DB2 テキスト・エクステンダー・サーバー管理者によって行われ、以下の作業が含まれます。

1. DB2 テキスト・エクステンダー・インスタンスの作成
2. DB2 テキスト・エクステンダー・サーバーの開始

DB2 テキスト・エクステンダー・サーバーを保守するための他のタスクは、以下のとおりです。

1. DB2 テキスト・エクステンダーの索引および使用可能なデータベースのバックアップと復元
2. 障害のトレース

コマンド構文については、137 ページの『第 10 章 サーバー用の管理コマンド』を参照してください。

DB2 テキスト・エクステンダー・インスタンスの作成

DB2 テキスト・エクステンダーを使った作業を開始する前に、DB2 テキスト・エクステンダー・インスタンスを作成する必要があります。このインスタンスにより、索引を保守および保管する管理環境が提供されます。

サーバーの設定および保守

この環境は、DB2 Extenders のインスタンスを作成する前に、セットアップしなければならない (詳しくは 9 ページの『DB2 テキスト・エクステンダー環境の設定』を参照) 点に注意してください。

インスタンスを作成するには、次のように入力します。

```
CALL PGM(QDB2TX/TXICRT)
```

インスタンスを除去するには、次のように入力します。

```
CALL PGM(QDB2TX/TXIDROP)
```

インスタンスの作成が必要なのは一度のみであり、その後は除去されるまでアクティブのままとなります。

DB2 テキスト・エクステンダー・サーバーの開始および停止

文書の索引付けや検索を行うには、まず、DB2 テキスト・エクステンダー・サーバーを始動しなければなりません。

DB2 テキスト・エクステンダー・サーバーを始動するには、テキスト・エクステンダー管理者としてログオンした後、次のように入力します。

```
CALL PGM(QDB2TX/TXSTART)
```

DB2 テキスト・エクステンダー・サーバーの状況を表示するには、以下のように入力します。

```
CALL PGM(QDB2TX/TXSTATUS)
```

DB2 テキスト・エクステンダー・サーバーを停止するには、ユーザー・プロファイル QDESADM でログオンしてから、次のように入力します。

```
CALL PGM(QDB2TX/TXSTOP)
```

索引と使用可能なサーバーのバックアップおよび復元

使用可能とされたデータベースおよび DB2 テキスト・エクステンダーが作成したテキスト索引をバックアップおよび復元することができます。

バックアップするには、

1. どの表が DB2 テキスト・エクステンダーにより使用可能とされたかを調べます。次のように入力します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET STATUS')
```
2. データベースにより使用される索引ディレクトリーの名前を見つけます。次のように入力します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET INDEX SETTINGS table-name')
```
3. 索引の更新が実行されていないことを確認してから、次のコマンドを使用して DB2 テキスト・エクステンダー・サーバーを停止します。

```
CALL PGM(QDB2TX/TXSTOP)
```
4. 索引ディレクトリー、およびそれらのサブディレクトリー `index` および `work` をバックアップします。
5. ファイル `imomastr.dat` をバックアップします。このファイルは以下の場所にあります。


```
/QIBM/UserData/DB2Extenders/Text/instance/txins000
```

- DB2 テキスト・エクステンダー・サーバーを再始動します。

```
CALL PGM(QDB2TX/TXSTART)
```

復元するには、

- DB2 テキスト・エクステンダー・サーバーを停止します。
- 既存の imomastr.dat ファイルを保管します。
- imomastr.dat ファイルのバックアップ・コピーを復元します。
- 索引ディレクトリーのバックアップ・コピーを前と同じパスに復元します。
- DB2 テキスト・エクステンダー・サーバーを再始動します。

```
CALL PGM(QDB2TX/TXSTART)
```

障害のトレース

IBM 担当員にエラーを報告する必要がある場合は、エラー箇所を突き止めるのに使用できる情報をファイルに書き込むために、トレース機能に切り替えるよう要求される場合があります。このトレース機能の使用は、IBM サポート・センターの担当員またはお客様の技術サポート担当者からの指導がある場合のみとしてください。

トレース機能に切り替えると、システム・パフォーマンスに影響するため、この機能はエラー条件が発生した場合のみ使用するようになります。

トレース機能をオンに切り替えるには、次のように入力します。

```
CALL PGM(QIMO/IMOTRACE) PARM('ON' '[options]')
```

構文、イベントのリスト、およびコンポーネントは、148 ページの『IMOTRACE』に記載してあります。また、これ以外のオプションも記載してあります。

トレース・レコードの ID に基づいて、それらの各レコードの受け入れまたは排除を行う「マスク」を指定することによって、トレース機能にフィルターをかけることができます。デフォルトには、すべてのトレースが行われます。

マスクは、2.2-6.1,3.* のように、ピリオドで区切られた 4 つの部分からなります。ここで、

- 2** DB2 UDB DB2 テキスト・エクステンダーを示します。
- 2-6** イベント ID が 2 から 6 までの項目のみを含めます。
- 1、3** コンポーネント 1 と 3 によって報告されるイベントのみを含めます。
- *** コンポーネントのすべての関数を含めます。

特定の重大度より低いシステム・エラーは排除することができ、また、トレース・バッファがいっぱいになった場合には、最初または最後のどちらのレコードを保持するかを指定できます。

エラーを複製して、そのトレース情報を 2 進数形式でダンプ・ファイルに書き込むには、次のように入力します。

```
CALL PGM(QIMO/IMOTRACE) PARM('dump' ['dump-filename'])
```

サーバーの設定および保守

形式設定されたダンプ・ファイルを生成するには、次のように入力します。

```
CALL PGM(QIMO/IMOTRACE) PARM('format' ['dump-filename'] ['formatted-filename'])
```

出力画面上の印刷機能を使用して、トレース情報をスプール・ファイルに書き込むこともできます。

トレース情報をファイルに書き込んだ後、次のコマンドを使用してトレース機能をオフにします。

```
CALL PGM(QIMO/IMOTRACE) PARM('off')
```

DB2 テキスト・エクステンダーの SQL 表を準備する

索引付けの表を使用して DB2 テキスト・エクステンダーで検索するには、まず、以下のコマンドを使用して、SQL コレクションおよび表に、適切なオブジェクト権限を授与する必要があります。

1. コレクションのオブジェクト権限を授与する場合

```
GRTOBJAUT OBJ(QSYS/<collection-name>) OBJTYPE(*LIB) USER(QDESADM) AUT(*ALL)
```

2. 表のオブジェクト権限を授与する場合

```
GRTOBJAUT OBJ(<collection name>/<table-name>)  
OBJTYPE(*FILE) USER(QDESADM) AUT(*ALL)
```

ユーザーに権限を提供する

DB2 テキスト・エクステンダーのインストール時には、次の 2 種類のユーザー・プロファイルが作成されます。

- テキスト・エクステンダー管理者 (グループ・プロファイル **QDESADM**)
- テキスト・エクステンダー・ユーザー (グループ・プロファイル **QDESUSR**)

作成するデータベース・オブジェクトに対して、これらのユーザー・プロファイルはそれぞれ、異なるレベルのユーザー・アクセスを提供します。ユーザーに適切な権限 (個別特権で構成される権限) を提供するには、ユーザー・プロファイルを適切なグループに割り当てる必要があります。どちらのタイプのユーザーにも、2 種類の特権 (サーバーとクライアント) があります。

注

テキスト・エクステンダー管理者は、テキスト・エクステンダーのユーザー特権をすべて使用することができます。

索引付けをしたデータや表を検索するには、SQL 表およびコレクションのユーザー・グループ・プロファイルと管理者グループ・プロファイルの両方に、**付加的な特権**を追加しなければならない点にも注意してください。詳細は、この前のセクションを参照してください。

管理者権限

テキスト・エクステンダー管理者は以下の特権を使用することができます。また、これらの特権は、テキスト・エクステンダー用のデータベース・サーバーを準備するため、およびテキスト・エクステンダーの索引を作成、保守、および除去するた

めに使用されます。

表 1. テキスト・エクステンダー - 管理者特権

コマンド	タイプ	影響を受けるデータベース・オブジェクトに必要なオブジェクト権限とデータ権限
TXICRT	サーバー	なし
TXIDROP	サーバー	なし
TXSTART	サーバー	なし
TXSTOP	サーバー	なし

注

テキスト・エクステンダーの管理者権限を持つ人は、テキスト・エクステンダー内部のデータ・ファイルや索引付けの済んだ表にアクセスできるのため、この権限は必要なユーザーのみに与えることをお勧めします。

ユーザー権限

以下の特権は、テキスト・エクステンダーのユーザーにも管理者にも、使用することができます。コマンドによっては、SQL 表 (索引付けの済んだデータ、または索引付けが予定されているデータを含む) のために追加のオブジェクトおよびデータ権限が必要になります。詳細については、14 ページの『DB2 テキスト・エクステンダーの SQL 表を準備する』を参照してください。

表 2. テキスト・エクステンダー - ユーザー特権

コマンド	タイプ	影響を受けるデータベース・オブジェクトに必要なオブジェクト権限とデータ権限
TXSTATUS	サーバー	なし
TXVERIFY	サーバー	なし
TXSAMPLE	サーバー	なし
CHANGE INDEX SETTINGS	クライアント	*CHANGE(1)
CONNECT	クライアント	なし
DELETE INDEX EVENTS	クライアント	なし
DISABLE TEXT COLUMN	クライアント	*CHANGE(1)
DISABLE TEXT FILES	クライアント	*CHANGE(1)
ENABLE TEXT COLUMN	クライアント	*OBJALTER および *CHANGE(1)
ENABLE TEXT FILES	クライアント	*OBJALTER および *CHANGE(1)
ENABLE TEXT TABLE	クライアント	*OBJALTER および *CHANGE(1)
GET INDEX SETTINGS	クライアント	*USE
GET INDEX STATUS	クライアント	*USE
GET STATUS	クライアント	なし
GET TEXT CONFIGURATION	クライアント	なし
GET TEXT INFO	クライアント	なし

サーバーの設定および保守

表 2. テキスト・エクステンダー — ユーザー特権 (続き)

コマンド	タイプ	影響を受けるデータベース・オブジェクトに必要なオブジェクト権限とデータ権限
REORGANIZE INDEX	クライアント	*USE
RESET INDEX STATUS	クライアント	なし
UPDATE INDEX	クライアント	*USE
CHANGE TEXT CONFIGURATION	クライアント	なし
DISABLE SERVER	クライアント	*OBJALTER および *CHANGE(1)

(1) *CHANGE 権限は実際には必須ではありませんが、少なくとも *OBJOPR、*READ、および *UPD 権限は必須です。

検索

テキスト・エクステンダー UDF を使用してデータベース表を検索するには、ユーザー権限か管理者権限のいずれかを有している必要があります。さらに、検索する SQL 表のための、以下のオブジェクト特権およびデータ特権も必要になります。

表 3. 管理者およびユーザー・グループ・プロファイルの追加の検索特権

検索関数	影響を受けるデータベース・オブジェクトに必要なオブジェクト権限とデータ権限
CCSID	*USE
CONTAINS	*USE
FILE	*USE または CHANGE(1)
FORMAT	*USE または CHANGE(1)
LANGUAGE	*USE または CHANGE(1)
NO_OF_DOCUMENTS	*USE
NO_OF_MATCHES	*USE
RANK	*USE
REFINE	*USE

(1) どちらの権限が必要かは、使用するコマンドによって決まります。情報を検索するためには、*USE 権限のみが必要です。情報を変更するためには、*CHANGE 権限か、少なくとも *OBJOPR、*READ、*UPD の各権限が必要です。

マイグレーション

OS/400 バージョン 5 リリース 2 において、DB2 テキスト・エクステンダーでは新たに、表の値に基づく関数 SEARCH_RESULT が提供されます。詳細は、153 ページの『第 11 章 検索関数』を参照してください。通常、この関数は ENABLE SERVER FOR DB2TEXT の間に定義されます。しかし、V5R1 の上にスリップ・インストールして、既存の索引を保持する場合は、テキスト列またはデータベース・サーバーを使用不可とはしないでください。この場合、以下のコマンドを実行してください。

サーバーの設定および保守

```
SBMJOB CMD(RUNSQLSTM SRCFILE(QDB2TX/QADESDDL) SRCMBR(DESMIGV5R2)
          NAMING(*SQL)
          ) JOB(DESMIGV5R2) USER(QDESADM)
```

これにより、変更された関数定義が更新され、SEARCHRESULT 関数の定義が、すでに使用可能なデータベース・サーバーに追加されます。

サーバーの設定および保守

第 3 章 はじめに

この章は、テキストを検索可能にするための基本的な点を理解するのに役立ちます。この章では、稼働中の DB2 テキスト・エクステンダー・システムで作業を行っていることを前提としています。つまり、システムのインストールと構成が済んでおり、DB2 テキスト・エクステンダー・インスタンスの作成と開始も済んでいるものとします。

ヒント

この章では、テキストを検索可能にするための基本的な点のみを解説します。自分のテキストを検索用に準備する前に、43 ページの『テキストを検索可能にする前の準備』をお読みください。

テキストを検索可能にする簡単な例

1. コマンド入力画面で次のように入力します。
`CALL(QDB2TX/DB2TX)`
2. サーバーをテキスト検索で使用できるようにする
接続されたサーバーを使用可能にするには、次のように入力します。
`db2tx=>ENABLE SERVER FOR DB2TEXT`
3. テキスト表をテキスト検索に使用できるようにする (オプション)
テキスト表を使用可能にする必要があるのは、表全体のために単一の索引を作成する場合のみです。

索引を 1 つにするか、それとも複数にするか

34 ページの『表に対する 1 つまたは複数のテキスト索引の作成』では、表を検索可能にするにあたり、表全体に 1 つのテキスト索引を作成する方法と、複数の索引 (各テキスト行に 1 つずつ) を作成する方法があることが説明されています。

- 表全体のために 1 つの索引を作成するには、この時点で `ENABLE TEXT TABLE` を実行して空の索引を作成してから、`ENABLE TEXT COLUMN` (次のステップを参照) を複数回 (テキスト列ごとに 1 回) 実行して、その 1 つの索引にデータを入れます。
- テキスト列ごとに別個の索引を作成する場合は、表全体のための索引を作成するステップを省略し (つまり `ENABLE TEXT TABLE` の実行を省略する)、`ENABLE TEXT COLUMN` を複数回実行することにより、テキスト列ごとに 1 つの索引を作成して、データを入れます。

この例では、表全体のために単一の索引を作成するのではなく、以下の手順に従って、テキスト列ごとに 1 つの索引を作成します。

はじめに

4. テキスト列をテキスト検索で使用できるようにする

以下のコマンドを入力して、DB2 テキスト・エクステンダーが表 `db2tx.sample` 内のテキスト列 `mycolumn` を検索できるようすると共に、このコマンドが作成するハンドル列に名前 `myhandle` を割り当てます。

```
db2tx=>ENABLE TEXT COLUMN db2tx.sample mycolumn HANDLE myhandle
```

このコマンドは、テキスト索引を作成します。索引付けされる文書のタイプおよび索引特性には、デフォルト値が使用されます。

5. 作成する索引の状況を調べる

次のように入力します。

```
db2tx=>GET INDEX STATUS db2tx.sample HANDLE myhandle
```

6. DB2 テキスト・エクステンダーのコマンド行プロセッサを終了する

次のように入力します。

```
db2tx=>QUIT
```

7. 対話式 SQL セッションを開始する

コマンド入力画面で次のように入力します。

```
STRSQL NAMING(*SQL)
```

8. テキストを検索する

これで、文書の検索が可能になりました。以下の `SELECT` コマンドを実行すると、索引付けしたテキスト内の `searchterm` のすべてのオカレンスが見つかります。

```
SELECT COUNT (*)  
FROM sample  
WHERE DB2TX.CONTAINS (myhandle,'"searchterm"') = 1
```

第 4 章 必要に応じた検索の計画

43 ページの『第 5 章 テキストを検索可能にする』で説明されているステップを開始する前に、以下の点を調べる必要があります。

- 使用する文書の形式とコード・ページ、および使用している言語
- コード・ページの問題を回避する方法
- 必要な検索機能の種類
- 必要なディスク・スペース
- テキスト索引とは何か、および 1 つの DB2 表に共通の索引を 1 つ設ける方法と、表列ごとに別個の索引を設ける方法のどちらを採用するか

この章では、上記の情報を収集する理由と方法を説明します。

索引のタイプは、基本語形、一致語句、および Ngram から選択します。どの索引タイプを選択するかは重要なことです。たとえば、索引のタイプとして基本語形を選択すると、検索項目の変化形や同義語を検索できます。また、索引のタイプによって索引付けのパフォーマンスや索引のサイズが異なってきます。1 つのテキスト列について、索引タイプの異なる索引をいくつか作成することにより、複数の索引タイプの検索機能を使用することもできます。

テキスト文書を索引付けする理由

高速情報検索システムでは、テキスト文書全体を順次走査するわけではありません。これでは非常に時間がかかります。そこで、あらかじめ作成したテキスト索引で操作を行います。テキスト索引は、テキスト文書から抽出された重要な用語から成るものと考えることができ、各用語は、その用語を含む文書に関する情報と一緒に保管されます。

テキスト索引には関係のある情報のみが含まれ、「and」、「of」、および「which」などの重要でない語は索引付けされません。(Ngram 索引では停止ワードのフィルター操作は行われません。) DB2 テキスト・エクステンダーは、停止ワードと呼ばれる語のリストを使用して、このような語が索引付けされないようにします。検索システムは、要求された用語を索引中で検索し、その用語を含んだテキスト文書を検出します。

ヒント

停止ワードのリストを変更する必要がある場合は、インストール時に一度だけ行うようにしてください。

停止ワードのリストは、ユーザーが変更可能なファイル内に言語ごとに保管されます (41 ページの『停止ワードと省略語ファイルの変更』を参照してください) が、ファイルはシステム全体で 1 つしかないため、変更は、DB2 テキスト・エクステンダーを最初にセットアップするときに一度のみ行うようにしてください。後でそのファイルを変更しても、既存の索引はその変更を反映しません。

テキスト文書を索引付けする理由

たとえば、「Now」という週刊誌名を含む文書があるものとして、この語を停止ワードから除去すると、これに索引が付けられ、後の検索で検出可能になります。ただし、停止ワードを除去する前に作成されたどの索引にも「now」という語は含まれず、その検索は失敗に終わります。

停止ワードを変更することにし、その変更をすべての索引に反映させたい場合は、全索引を作成し直す必要があります。

索引付けは 2 段階のプロセスで行われます。第 1 ステップでは、索引付けを行うテキスト文書がログ表に記録されます。これは、列中でテキスト文書の挿入、更新、または削除を行うたびに DB2 トリガーによって自動的に実行されます。

第 2 ステップでは、ログ表にリストされたテキスト文書を索引付けします。これは定期的に行うことができます。列中で挿入または変更が行われた文書の用語は、索引に追加されます。列から削除された文書の用語は、索引から除去されます。

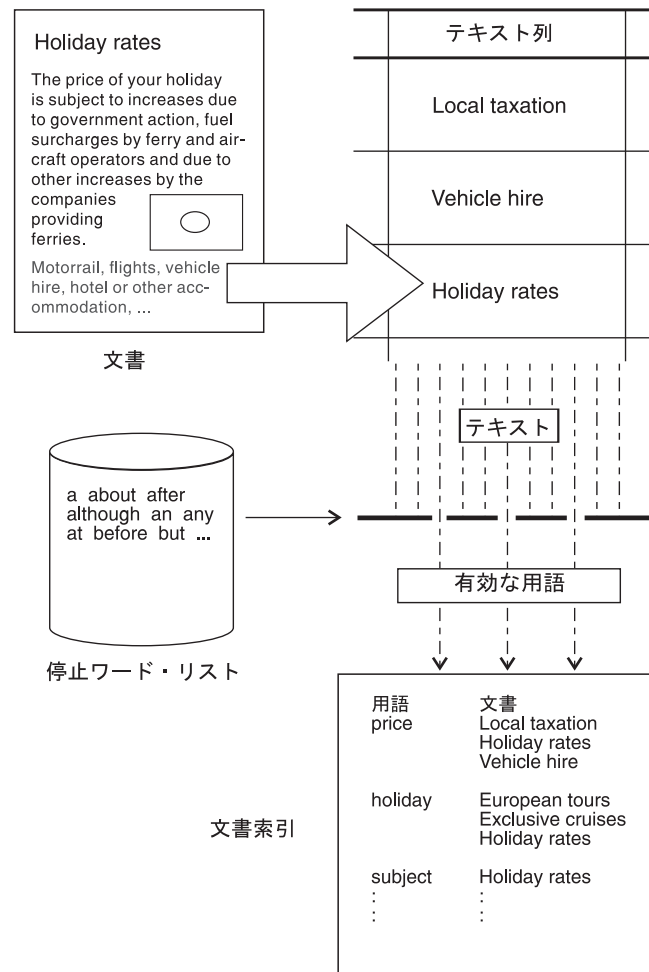


図 3. 重要な用語のみの索引付け

サポートされる文書形式

DB2 テキスト・エクステンダーは、検索する予定のテキスト文書の形式 (またはタイプ) を知る必要があります (HTML や ASCII など)。この情報は、テキスト文書の索引付けの際に必要です。

サポートされるテキスト文書形式は以下のとおりです。

HTML	ハイパーテキスト・マークアップ言語
XML	拡張マークアップ言語
ASCII_SECTIONS	セクションを含む構造化 ASCII
TDS	フラットな ASCII

次の文書形式は、後方互換性の理由によりサポートされています。

AMI	AmiPro アーキテクチャー バージョン 4
FFT	IBM 最終形式テキスト: 文書コンテンツ・アーキテクチャー
MSWORD	Microsoft Word、バージョン 5.0 および 5.5
RFT	IBM 変更可能テキスト: 文書コンテンツ・アーキテクチャー
RTF	Microsoft Rich Text Format (RTF)、バージョン 1
WP5	WordPerfect (OS/2 および Windows)、バージョン 5.0、5.1、および 5.2

HTML 文書に関する特別な考慮事項

HTML 文書のウムラウトや発音区別文字の扱いは、文書のコード・ページに依存しています。

- コード・ページが 37、273、277、278、280、284、297、437、500、819、850、858、860、863、865、871、923、924、および 1252 の場合、以下が適用されます。
 - ウムラウトや特殊文字を表す場合に、エンティティー表記が使用されます。たとえば、ä を表すためには ä が使用されます。
 - コード・ページ 819 (ASCII) または 500 (EBCDIC) でコード・ポイントを持つ文字のみが有効です。
 - 文書のコード・ページが 819 または 500 でない場合、ä のような言語固有の文字が含まれる文書では、索引付けの時に言語認識エラーが起きます。
 - 文書を Ngram 索引に追加する場合、索引は、コード・ページ 819、500、または UTF8 を使用して作成されたものでなければなりません。
- 他のすべての 1 バイト文字セットのコード・ページの場合、以下が適用されません。
 - エンティティーは解決されません。
 - 特殊文字は言語固有のコード・ポイントで書かなければなりません。

XML 文書に関する特別な考慮事項

XML 文書には、エンコード・ストリングが含まれている必要があります。これは構文解析の時に使用されます。エンコード・ストリングが含まれていない場合は、UTF8 がエンコードに使用されます。以下のエンコードがサポートされています。

```
UTF8, utf8, utf-8, ibm-1208, utf_8,
UTF16_BigEndian, utf-16be, utf16
UTF16_LittleEndian, utf-16le
LATIN_1, latin1, latin-1, ascii, ibm-819, iso-8859-1, Latin-1
ibm-912, iso-8859-2
ibm-913, iso-8859-3
ibm-914, iso-8859-4
ibm-915, iso-8859-5
ibm-1089, iso-8859-6
ibm-813, iso-8859-7
ibm-916, iso-8859-8i
ibm-920, iso-8859-9
ibm-0037, ebcdic-cp-us, ebcdic-cp-ca, ebcdic-cp-nl, ebcdic-cp-dk,
ebcdic-cp-no, ebcdic-cp-fi, ebcdic-cp-se, ebcdic-cp-it,
ebcdic-cp-es, ebcdic-cp-gb
ibm-297, ebcdic-cp-fr, ebcdic-cp-ar1, ebcdic-cp-he, ebcdic-cp-ch,
ebcdic-cp-roece, ebcdic-cp-yu, ebcdic-cp-is, ebcdic-cp-ar2
ibm-954, euc-jp eucjis
ibm-943, shift_jis, sjis, shiftjis, shift-jis
ibm-950, big-5, big5
ibm-949, iso2022kr, euc-kr
ibm-878, koi8-r
```

XML 文書の場合、デフォルトのコード・ページは無視されます。

XML 文書を Ngram 索引に追加する場合、索引は、コード・ページ UTF8 を使用して作成されたものでなければなりません。索引にセクションのサポートがある場合、索引付けの際、葉レベルの ID のみが使用されます。ネストされたセクション内のテキストの解決はサポートされていません。

サポートされていない文書形式の使用

サポートなしの文書形式には、数値 ID を指定してください。有効な値は、8192 ~ 65535 です。この値は、ユーザー出口からソース形式に渡され、オリジナル形式から TDS に変換します。

索引付けの実行時に、サポートされていないタイプの文書が見つかったら、DB2 テキスト・エクステンダーは出口を提供します。出口では、その文書をディスクに書き込み、ユーザーが提供したプログラムを呼び出して、テキストを抽出し、サポートされるいずれかの形式にします。ユーザー出口は、サーバー構成ファイル (imosrv.ini) およびクライアント構成ストリーム・ファイル (imocl.ini) の両方に登録する必要があります。

[DOCUMENTFORMAT] セクション内の USEREXIT オプションを、ユーザー出口の名前に更新します。

ユーザー出口を使用可能にするには、以下のストリーム・ファイルを編集します。

```
/QIBM/UserData/DB2Extenders/Text/instance/imocl.ini
/QIBM/UserData/DB2Extenders/Text/instance/txins000/imosrv.ini
```

以下のステートメントを追加します。

```
[DOCUMENTFORMAT]
USEREXIT=name_of_executable
```

ここで、<name_of_executable> はユーザー出口の名前です。完全修飾ファイル名を指定するか、または、ユーザー出口が PATH ステートメント内のディレクトリーに保管されている場合には、ファイル名のみ指定することもできます。

ユーザー出口を呼び出すには、以下の構文を使用します。

```
<name_of_user_exit> -sourcefile <sourcefilename>
                   -targetfile <targetfilename>
                   -sourceccsid <sourceccsid>
                   -targetccsid <targetccsid>
                   -sourceformat <sourceformat>
                   -targetformat <targetformat>
```

sourcefilename

ユーザー出口プログラムによって変換されるファイル。ファイル名は完全修飾名であり、クライアント・プロファイルまたはサーバー・インスタンスのいずれかで指定した作業ディレクトリーに置かれます。

targetfilename

ユーザー出口の出力を収めるファイル。この後、DB2 テキスト・エクステンダーはこのファイルを使って処理を実行します。ファイル名は完全修飾名であり、クライアント・プロファイルまたはサーバー・インスタンスのいずれかで指定した作業ディレクトリーを指します。クライアント・プロファイル内の項目は API 呼び出しの EhwGetMatches で、またサーバー・インスタンス内の項目は API 呼び出しの EhwUpdateIndex で使用されます。

sourceccsid

ソース・ファイルのコード・ページ。これはデフォルトのコード・ページです。

targetccsid

DB2 テキスト・エクステンダーが预期するコード・ページ。コード・ページは 500 です。

sourceformat

ソース・ファイルの形式。これはデフォルトの形式です。

targetformat

DB2 テキスト・エクステンダーが预期するファイルの形式。現行では、フラット・ファイル形式 (TDS) のみ、または、セクションが使用可能な索引の場合は ASCIISECTION のみがサポートされています。

ユーザー出口は以下の値を戻すことができなければなりません。

- 0 形式の変換が正常に行われたことを示します。
- >0 形式の変換が正常に行われなかったことを示します。索引付けの時、エラー・メッセージは文書のエラー表に書き出されます。エラー・メッセージを表示するには、desmsgix コマンドを使用します。

言語

DB2 テキスト・エクステンダーは、実行する言語処理に正しい辞書が使用されるように、文書がどの言語で書かれているかを知る必要もあります。40 ページの表 4 は、テキスト列または外部文書を使用可能にするとき、指定できる言語パラメーターのリストを示しています。

CCSID

ヒント

テキスト列を使用可能にする際に CCSID を指定する場合は、その前に 28 ページの『テキストの保管および使用可能化の際に起きるコード・ページに関する問題の回避』をお読みください。

CCSID が以下の表のいずれかである文書は、索引付けすることができます。

注: CCSID 861、865、および 4946 は、DB2 UDB for iSeries ではサポートされていません。これらの CCSID の文書を索引付けするには、バイナリー・データ・タイプ (BLOB または FOR BIT DATA) で文書を列に保管してください。

EBCDIC

37	米国、カナダ英語
273	オーストリア、ドイツ語
277	デンマーク語、ノルウェー語
278	フィンランド語、スウェーデン語
280	イタリア語
284	スペイン語 (ラテン・アメリカ)
285	英国英語
297	フランス語
420	アラビア語
424	ヘブライ語
500	国際ラテン -1
871	アイスランド語
875	ギリシャ語
1025	ロシア語
1112	ラトビア語
1122	エストニア語
1123	ウクライナ語
1130	ベトナム語

ASCII

437 OS/2	米国英語
813 AIX、HP、SUN	ギリシャ語
819 AIX、HP、SUN	ラテン -1
850 AIX、OS/2	ラテン -1
855 OS/2	ブルガリア語
860 OS/2	ポルトガル語
861 注を参照	アイスランド語
862 OS/2	ヘブライ語
863 OS/2	カナダ英語
864 OS/2	アラビア語
865 注を参照	デンマーク語、ノルウェー語
866 OS/2	ロシア語
869 OS/2	ギリシャ語
915 AIX、OS/2、HP	ロシア語
916 AIX	ヘブライ語
921 AIX、OS/2、WIN	ラトビア語
922 AIX、OS/2、WIN	エストニア語
1046 AIX	アラビア語
1089 AIX、HP	アラビア語
1124 AIX	ウクライナ語
1125 OS/2	ウクライナ語
1129 AIX	ベトナム語
1131	ベトナム語
1250 WIN	クロアチア語、ベロルシア語
1251 WIN	ロシア語
1252 WIN	ラテン -1
1253 WIN	チェコ語
1255 WIN	ヘブライ語
1256 WIN	アラビア語
1257 WIN	ギリシャ語
1258 WIN	ベトナム語

DBCS

932 AIX、OS/2	日本語、SBCS/DBCS の結合
942 OS/2	日本語、SBCS/DBCS の結合

CCSID

943 OS/2、WIN、AIX	日本語、SBCS/DBCS の結合
5039 HP	日本語、SBCS/DBCS の結合
954 AIX、HP、SUN	日本語
933 AS/400	韓国語
949 OS/2	韓国語
970 AIX、HP、SUN	韓国語
1363 WIN	韓国語
937 AS/400	中国語 (繁体字)、SBCS/DBCS の結合
948 OS/2	中国語 (繁体字)、SBCS/DBCS の結合
950 AIX、HP、OS/2、SUN、WIN	中国語 (繁体字)、SBCS/DBCS の結合
964 AIX、HP、SUN	中国語 (繁体字)、SBCS/DBCS の結合
1388 AS/400	中国語 (簡体字)、SBCS/DBCS の結合
1381 OS/2、WIN	中国語 (簡体字)、SBCS/DBCS の結合
1383 AIX、HP、SUN	中国語 (簡体字)、SBCS/DBCS の結合
1386 AIX、OS/2、WIN	中国語 (簡体字)、SBCS/DBCS の結合
4946 注を参照	ラテン -1 (CP850)
5035 AS/400	日本語、SBCS/DBCS の結合
5039 HP	日本語

テキストの保管および使用可能化の際に起きるコード・ページに関する問題の回避

以下の各項目にはコード・ページの設定値があります。

- アクティブ・アプリケーション環境
- 各文書
- 各 DB2 データベース
- 各 DB2 テキスト・エクステンダーの索引

VARCHAR や CLOB などの文字データ・タイプを持つ DB2 データベース列に文書を保管する場合、DB2 は、各文書のコード・ページはアクティブ・アプリケーション環境と同じであると想定して、文書のコード・ページをデータベースのコード・ページに変換します。データベースのコード・ページは、すでにアクティブ・アプリケーション環境のコード・ページと同じになっているか (この場合変換は行われません)、またはデータベースの作成時に指定した、アプリケーションとは異なるコード・ページ (この場合は変換が実行されます) のいずれかです。

BLOB や FOR BIT DATA などのバイナリー・データ・タイプを持つ DB2 データベースの列にデータを保管する場合、DB2 はデータを変換せず、文書は元の CCSID を保持します。

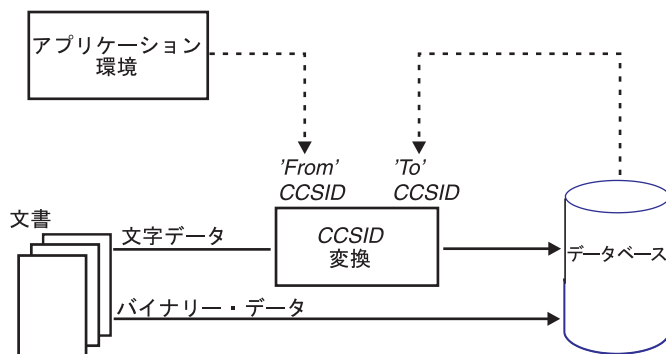


図4. DB2 がデータベース文書の CCSID を設定する方法

テキスト列を DB2 テキスト・エクステンダーが使用できるようにする、つまり ENABLE TEXT COLUMN コマンドを使用して検索用の索引を作成すると、索引のコード・ページは、データベースのコード・ページ (デフォルト) か、CHANGE TEXT CONFIGURATION コマンドを使用して設定できる現行のデフォルトか、または ENABLE TEXT COLUMN コマンドで指定したコード・ページのいずれかに設定されます。

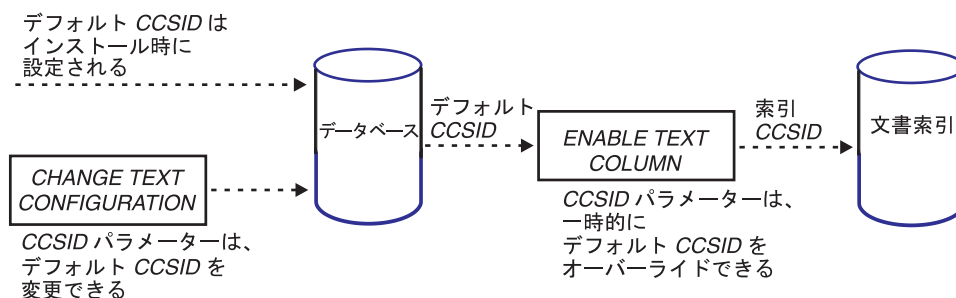


図5. DB2 テキスト・エクステンダーがテキスト索引の CCSID を設定する方法

検索の間に、データベースの CCSID を使用して検索ストリングの CCSID が解釈されます。

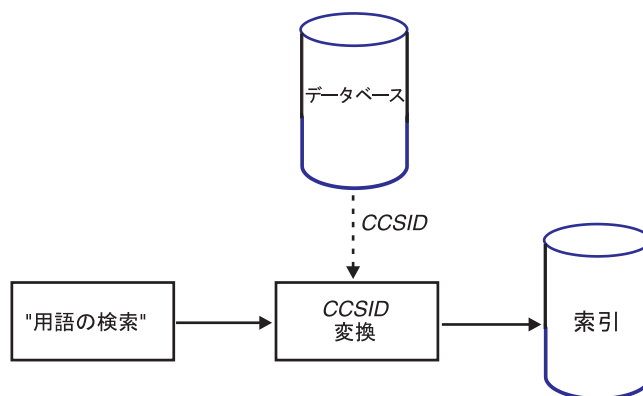


図6. 検索でデータベースの CCSID が使用される

以下はコード・ページの設定を調べる方法です。

- データベースのコード・ページを調べるには、以下の DB2 コマンドを使用します。

```
db2 get db cfg
```

- デフォルトの索引コード・ページを調べるには、以下の DB2 テキスト・エクステンダー・コマンドを使用します。

```
db2tx get text cfg
```

コード・ページの問題を回避するには、これらのコード・ページの設定を正確に調整することが重要です。

- **例 1:** 環境 850、文書 850、データベース 850、索引 850

正常。DB2 は、文書と環境のコード・ページが同じであると正しく判断します。データベースと環境のコード・ページが同じであるため、DB2 は変換を行わず、コード・ページ 850 で文書をデータベースに保管します。後でテキスト列を使用可能にする際、デフォルトのコード・ページの設定 (データベースのコード・ページである 850) を採用すると、文書は、850 のコード・ページの索引に正しく索引付けされます。

- **例 2:** 環境 1252、文書 1252、データベース 850、索引 850

正常。DB2 は文書と環境のコード・ページが同じであると正しく判断し、文書をデータベースへ保管する際に、1252 から 850 へのコード・ページ変換を行います。後でテキスト列を使用可能にする際、デフォルトのコード・ページの設定である 850 (データベースのコード・ページ) を採用すると、文書は 850 のコード・ページの索引に正しく索引付けされます。

- **例 3:** 環境 1252、文書 850、データベース 850、(索引は任意)

エラー。DB2 は、文書 (コード・ページ 850) と環境 (コード・ページ 1252) のコード・ページが同じである、と間違っただけで判断し、文書をデータベースに保管する際に、1252 から 850 への誤ったコード・ページ変換を行います。

- **例 4:** 環境 1252、文書 850、データベース 1252、索引 850

正常。DB2 は、文書と環境のコード・ページが同じであると間違っただけで判断をしますが、環境とデータベースのコード・ページが同じであるため、DB2 は変換を行わず、コード・ページ 850 で文書をデータベースに保管します。ただし、後にテキスト列を使用可能にする際、文書のコード・ページを 850 に指定して、文書が 850 のコード・ページの索引に正しく索引付けされるようにしなければなりません。

- **例 5:** 環境 1252、文書 1252、データベース 850、索引 1252

エラーになる可能性あり。DB2 は、文書と環境のコード・ページが同じ 1252 であると正しく判断し、文書をデータベースに保管する際に、そのコード・ページを 850 に変換します。コード・ページ 850 の索引に対してテキスト列を使用可能にする際に、文書の元のコード・ページである 1252 を指定すると、エラーが起きる可能性があります。デフォルトのコード・ページ設定の 850 (データベースのコード・ページ) を採用する、ということが正しいアクションとなります。

検索のタイプ

検索するテキストを含む列に対して、以下のいずれかの検索タイプおよびさまざまなオプションを割り当てることができます。すなわち、基本語形、一致語句、および *Ngram* です。このような列を作成して DB2 テキスト・エクステンダーで使用するには、その前にまずどの索引タイプを作成するかを決めておかなければなりません。各索引タイプが言語処理にどのような影響を与えるかについて詳しくは 197 ページの『第 15 章 基本語形索引と一致語句索引の言語的な処理』を参照してください。

要約

- **基本語形の変化形を検索する場合は、基本語形索引を使用します。**
語の変化形を、正規化と語幹処理、および辞書の使用に基づいて検索します。ディスク・スペースが一番少なく済みます。
- **完全一致検索を行うを行う場合は、一致語句索引を使用します。**
入力した通りの用語を検索します。索引付けと検索がより速く行えます。より多くのディスク・スペースを必要とします。正規化する場合、検索は大文字小文字を区別しません。
- **文字の変化形を検索する場合は、Ngram 索引を使用します。**
つづりの間違っている語も検出します。CASE_ENABLED を指定して大文字小文字の区別をする検索を行う場合、索引はより多くの容量を使用し、検索により時間がかかることがあります。
- **DBCS 文書の検索を行う場合、Ngram 索引を使用します。**
DBCS 文書の唯一の選択です。ただし、タイプ TDS の SBCS 文書でも使用することができます。

DB2 テキスト・エクステンダーは、多様な検索オプションを備えていますが、そのすべてをすべての索引タイプで使用できるわけではありません。どの索引タイプを使用するかを決める前に、171 ページの表 7 および 171 ページの表 8 を参照してください。

注

すべての索引タイプが、テキスト・エクステンダーがサポートしている言語に使用できるわけではありません。詳しくは、40 ページの『辞書、停止ワード・リスト、省略後リスト、および言語パラメーター』の表を参照してください。

基本語形検索

基本語形索引の場合、各文書テキストを分析して索引付けを行う際に、言語的な処理が適用されます。つまり、語は基本形に直されてから索引に保管されます。たとえば、「mice」は mouse として索引に保管されます。

検索のタイプ

基本語形索引に対して照会を行う場合も、これと同じ言語的な処理が検索項目に適用された後にテキスト索引が検索されます。このため「mice」を検索する場合は、基本形の mouse に直されてから検索が開始されます。197 ページの表 17 に、基本語形索引を用いた索引付けの場合に、用語がどのように抽出されるのかを要約しています。

この索引タイプの利点は、検索項目としてどのような変化形を指定しても、索引付けされたテキスト文書のいずれかに出てくる他の変化形と一致するということです。mouse という検索項目は、文書用語の「mouse」、「mice」、「MICE」(大文字) などと一致します。同様に Mice という検索項目も、同じ文書用語と一致します。

この索引タイプは、最もディスク容量が少なくて済みます。ただし、索引付けと検索にかかる時間が一致語句索引の場合より長くなります。

使用できる言語的な処理のタイプは文書の言語によって異なります。以下にタイプをリストします。

- 語および文の分割
- 文の開始処理
- ハイフンの除去
- 用語の標準形への正規化。これには大文字は使われず、「ü」などのアクセント記号がついた文字はアクセント記号のない書式に変更されます。たとえば、ドイツ語の「Tür」(ドア) は、tuer として索引付けされます。
- 用語の基本形への変形。たとえば、「bought」は buy、「mice」は mouse として索引付けされます。

ヒント

語の断片 (ワイルドカード文字でマスクされた語) は、基本形に変形できません。したがって、swu% を検索すると、「swum」という語は索引の中で基本形に変形されるので検出されません。この語を検出するには、swi% により検索しなければなりません。

つづりが正確でない語の変化形は、基本形に直すことができません。ドイツ語の röstete を例に考えてみましょう。この語は rösten という動詞の基本形に正しく索引付けされます。検索項目 röstete または rösteten は、基本形の rösten に正しく正規化され、この用語が索引で見つかります。rostete という検索項目では、rösten に正規化することができず、この用語は索引で見つかりません。

- 語の分解。これによりドイツ語の「Wetterbericht」(天気予報) などの複合語は、wetterbericht だけでなく wetter および bericht としても索引付けされます。
- 停止ワードのフィルター操作。これは、関係のない用語を索引付けしません。「A report about all animals」は、report および animal として索引付けされません。
- 品詞のフィルター操作。これは停止ワードのフィルター操作に似ており、名詞、動詞、形容詞のみが索引付けされます。「I drive my car quickly」は、drive お

よび car として索引付けされます。停止ワードとして「I」と「my」の語が除去され、さらに副詞「quickly」が品詞のフィルター操作によって除去されます。

一致語句検索

一致語句索引では、テキスト文書内の用語は文書に出てくるとおりに索引付けされます。たとえば、mouse という検索項目で「mouse」を検出できますが、「mice」や「Mouse」は検出できません。一致語句索引での検索では大文字と小文字が区別されます。

照会でも同じ処理が照会用語に適用され、照会用語は索引中の検出される用語と比較されます。すなわち、検出される用語は検索用語と全く同じになります。マスキング文字を使用すれば検索範囲を広げることができます。たとえば、experiment% という検索項目で「experimental」や「experimented」などを検索できます。

198 ページの表 18 に、一致語句索引を使用する索引付けの場合に、どのように用語が文書テキストから抽出されるか、いくつか例を挙げてあります。

この索引タイプの利点は、検索が正確であることと、索引付けと検索の速度が速いことです。あらゆる用語の変化形ごと、つまり、つづりが違うとそのたびに索引付けされるため、基本語形索引の場合よりも多くのディスク・スペースが必要です。

一致語句索引の場合に索引のテキスト文書に使用される言語的な処理は、以下のとおりです。

語および文の分割

停止ワードのフィルター操作

ファジー検索または DBCS 文書の検索

Ngram 索引は、文字セットを解析することによって、テキストを分析します。この分析は辞書に基づいて行われるわけではありません。

テキストに DBCS 文字が含まれている場合は、Ngram 索引を使用する必要があります。これ以外の索引タイプは、DBCS 文字をサポートしません。

この索引タイプは、「ファジー」検索をサポートします。すなわち、指定した検索項目に類似した文字ストリングを検出することができます。たとえば、Extender を検索すると、つづりを誤った語 Extendrrs も検出されます。また、要求する類似性のレベルを指定することもできます。

注: ファジー検索を使用する場合でも、先頭の 3 文字は一致していなければなりません。

Ngram 索引で大文字小文字の区別をする検索を行うためには、その照会で PRECISE FORM OF キーワードを指定するのみでは十分ではありません。これは、Ngram 索引では通常、索引付き文字が大文字か小文字かを区別しないためです。しかし、Ngram 索引の作成時に、CASE_ENABLED オプションを指定することによって、その索引で大文字小文字が区別されるようにすることができます。さらに、照会時に、PRECISE FORM OF キーワードを指定します。

検索のタイプ

CASE_ENABLED オプションを使用すると、索引用に必要なスペースが増え、検索により時間がかかることがあります。

Ngram 索引でサポートされる CCSID のリストについては、26 ページの『CCSID』を参照してください。Ngram 索引は、ネイティブの CCSID のリストをサポートします。他のすべての CCSID の場合、データはこの CCSID から UTF8 にマップされます。

Ngram 索引タイプは DBCS 文書の索引付けのために使用するよう設計されていますが、SBCS 文書に対して使用することもできます。

また、すべての構文検索オプションがサポートされるわけではないことにも注意してください。165 ページの『第 12 章 検索引き数の構文』に記載されている規則と制限の要約を参照してください。

索引タイプの変更

使用中の索引タイプが適切でないと判断したならば、まずテキスト列またはテキスト表を使用不可にして索引を削除し、その後テキスト列またはテキスト表を再び使用可能にして索引を再作成します。

表に対する 1 つまたは複数のテキスト索引の作成

43 ページの『第 5 章 テキストを検索可能にする』では、テキストを検索するための表の作成方法について説明します。しかしその前に、表中のすべての索引付きテキスト列に対して共通なテキスト索引を 1 つ作成するか、または索引付きテキスト列ごとに 1 つずつ、つまり複数のテキスト索引を作成するのかを決めておかなければなりません。テキスト列ごとに別個の索引を持つ表は、複数索引表と呼ばれます。

ヒント

外部ファイルを索引付けする予定である場合 (53 ページの『外部テキスト・ファイルの使用可能化』を参照)、関連する表は複数索引表でなければなりません。

複数索引を使用すると以下の利点があります。

- テキスト列ごとに異なる索引タイプを作成できる
テキスト列の索引を更新する周期や索引の保管先のディレクトリーなどの、テキスト列に関連した特性に柔軟性が増します。これらの特性については 113 ページの『ENABLE TEXT COLUMN』を参照してください。
- 列の索引付けを別々に行える
索引付けの活動に多くの時間とリソースを費やす場合があります。複数索引表を作成すると、列の索引付けをさまざまな時点で行うため、この活動にある程度余計に時間がかかることがあります。

複数索引表を作成して柔軟性を高める必要がない場合は、共通索引を使用して DB2 テキスト・エクステンダーでの保守を簡単にすることができます。テキスト表を使

表に対する 1 つまたは複数のテキスト索引の作成

用可能にするときに設定する索引パラメーターが、そのテキスト列すべてに対するデフォルト値として使用されます。また、列を使用不可にする必要がある場合も、テキスト表を使用不可にするためのコマンドを 1 つ使用するのみで済みます。

索引サイズの計算

索引に必要なディスク・スペースは、索引付けされるデータのサイズおよびタイプと索引タイプによって決まります。ワード・プロセッサによって作成されたテキスト文書は、多くの内容に制御文字を含んでいるため、スペースは少なく済みます。目安としては、基本語形索引の場合、索引付けされる文書のサイズの 0.7 倍のディスク・スペースを予約し、さらにこの 2 倍のスペースを、索引を再編成するための一時スペースとして予約します。Ngram 索引の場合は、ディスク・スペースの約 2 倍が必要になります。

大量のスペースを要する索引がいくつかあり、索引の更新または検索中にそれらの索引に同時にアクセスする場合には特に、それらを別個のディスク装置に格納する必要があります。

索引の更新

データベースにテキスト文書を追加するか、またはデータベース内の既存の文書を変更する場合は、その文書を索引付けして、索引の内容とデータベースの内容を同期化させておく必要があります。テキスト文書をデータベースから削除する場合は、その用語を索引から除去する必要があります。

文書が新規のものか、変更を加えられているか、削除されたかに関する情報は、トリガーによって、自動的にログ表に保管されます。ログ表にリストされた文書は、次に索引が更新されたときに索引付けされます。

UPDATE INDEX コマンドを使用すると、要求した時すぐに索引を更新できます。

しかし一般には、環境変数 DB2TXUPDATEFREQ に指定した間隔で索引を自動的に更新するのが普通です。この環境変数にはデフォルトの設定が指定されています。デフォルトの設定は、ENABLE TEXT COLUMN または ENABLE TEXT TABLE コマンドを使用して索引を作成すると一時変更することができます。既存の索引に関する更新頻度は、CHANGE INDEX SETTINGS コマンドを使用して変更することができます。

索引更新の頻度は、更新を行う時点と、待ち行列に入れなければならないテキスト文書の最小数とに基づいて指定します。指定された日時にログ表内に十分な数の文書がない場合、索引は更新されません。

テキスト文書の索引付けは時間がかかり、リソースも多く必要とするタスクであるため、周期的な索引付けを慎重に計画する必要があります。索引付けにかかる時間は、前回の索引更新以後追加または変更されたテキスト文書の数、文書のサイズ、およびプロセッサの処理能力など多くの要因により異なります。

構造化文書 (セクション・サポート) での作業

セクション・サポートにより、構造化された文書の中の特定のセクション (表題、著者、説明など) に索引を付けて、検索することができます。使用できる文書形式は XML または HTML 形式、または HTML に似たタグを使用したフラット・ファイル文書です。マークアップのタグとそれに対応するセクション名は、文書モデルに定義します。文書モデルには、文書内のどのセクションに索引を付けて検索できるようにするかを定義します。セクション名は、そのセクションに対する照会で使用される記述的な名前です。

セクション・サポートを使用可能にするには、文書を含むテキスト列を使用可能にする際に INDEXPROPERTY SECTIONS_ENABLED を指定する必要があります。

文書モデル・ファイル には、サーバー・インスタンスの定義済み文書モデルがすべてリストされています。サンプル文書モデル・ファイル imomodel.ini は、ディレクトリー /QIBM/UserData/DB2Extenders/Text/instance/txins000 に配置されています。ファイルのコード・ページは EBCDIC です。

文書モデルの情報は、索引ディレクトリーにコピーされます。索引を作成した後にサーバー・インスタンスの文書モデルを変更しても、作成した索引のセクション・サポートには影響ありません。

セクション・サポート付きの索引を検索する (たとえば、著者のセクションから McDaniel を検索する) 方法は以下のとおりです。セクション (この場合は Author) の前には必ずモデル名が付きます。

```
select count (*)
from db2tx.htmltable
where db2tx.contains(myhandle,'MODEL myhtmlmodel SECTION
                    (author) "Schmidt") = 1
```

フラット・ファイルおよび HTML 文書

フラット・ファイルの場合、セクションは、<title> や <subject> などの HTML に似たタグを使用してマークアップされます。以下は、マークアップされたセクションを含む文書の例です。

```
<title> IBM Dictionary of Computing
<author> McDaniel, George
<subject> Computers, Reference, ....
```

以下は、フラット・ファイルまたは HTML 文書用の文書モデル・ファイルの例です。モデル名およびセクション名に含めることができるのは、A-Z、a-z、および 0-9 のみです。モデル名には常に大文字小文字の区別があります。セクション名は、大文字小文字を区別できるようにすることもしないようにすることもできます。モデルの作成時に設定を指定します。

```
;list of document models
;model always starts with 'modelname' and the name of the model
[MODELS]
modelname=sample
modelname=sample2
modelname=sample3

; a 'sample' document model definition
; left - logical section name identifier used for searching in a document
; right - section name tag
```

構造化文書 (セクション・サポート)

```
[sample]
Title = title
Author = author
Subject = subject
Abstract = abstract
Content = content
Date=publishingdate
```

```
[sample2]
Title = title
Author = author
Subject = subject
```

```
[sample3]
Title = title
Author = author
Abstract = abstract
Docnum = docnum
```

文書中に現れるマークアップされたセクションが、文書モデルで定義されていない場合、そのセクションの内容は、定義されている直前のセクションに含まれて、索引付けと検索がなされます。たとえば、ある文書に以下のマークアップされたセクションが含まれているとします。

```
<title> IBM Dictionary of Computing
<subject> Computers, Reference, ....
<author> McDaniel, George
<abstract> Contains up-to-the-minute coverage of information processing
systems, communication products and facilities, personal computers, and office
systems, as well as the full range of IBM hardware and software products.
```

また、文書モデル `book` が以下のように定義されているとします。

```
[MODELS]
modelName=book
[book]
Title = title
Author = author
Abstract = abstract
```

`book` 文書モデルには、`<subject>` セクションが含まれていません。この文書が索引付けされる際、`subject` セクションの内容は `title` セクションの内容と一緒に索引付けされることとなります。`subject` セクションの内容が、`title` セクション内の検索でも使用可能になります。

索引の作成時にモデルのリストを指定する場合、デフォルトのモデルになるのはリストの先頭のモデルです。`desmodix` コマンドを使用して、デフォルトのモデルを変更することができます。

XML 文書

セクション処理の可能な索引を使用する場合、XML 文書は正しく構造化されていなければならない、ルート・エレメントが含まれていなければなりません。ルート・エレメントの名前は、定義済みのモデル名のいずれかと同じでなければなりません。大文字小文字も一致していなければなりません。文書モデル・ファイル内のモデルの記述は、文書の DTD (文書タイプ定義) ファイルで定義された文書モデルのサブセットとなる必要があります。

モデルの記述はルート・エレメントから始める必要があります。セクションとして使用したい XML エレメントごとに、その完全な階層をモデルの記述に含める必要

があります。セクションのタイプが日付の場合、このセクションは文書モデル・ツリーの中で、葉となる必要があります。属性セクションのネストはサポートされていません。

以下は、XML 文書のモデルの説明の例です。

```
; list of document models
[MODELS]
modelname = LETTER
; sample for XML model definition
; left-hand side = logical section name identifier used for searching
; right-hand side = section name tags specifying the tags for each
;
;           element of the path through the tree down to
;           the specified node. Tag delimiter is /.
[LETTER]
LETTER = LETTER
LETTER/date = LETTER/DATE
LETTER/address = LETTER/ADDRESS
LETTER/address/City = LETTER/ADDRESS/CITY
LETTER/Content = LETTER/CONTENT
LETTER/Content/Greetings = LETTER/CONTENT/GREETINGS
```

注

論理セクション名には、タグ区切り文字を使用しない論理名か、または完全なパスを使用する論理名のいずれかを使用することができます。例:

```
LETTER = LETTER
date = LETTER/DATE
address = LETTER/ADDRESS
```

以下は、XML 文書の例です。モデルで定義されないセクションがどのように索引付けされるかも示されています。

```
<?xml version="1.0"?>
<!DOCTYPE LETTER SYSTEM "letter.dtd">

<LETTER>
  <HEADER>This tag has been skipped in the definition, to this text will
           be added to the section named LETTER
  </HEADER>
  <DATE>
           01.01.2000  03.02.2000
  </DATE>
  <ADDRESS>
           Text will be added to the section named LETTER/address.
  <CITY>
           Text will be added to section named LETTER/address/City.
  </CITY>
  </ADDRESS>
  <CONTENT>
           Text will be added to the section named LETTER/Content.

  <NOSECTION>Text will be added to the section named LETTER/Content
           because NOSECTION is not defined.
  </NOSECTION>
  <GREETINGS>
           Text will be added to section named LETTER/Content/Greetings.
  </GREETINGS>
  </CONTENT>
</LETTER>
```

辞書、停止ワード・リスト、省略後リスト、および言語パラメーター

表4 は、サポートされる言語と、辞書、停止ワード・リスト、および省略語リストとして提供されるファイルの名前を示しています。辞書ファイルは 2 進数形式で、変更できません。停止ワード・ファイルおよび省略語ファイル (もしあれば) は、フラット・ファイル形式で、変更できます。これらのファイルのいずれかを変更する場合には、必ずその言語のコード・ページを使用してください。

この表は、また、テキスト列または外部文書を使用可能にするときに指定する必要がある言語パラメーターも示しています。言語パラメーターは、文書がどの言語で書かれているかを DB2 テキスト・エクステンダーに知らせ、こうして言語処理が必要になったときに正しい辞書が使用されるようになります。

表4. さまざまな言語で使用される言語機能

言語	ファイル名	言語パラメーター	コード・ページ
アラビア語	arabic	ARABIC	420
ブラジル・ポルトガル語	brazil	BRAZILIAN	500
カナダ・フランス語	canadien	CAN_FRENCH	500
カタロニア語	catala	CATALAN	500
デンマーク語	dansk	DANISH	500
オランダ語	nederlnd	DUTCH	500
フィンランド語	suomi	FINNISH	500
フランス語	francais	FRENCH	500
ドイツ語	deutsch	GERMAN	500
ヘブライ語	hebrew	HEBREW	424
アイスランド語	islensk	ICELANDIC	500
イタリア語	italiano	ITALIAN	500
ノルウェー語 (ブークモール)	norbook	BM_NORWEGIAN	500
ノルウェー語 (ニーノシュク)	norntn	NN_NORWEGIAN	500
ポルトガル語	portugal	PORTUGUESE	500
ロシア語	russian	RUSSIAN	1025
スペイン語	espana	SPANISH	500
スウェーデン語	svensk	SWEDISH	500
スイス・ドイツ語	dschweiz	SWISS_GERMAN	500
英国英語	uk	UK_ENGLISH	500
米国英語	us	US_ENGLISH	500

(1) 停止ワード・ファイルのファイル名 (拡張子 STW) および省略語ファイル (拡張子 ABR)

(2) 言語的な処理では、古ドイツ語および新ドイツ語の両方を使用します。

ファイルは、拡張子によって識別されます。

内容	拡張子
辞書	DIC

辞書、停止ワード・リスト、および省略語リスト

内容	拡張子
停止ワード・リスト	STW
省略語リスト	ABR

DB2 テキスト・エクステンダーは、DBCS 言語の基本語形サポートを提供していません。下記の表には、これらの言語に対する適切な言語パラメーターがリストされています。

言語	言語パラメーター
中国語 (簡体字)	S_CHINESE
中国語 (繁体字)	T_CHINESE
日本語	JAPANESE
韓国語	KOREAN

停止ワードと省略語ファイルの変更

言語ごとに、1 つの停止ワード・ファイルと省略語ファイルが存在します。これらのファイルを編集する意味を知るためには 21 ページの『テキスト文書を索引付けする理由』を参照してください。

ヒント

これらのファイルのいずれかの編集を始める前に、バックアップ・コピーをとってください。

停止ワード・ファイルおよび省略語ファイルは、以下に格納されています。

/QIBM/ProData/imo/dict

使用しているエディターで、これらのファイルを編集します。これらは CCSID 850 を使用しているため、編集を始める前に、ユーザーのアプリケーションの CCSID も必ず 850 に設定します。

索引付けする語と省略語を除去します。索引付けしない語を加えます。

第 5 章 テキストを検索可能にする

19 ページの『第 3 章 はじめに』を読むと、DB2 テキスト・エクステンダーによってテキストの検索を可能にすることに慣れ親しむことができます。その章では簡単な例を使って説明しました。この章では、テキストを検索可能にするについて、より詳細に説明しています。また、開始する前に考慮すべき局面すべてが説明されています。

テキストを検索可能にするためのステップは以下のとおりです。

1. 綿密に準備する
2. DB2 テキスト・エクステンダー・コマンド行プロセッサを開始する
3. サーバーに接続する
4. データベースをテキスト検索で使えるようにする
5. テキスト表をテキストの検索で使えるようにする (テキスト列につき 1 つの索引を作成している場合は必要ありません)
6. テキスト列をテキストの検索で使えるようにする

テキストを検索可能にする前の準備

ヒント

このセクションを注意深くお読みください。テキストを検索可能にする前に知っておく必要のある選択事項がリストされています。

- **表全体に 1 つの索引を作成するか？**

テキスト表全体について 1 つの索引を作成するか、それともテキスト列ごとに別々の索引を作成するかを決める必要があります。34 ページの『表に対する 1 つまたは複数のテキスト索引の作成』は決定に役立ちます。

- **文書について理解する**

文書を検索可能にするときに、その CCSID、言語、およびテキストの形式を指定する必要があります。詳細については、21 ページの『第 4 章 必要に応じた検索の計画』を参照してください。

- **必要なテキスト索引のタイプを決める**

必要とする索引のタイプは、行いたい検索の種類 (一致語句、ファジーなど)、および文書が SBCS か DBCS によって決まります。21 ページの『第 4 章 必要に応じた検索の計画』により多くの情報が記されています。

- **索引を保管する場所を決める**

文書を検索可能にすると、DB2 テキスト・エクステンダーはテキスト索引を作成します。索引を保管するディレクトリーを指定する必要があります。十分なディスク・スペースがあることを確認してください (35 ページの『索引サイズの計算』を参照してください)。

テキストを検索可能にする前の準備

- **テキスト構成をセットアップする**

テキスト構成は、索引 CCSID のデフォルトの設定、文書の言語、文書の形式、索引タイプ、索引の更新の頻度、表スペース名、および索引ディレクトリーを決定します。

テキストを使用可能にする際にこれらの設定を変更することができますが、前もってデフォルトに正しく設定しておくにより便利です。DB2 テキスト・エクステンダーがインストールされたときのこの表の初期テキスト構成設定値については、10 ページの『テキスト構成設定値』で説明しています。インストールの設定を変更して、独自のデフォルト値をセットアップするには、103 ページの『CHANGE TEXT CONFIGURATION』を利用してください。

- **セクション・サポートをセットアップする**

検索をドキュメントの特定のセクション内でのみ行う必要がある場合、37 ページの『構造化文書 (セクション・サポート) での作業』を読み、文書モデル・ファイル内のモデルを指定する方法を学習してください。

- **停止ワードおよび省略語リストを変更する**

21 ページの『テキスト文書を索引付けする理由』および 41 ページの『停止ワードと省略語ファイルの変更』を読み、停止ワード・リストと省略語リストの概念を理解して、索引付けをする前に、それらを変更するかどうかを決めます。

43 ページの『テキストを検索可能にする前の準備』の説明にしたがって情報を収集し、決定を行うと、テキストを検索可能にする準備が整ったことになります。

DB2 テキスト・エクステンダー・コマンド行プロセッサの開始

要約	
実行時	オプション。個々のセッションの開始時
コマンド	CALL PGM(QDB2TX/DB2TX)
権限	すべて

CALL コマンドの PARM 部分にコマンド・パラメーターを渡すことによってコマンドを 1 つのみ実行する場合は、コマンド行プロセッサは、そのコマンドを完了した後終了します。コマンド入力画面から出すコマンドの例を以下に示します。

```
CALL PGM(QDB2TX/DB2TX) PARM('enable server for db2text')
```

コマンド行プロセッサは、下記のように何のパラメーターも渡さずに開始すると、アクティブのままとなります。

```
CALL PGM(QDB2TX/DB2TX)
```

このモードを終了するには、以下を入力します。

```
QUIT
```


注

SQL 表名は、SQL の命名規則 (集合名.表名) か、またはシステムの命名規則 (ライブラリー名/ファイル名) のいずれかを使用して指定することができます。

コマンド行プロセッサのヘルプ

コマンドのリストを表示するには次のように入力します。

```
CALL PGM(QDB2TX/DB2TX) PARM('?')
```

個々のコマンドの構文を表示するには次のように入力します。

```
CALL PGM(QDB2TX/DB2TX) PARM('? command')
```

例:

```
CALL PGM(QDB2TX/DB2TX) PARM('? CHANGE TEXT CONFIGURATION')
```

データベースの使用可能化

要約

実行時	検索するテキスト列が含まれたデータベースごとに 1 度。
コマンド	CALL PGM(QDB2TX/DB2TX) PARM('ENABLE SERVER FOR DB2TEXT')
権限	ユーザー・プロファイル QDESADM

このコマンドには、他のパラメーターはありません。このコマンドは、接続しているデータベースを DB2 テキスト・エクステンダーで使用するよう準備します。暗黙接続は、現行のアクティブなネーム・スペースに関連付けられているデータベースを常に使用します。

システム/ユーザー ASP 上のデータベースに加え、1 つまたは複数の独立 ASP 上にもデータベースがある場合、使用可能とすることのできるデータベースは制限されます。システム/ユーザー ASP 上のデータベース、あるいは独立 ASP 上の 1 つまたは複数のデータベースのいずれかを使用可能にできますが、これら両方を使用可能にはできません。

使用可能なテキスト列を追跡する、TEXTINDEXES というカタログ・ビューが作成されます。86 ページの『DB2 テキスト・エクステンダー・カタログ・ビューの処理』を参照してください。

このコマンドは、データベースのテキスト構成情報を作成します。これには、索引、テキスト、および処理特性のデフォルト値が含まれます。これらについては、10 ページの『テキスト構成設定値』で説明しています。

データベースの使用可能化

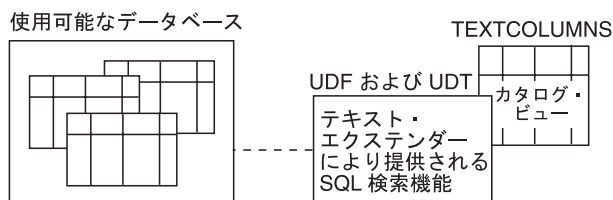


図7. データベースの使用可能化

データベースはいったん使用可能になると、使用不可にするまでは使用可能のままとなります。

テキスト表の使用可能化 (オプション)

要約	
実行時	オプション。表中のすべてのテキスト列に関する共通索引を作成する場合のみこのコマンドを使用します。34 ページの『表に対する 1 つまたは複数のテキスト索引の作成』を参照してください。
コマンド	CALL PGM(QDB2TX/DB2TX) PARM('ENABLE TEXT TABLE ...' (例を参照))
権限	ユーザー・プロファイル QDESADM

このステップでは、表中のすべてのテキスト列について共通索引を 1 つ作成するか、またはテキスト列ごとに別々の索引を付けて複数の索引を作成するのかを決めます。

共通索引を作成する場合は、まず ENABLE TEXT TABLE を実行し、次にテキスト列ごとに ENABLE TEXT COLUMN を実行します。別々の索引を作成する場合は、ENABLE TEXT TABLE はスキップし、テキスト列ごとに ENABLE TEXT COLUMN のみを実行します。これについては、図 8 と 48 ページの図 9 に記載してあります。

このステップの間に DB2 テキスト・エクステンダーにより空のテキスト索引が作成されますが、これは後で使用可能にするすべてのテキスト列に共通なものです。索引のタイプ、索引を更新する周期、および索引を保管するディレクトリーを指定します。指定しないパラメーターのデフォルト値は、テキスト構成設定値から取得されます。

ヒント	ほとんどのテキスト列について、索引タイプなどの設定を同じにする場合は、テキスト構成情報を使用してデフォルト設定を指定してください。10 ページの『テキスト構成設定値』を参照してください。
-----	---

テキスト表の使用可能化 (オプション)

またこのステップでは、表中の追加、変更、または削除が行われた文書を記録するための空のログ表も作成されます。更新されたログ表を保持するためにトリガーが作成されます。

ENABLE TEXT TABLE は、DB2 テキスト・エクステンダー用に使用可能にしたテキスト列がすでに含まれている表には実行できません。

ENABLE TEXT TABLE によって作成された索引を削除するには、89 ページの『テキスト表を使用不可にする』を参照してください。

ヒント

使用可能なテキスト表を除去する場合には、最初にこれを使用不可にして、索引、ログ表などを除去するようにならなければなりません。

例

次の例を実行すると、DB2TX.MYTABLE テキスト表が使用可能になります。

```
CALL PGM(QDB2TX/DB2TX) PARM('ENABLE TEXT TABLE db2tx.mytable')
```

索引特性のデフォルト値は、テキスト構成設定値から得られます。

次の例では、テキスト表用に作成される共通索引の特性が明示的に設定されます。

```
CALL PGM(QDB2TX/DB2TX) PARM('ENABLE TEXT TABLE db2tx.mytable
                               INDEXTYPE linguistic
                               DIRECTORY QIBM/UserData/DB2Extenders/Text/indexes')
```

テキスト表の使用可能化 (オプション)

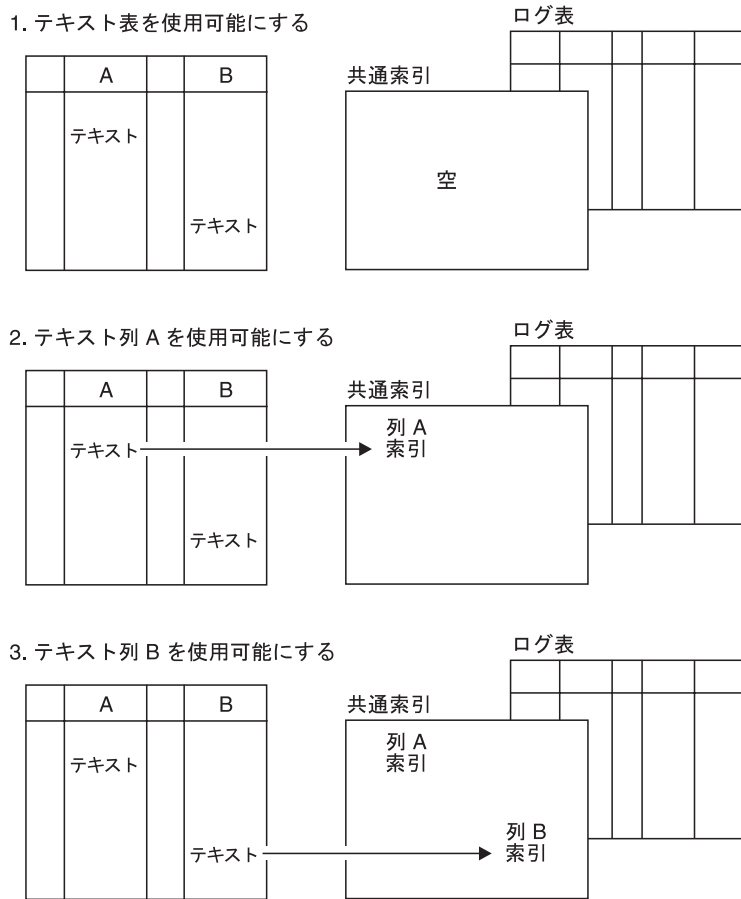


図 8. 表中のすべてのテキスト列に関する共通索引の作成

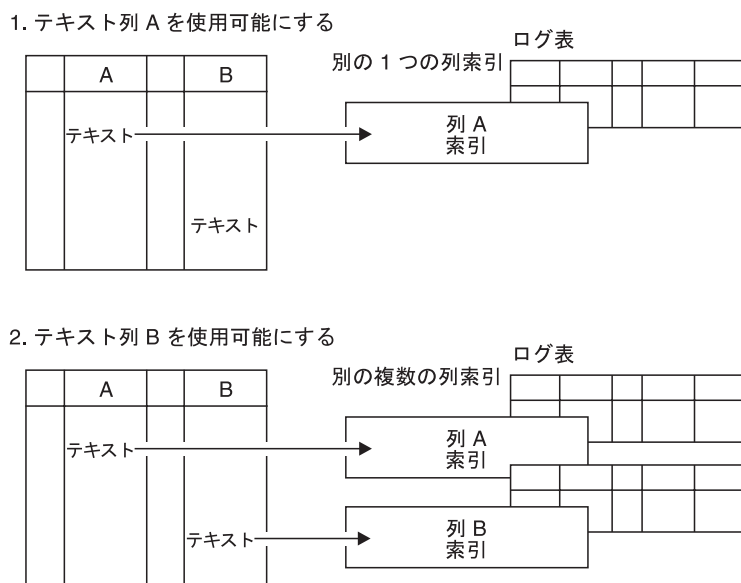


図 9. テキスト列ごとに別個の索引の作成

テキスト列の使用可能化

要約

実行時	検索するテキストが含まれる列ごとに 1 度ずつ
コマンド	CALL PGM(QDB2TX/DB2TX) PARM('ENABLE TEXT COLUMN ... (例を参照)')
権限	ユーザー・プロファイル QDESADM

ヒント

ほとんどのテキスト列について、索引タイプなどの設定を同じにする場合は、テキスト構成情報を使用してデフォルト設定を指定してください。10 ページの『テキスト構成設定値』を参照してください。

ENABLE TEXT COLUMN によって加えられた変更を取り消すには、DISABLE TEXT COLUMN コマンドを使用します。表中の使用可能なテキスト列をすべて使用不能にする場合は、DISABLE TEXT TABLE コマンドを使用します。

テキスト列を使用可能にする際には、表へのハンドル列の追加、文書情報 (形式、言語、CCSID) の設定、ログ表の作成、および索引の作成を行います。

ハンドル列の追加

このステップでは、DB2 テキスト・エクステンダーは表に 60 バイト VARCHAR ハンドル列を追加します。この列には、使用可能化されたテキスト列に関連するハンドルが含まれています。ハンドルには関連したテキスト列および関連した外部ファイル中のテキストに関する情報が含まれています。この情報には、固有の文書 ID、文書の言語、形式、CCSID、および索引名が含まれています。

DB2TX.SAMPLE

DOCID	AUTHOR	SUBJECT	DATE	COMMENT
データ	データ	データ	データ	テキスト

図 10. 使用可能にする前の DB2TX.MYTABLE 表の構造

テキスト・ブロックが含まれた列は COMMENT です。この列中のテキストを検索するには、まず DB2 テキスト・エクステンダーで使用するためのデータベースおよび COMMENT 列を作成しておかなければなりません。

こうした準備のステップを経て、DB2TX.MYTABLE 表には、次のようなハンドル用の追加列が含まれるようになります。

テキスト列の使用可能化

DB2TX.SAMPLE

DOCID	AUTHOR	SUBJECT	DATE	COMMENT	COMMENTHANDLE
データ	データ	データ	データ	テキスト	コメント・ハンドル

図 11. 使用可能にした後の DB2TX.MYTABLE 表の構造

注: 以後テキストの検索を行う際は、検索する列としてテキスト列ではなくハンドル列を指定します。

文書情報の設定

このテキスト列に保管する代表的なテキスト文書のタイプ、すなわちそれらの形式 (ASCII など)、言語、および CCSID を指定します。この情報のデフォルトをテキスト構成設定値に指定することができます。10 ページの『テキスト構成設定値』を参照してください。

ログ表の作成

このステップで、LOGIXnnnnnn と呼ばれるログ表およびビューが作成されます。ここで、IXnnnnnn は索引名です (カタログ・ビューから使用できます)。

また、トリガーも作成され、これにより、列内に文書の追加や変更が行われるたびに、ログに情報が追加されます。次に索引付けが行われるときに、この情報によって、これらの文書が索引付けされます。

外部ファイルが追加または変更される場合は、これらのトリガーは、その変更を感知しません。このような場合に、トリガーによってログ表に情報が追加されるようにするには、78 ページの『外部ファイルの索引の更新』に記載されている例のように、UPDATE ステートメントを使用します。

索引付け中に、索引付けの対象となる文書が待ち行列内に見つからないといったエラーが発生した場合は、エラー・イベント と呼ばれるイベントがログに追加され、84 ページの『エラー・イベントの表示』で説明するように、それを表示することができます。

ヒント

このステップで、ログ用スペースを使い尽くしてしまった場合は、51 ページの『ラージ表でのテキスト列の使用可能化』を参照して、可能な解決方法を見つけてください。

索引の作成

テキスト列ごとに別々の索引を作成する、すなわち ENABLE TEXT TABLE ステップをスキップする場合は、DB2 テキスト・エクステンダーはこのステップの間にそれぞれのテキスト列に別々の索引を作成します。索引のタイプ、索引を更新する周期、および索引を保管するディレクトリーを指定します。このようにせずに、表全体に 1 つの索引を作成する場合は、すでに ENABLE TEXT TABLE の実行と索引パラメーターの指定は行われているため、ここでこれらの手順を繰り返しても無視されます。

テキスト列の使用可能化

指定のテキスト列でテキスト文書の索引付けを即時に開始するかどうか、また次回いつ周期的な索引付けをスケジュールするかを決めるには、UPDATEINDEX キーワードを使用します。このキーワードを使用しない場合は、テキスト構成設定値内に指定された値が採用されます。

テキスト列に対するさまざまなタイプの索引の作成。 テキスト列には複数の索引を作成することができます。これは、たとえば、基本語形および Ngram など異なる索引タイプに列を関連させることによって、同じテキスト列に対して基本語形検索およびファジー検索を実行する場合に有用です。これを行うには、ENABLE TEXT COLUMN をもう一度実行し、作成する索引の追加タイプのほかに、固有のハンドル列名も指定します。

例

次の例では、DB2TX.MYTABLE 表中の COMMENT テキスト列を使用可能にし、作成されるハンドル列に COMMENTHANDLE という名前を割り当てます。

```
CALL PGM(QDB2TX/DB2TX) PARM('ENABLE TEXT COLUMN db2tx.mytable comment
                               HANDLE commenthandle')
```

テキスト情報および索引特性のデフォルト値は、テキスト構成設定値から得られません。

また、次の例は COMMENT 列中の文書タイプの値を明示的に設定します。索引特性のデフォルト値は、テキスト構成設定値から得られます。

```
CALL PGM(QDB2TX/DB2TX) PARM('ENABLE TEXT COLUMN db2tx.mytable comment
                               HANDLE commenthandle
                               CCSID      500
                               LANGUAGE   uk_english
                               FORMAT     rft')
```

次の例は COMMENT 列のために作成される索引の特性の値を明示的に設定します。この例では索引タイプと索引ディレクトリーを設定し、最少 100 のテキスト文書が待ち行列にある場合に月曜から金曜の 12:00 または 15:00 に索引が更新されるように索引の更新頻度を設定します。テキスト情報のデフォルト値は、テキスト構成設定値から得られます。

```
CALL PGM (QBD2TX/DB22TX) PARM('ENABLE TEXT COLUMN db2tx.mytable comment
                               HANDLE      commenthandle
                               INDEXTYPE  linguistic
                               UPDATEFREQ  min(100) d(1,2,3,4,5) h(12,15) m(00)
                               UPDATEINDEX UPDATE')
```

ラージ表でのテキスト列の使用可能化

ラージ行の長さの表で作業している場合、テキスト列を使用可能にすると、タイプ DB2TEXTH (VARCHAR 60) のハンドル列を追加することになることに注意してください。これは、表が DB2 により定められた最大の行長に近づいている場合には、重要です。

ラージ表でテキスト列を使用可能にする場合は、DB2 UDB for iSeries REORGANIZE ユーティリティーを使用して、その表を再編成する必要があるかどうかをチェックします。最初にラージ表を使用可能にするときには、以下のステップにより、索引付けを迅速に行うことができます。

テキスト列の使用可能化

1. NOUPDATE オプションを使用して、表を使用可能にする。これによって、ハンドルが作成されるが、文書の索引付けはまだ行われない。
2. RGZPFM ユーティリティを使用して、表を再編成する。
3. UPDATE INDEX を実行して、索引を作成する。

テキスト列または外部ファイルを使用可能にすると、DB2 テキスト・エクステンダーは、表にハンドル列を追加し、ハンドル値を初期化します。これによって、DB2 UDB for iSeries ログ項目が書き込まれます。書き込まれるログ項目の数が例外的に大きいと、DB2 UDB for iSeries によってログ用スペースが使い尽くされることがあります。

この状態に対処するには次の 2 つの方法がありますが、パフォーマンス上の理由から、最初の方法の使用をお勧めします。

- DB2 UDB for iSeries UPDATE DB CFG コマンドを使用して、LOGPRIMARY、LOGSECOND、および LOGFILSIZ のデータベース構成パラメーターを変更することによって、使用可能なログ用スペースを増やします。経験上、以下の値が採用されます。ユーザーのインストール状況に合うようにこれらの値を変更する必要がある場合もあります。

```
LOGSECOND      50
```

LOGPRIMARY と LOGSECOND の合計が 128 以下であることを確認してください。アプリケーション・ヒープ・サイズも増やす必要があります。

```
APPLHEAPSZ    512
```

- 10 ページの『構成』で説明されている COMMITCOUNT 構成パラメーターを使用して、DB2 UDB for iSeries が必ず中間 COMMIT を行うようにします。ユーザーが指定した値は、いくつの挿入ステートメントまたは更新ステートメントが出されると、DB2 テキスト・エクステンダーが DB2 UDB for iSeries コミット・ステートメントを出すかを示します。これによって、使用可能化ステップにかかる時間は増えますが、ログ表に必要なサイズは削減されます。

サポートされないデータ・タイプのテキスト列の使用可能化

DB2 テキスト・エクステンダーで使用可能化を行うためには、テキスト列は、CHAR、GRAPHIC、VARGRAPHIC、LONG VARGRAPHIC、DBCLOB、VARCHAR、LONG VARCHAR、または CLOB でなければなりません。ユーザー定義特殊タイプ (UDT) のように、これらと異なるタイプの列内の文書の場合は、ユーザー・タイプを入力とし、出力タイプとして CHAR、GRAPHIC、VARGRAPHIC、LONG VARGRAPHIC、DBCLOB、VARCHAR、LONG VARCHAR、または CLOB を提供する関数を用意する必要があります。

この関数の名前を指定するには、ENABLE TEXT COLUMN 中で FUNCTION キーワードを使用します。

例: 圧縮テキストを表に保管するとします。

1. テキストのユーザー定義特殊タイプ (UDT) を作成します。

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

2. 表を作成し、これにテキストを挿入します。


```
CREATE TABLE MYTABLE (author VARCHAR(50),
                       text COMPRESSED_TEXT)
INSERT ...
```

注

表の作成については、このガイドのパラメーターの範囲外です。詳しくは、「SQL 解説書」ガイドを参照してください。

テキスト列を DB2 テキスト・エクステンダーで使用できるようにするには、以下を実行します。

1. COMPRESSED_TEXT タイプの値を受け取って、対応する圧縮解除テキストを (たとえば CLOB(10M) 値として) 返すユーザー定義関数 (UDF) を作成し、その UDF に、たとえば UNCOMPRESS という名前を付けます。
2. FUNCTION キーワードを使用してテキスト列を使用可能にし、UNCOMPRESS UDF を識別します。

```
CALL PGM(QDB2TX/DB2TX) PARM('ENABLE TEXT COLUMN MYTABLE text
                             FUNCTION uncompress
                             HANDLE handle
                             ...')
```

外部テキスト・ファイルの使用可能化

このオプションは、以前のリリースとの互換性を保つためのみの目的で提供されています。

要約

実行時	オプション。検索される外部ファイルに関連した表ごとに 1 度。
コマンド	db2tx=>ENABLE TEXT FILES ... (例を参照)
権限	ユーザー・プロファイル QDESADM

DB2 テキスト・エクステンダーは、DB2 UDB for iSeries 表に保管されている文書のほか、IFS のルート部分に保管されたストリーム・ファイルに保管されたテキスト文書も検索することができます。この準備のステップは、外部ファイル内のテキストを検索しようとする場合に必要になります。外部テキスト・ファイルに関連した表を ENABLE TEXT TABLE コマンドによって使用可能にすることはできません。

49 ページの『テキスト列の使用可能化』で説明する方法と同様に、索引の作成、ログ表の作成、および文書情報の設定を行います。

ヒント

このステップで、ログ用スペースを使い尽くしてしまった場合は、51 ページの『ラージ表でのテキスト列の使用可能化』を参照して、可能な解決方法を見つけてください。

既存の DB2 UDB for iSeries 表にタイプが DB2TEXTFH のハンドル列が追加されます。ハンドル列には、外部ファイルへの参照、索引を含む各ハンドル、および文書情報 (CCSID、形式、および言語) が保持されます。

詳しくは 61 ページの『外部ファイルのハンドル』を参照してください。

ユーザーは、テキスト列の使用可能化の場合と同様に、デフォルト索引特性といった追加のパラメーターを指定することができます。

索引が作成された後は、外部ファイルを移動または削除することができます。ファイルでの検索は依然として実行できます。表に新しい列を挿入し、UPDATE INDEX を使用して新しいファイル参照で索引を更新することができます。

使用可能にしている表が複数の物理ノードのあるノード・グループを使用している場合には、表区画が常駐するノードに、表の列で参照する外部ファイルが位置していることを確認してください。

例

1. 少なくとも 1 つの列をもつ表 DB2TX.EXTFILE を作成する。
2. 表列にデータを挿入する。

対話式 SQL セッション内の場合:

```
INSERT INTO DB2TX.EXTFILE (DOCID) VALUES ('doc1')
```

3. ハンドル列 FILEHANDLES を表 DB2TX.EXTFILE に追加する。

```
CALL PGM(QDB2TX/DB2TX) PARM('ENABLE TEXT FILES db2tx.extfile
                               HANDLE      filehandles
                               INDEXTYPE   linguistic
                               UPDATEFREQ  min(100) d(1,2,3,4,5) h(12,15) m(00)
                               UPDATEINDEX NOUPDATE" ')
```

4. 外部ファイルの名前を指定して、ハンドル「ハンドルの更新」列を初期化し、外部ファイル参照を反映します。

```
UPDATE db2tx.EXTFILE
SET FILEHANDLES = db2tx.file(FILEHANDLES,
                              'QIBM/UserData/DB2Extenders/Text/docs/doc1.txt')
WHERE DOCID = 'doc1'
```

ヒント

外部ファイルを参照するハンドル列の更新には INIT_TEXT_HANDLE を使用してはなりません。

5. 索引を更新する。

```
CALL PGM (QDB2TX/DB2TX)  
      PARM ('UPDATE INDEX db2tx.extfile HANDLE filehandles')
```

セッションの終了

これで検索するテキスト文書を準備するステップが完了しました。

テキスト列を使用可能にするときに、UPDATEINDEX キーワードとして NOUPDATE を指定すると、DB2 テキスト・エクステンダーは、そのテキストをすぐには索引付けしないで、次の索引付けの周期がくるまで待ちます。今すぐ索引を更新する場合は、78 ページの『索引の更新』を参照してください。

文書の索引付けが終了すれば、57 ページの『第 6 章 検索方法』で説明されているような情報検索を開始できます。

ヒント

いつ索引付けが終了したかを判断するためには、GET INDEX STATUS を使用します。

DB2 テキスト・エクステンダー・コマンド・プロセッサを終了するには以下を入力します。

```
db2tx=>quit
```

セッションの終了

第 6 章 検索方法

DB2 テキスト・エクステンダーには、SQL 照会へテキスト検索副照会を組み込めるようにする SQL 関数が備わっています。この種の関数は、SQL で一般的に利用できる関数に加えて用意されており、ここでは DB2 テキスト・エクステンダー関数と呼ばれます。

これらの関数の構文の詳細は、153 ページの『第 11 章 検索関数』を参照してください。

検索を行う前に、31 ページの『検索のタイプ』を読み、さらに GET INDEX SETTINGS を使用して検索するテキストに関連した索引タイプを見つけ出してください。検索結果は、索引タイプにより異なる場合があります。この章の例では、索引タイプを基本語形索引と仮定しています。

この章では、以下の項目について説明します。

- サンプル DB2 テキスト・エクステンダー関数
- DB2TX.SAMPLE サンプル表
- 外部ファイルのハンドル
- SQL アクセス権を DB2 テキスト・エクステンダー関数に与える関数パスの設定
- CONTAINS、NO_OF_MATCHES、および RANK を使用するテキストの検索
- CONTAINS の例を使用した、DB2 テキスト・エクステンダー関数の検索引き数の指定
- CONTAINS と REFINE を使用した、直前の検索の詳細化
- CCSID、FORMAT、および LANGUAGE を使用する、ハンドル内の情報の設定と抽出

検索関数の構文例の情報源

DB2 テキスト・エクステンダーは、DB2 テキスト・エクステンダーの製品ライブラリー QDB2TX の物理ファイル SAMPLES に保管されている、TXQUERIES と呼ばれるサンプル・ファイル・メンバーを提供しています。これには、サンプル表に対して実行される DB2 テキスト・エクステンダー検索関数の例が含まれています。このファイルは、DB2 テキスト・エクステンダーのテキスト準備および検索関数の構文や検索引き数で使われる構文の例を調べるために使用してください。

DB2TX.SAMPLE サンプル表

検索構文例の多くは、サンプル表 DB2TX.SAMPLE を使用しています。これらの例は、DB2 テキスト・エクステンダーを使用して自分で実行してみることができます。

表 DB2TX.SAMPLE を作成するには、9 ページの『インストール検査のためにサンプル・データベースを準備する』を参照してください。

DB2TX.SAMPLE サンプル表

表 DB2TX.SAMPLE から抜粋したものを表 5 に示してあります。

表 5. DB2TX.SAMPLE 表の例からの抜粋

DOCID	AUTHOR	SUBJECT	DATE	COMMENT
doc 5	CHGVMIC1 の RSSHERM	LIBDB2E.A エ ラー	1995-07-25 -20.13.59	顧客は、「No such file or directory」とい うエラー・メッセージを LIBDB2E.A に受 け取っています。これは、非同期入出力ド ライバーに関連するエラー・メッセージで はないようです。この顧客は、3.2.5 でベ ータ 4 を使用しています。顧客に /usr/lpp/db2_02_01/lib のファイルの許可と所 有権を私のものと比較してもらいま したが、現在それらは同じものになっていま す。この顧客の .profile および ENV も問 題ないようです。あいにくこの顧客は、イ ンストールを行って (COMMIT を実行し て) いました。そのほかに何かが誤ってい る可能性があります。
doc 6	SYDVM1 の EDWARDSC	DDCS/2 からの 小文字のユーザ ー ID およびパ スワード	1995-07-25 -20.15.20	再チェック後、大文字小文字の区別に問題 のあったインスタンスが、MVS への DB2/2 ゲートウェイを使用していました。 インスタンスに小文字のユーザー ID を渡 した場合には、そのゲートウェイは使用で きませんでした (パスワードは無関係)。大 文字のユーザー ID を実際に入力したとき だけ接続に成功しました。そこで、残念で すがこれはあなたの状態の参考にはならな いと思います。
doc 7	TOROLAB4 の SKY	ODBC および ストアード・プ ロシージャ	1995-07-25 -20.42.27	CLI (ODBC) を使用するストアード・プロ シージャの使用法を説明するサンプル・ プログラムが 2 セットあります。 C ファイル inpsrv2.c (サーバー上にある) および C ファイル inpli2.c (クライアント 上にある) は、入力用にストアード・プロ シージャを使用する方法を説明するサン プルを作成します。ファイル outsrv2.c およ び outcli2.c は、出力用にストアード・プロ シージャを使用する方法を説明するサン プルを作成します。 これらのファイルは、.../sqllib/samples/cli フ ァイルの一部です。MAKE ファイルは、 これらのファイルを自動的に作成し、サー バーのファイルを正しいサブディレクトリ ーに転送します。

表 5. DB2TX.SAMPLE 表の例からの抜粋 (続き)

DOCID	AUTHOR	SUBJECT	DATE	COMMENT
doc 8	TOROLAB2 の ADAMACHE	DB2SYS.DLL アクセス違反	1995-07-25 -21.13.22	<p>前のベータ・バージョンをインストールしてありましたか。もしそうなら、ソフトウェア・インストーラーを使用してそれを除去しましたか。</p> <p>データベース・ディレクトリー (SQLDBDIR および SQL00001 など) を以前のベータ・ドライバーから除去しましたか。</p>
doc 9	TOROLAB2 の ADAMACHE	CREATE DB = SYS3175: db2eng.dll 内の db2sysc.exe	1995-07-25 -21.40.09	<p>DB2/2 ベータ・ユーザーの多くは、ソフトウェア・インストーラーを使って前バージョンのベータを削除し、ベータ 5 (現在ではゴールドデン・コード) をインストールしてデータベースを作成し、SYS3175: db2eng.dll 内の db2sysc.exe を入手します。</p> <p>これは、ディレクトリー形式がベータ 4 とベータ 5 では変わっているために起こります。DB2/2 インストールの際に、ベータ・ドライバー間で sqlbdir ディレクトリーはマイグレーションされません。sqlbdir と sql000x ディレクトリーおよび ¥sqllib¥db2¥sqlbdir ディレクトリーがあればすべて除去する必要があります。</p> <p>行うべきことは、ソフトウェア・インストーラーを使用して前バージョンのベータを削除し、sqlbdir、sql000x、および ¥sqllib¥db2¥sqlbdir ディレクトリーがあればすべて除去した後で、新しいコードをインストールすることです。</p>
doc 10	CHGVMIC1 の RSSHERM	DB2/NT - SNA サポート	1995-07-25 -22.10.15	<p>DB2/NT は、CAE クライアントと WIN クライアント両方に対するサーバーとして、および SNA ネットワーク上の DB2/6000 か DB2/MVS (またはその両方) に対するクライアント (ホップ) として稼働可能ですか。そうでなければ、DB2/NT から、DB2/6000 または DB2/MVS (あるいはその両方) への、再度 SNA を経由する DRDA がサポートされていますか。</p>

DB2TX.SAMPLE サンプル表

次に、この表の構造の一部を、最初と最後の列を記述して示します。

DB2TX.SAMPLE

DOCID	COMMENT
文書 1	Customer is ...
文書 2	After rechecking ...

図 12. DB2TX.SAMPLE 表の構造

検索されるテキストが入っている列は COMMENT です。しかし、この列の中のテキストを使用して検索を行うには、ENABLE TEXT COLUMN コマンドを実行して、DB2 テキスト・エクステンダーが使用する COMMENT 列を事前に作成しておく必要があります。

こうした準備段階を経て、DB2TX.SAMPLE 表は次のようになります。

DB2TX.SAMPLE

DOCID	COMMENT	COMMENTHANDLE
文書 1	Customer is ...	X'..handle..'
文書 2	After rechecking ...	X'..handle..'

図 13. 使用可能にした後の DB2TX.SAMPLE 表

ここで、上記の表にはハンドルに関する列が追加され、個々のテキスト・オブジェクトにそれを表す固有のハンドルが与えられています。

使用可能なテキスト列に後でテキストを挿入すると、挿入トリガーがそのテキストのハンドルを作成します。

DB2TX.SAMPLE

DOCID	COMMENT	COMMENTHANDLE
文書 1	Customer is ...	X'..handle..'
文書 2	After rechecking ...	X'..handle..'

ENABLE TEXT COLUMN によって作成されるハンドル

次の行を追加する。

文書 11	I have installed ...	X'..handle..'
-------	----------------------	---------------

挿入トリガーによって作成されるハンドル

図 14. トリガーによる挿入行のハンドル作成

ハンドルには、以下の情報が含まれます。

文書 ID

関連する索引の名前と場所

文書情報: CCSID、形式、および言語

DB2 テキスト・エクステンダーが提供する SQL 関数は、ハンドルをパラメーターとして、表の SQL 処理の一環として、テキストの保管、アクセス、検索、および処理を行います。

外部ファイルのハンドル

DB2 テキスト・エクステンダーは、DB2 UDB for iSeries 表に保管されているテキスト以外に、あらゆる保管されたテキスト・ファイルを検索することができます。53 ページの『外部テキスト・ファイルの使用可能化』で、DB2 UDB for iSeries 表に保管されていないテキスト文書内での検索を可能にする準備ステップを説明しています。このステップでは、ENABLE TEXT FILES コマンドによって、外部ファイルのハンドル用にタイプ DB2TEXTFH のハンドル列が作成されます。既存の表にハンドル列が追加されます。

たとえば、作成者の名前と作成日付のための列を含む表を作成しておくことができます。

INIT_TEXT_HANDLE を使用して、ファイル・ハンドルを初期化します。各ハンドルには、文書 ID、関連した索引の名前と位置、および文書情報 (CCSID、形式、および言語) の他に、外部ファイルへの参照も含まれます。

現行の関数パスの設定

▶▶ SET CURRENT FUNCTION PATH [=] DB2TX, ... ▶▶

DB2TX を現行パス名に追加して SQL が DB2 テキスト・エクステンダー関数を検出できるようにするには、SQL ステートメントの SET CURRENT FUNCTION PATH を使用します。これを行わない場合、関数名を明示的に修飾できます。たとえば CONTAINS という関数の場合は DB2TX.CONTAINS と入力します。

この章の例では、修飾形式の DB2 テキスト・エクステンダー関数を使用しています。例にあるステートメントは、現行関数パスを設定せずに、例の中で記述されているとおりに入力すれば使用できます。

ヒント

データベースへの接続時には、現行関数パスを常に忘れずに設定してください。

テキストの検索

▶▶ CONTAINS (—handle—, —search-argument—) ▶▶
 NO_OF_MATCHES
 RANK

このセクションでは、DB2 テキスト・エクステンダーに備えられている SQL 関数を使用して、テキストを含んだ DB2 データベースを検索する方法について説明します。以下の検索方法について説明します。

- 照会の作成
- テキスト文書中で検出された一致数の判別
- 検出されたテキスト文書のランクの入手

テキストの検索

この種の DB2 テキスト・エクステンダー関数はそれぞれ、テキスト索引中で検索引き数を検索します。表の中に 100 000 個のテキスト文書があるとすれば、CONTAINS、RANK、または NO_OF_MATCHES 関数は 100 000 回呼び出されます。ただし、テキスト索引が 100 000 回検索されることはありません。その代わりに、関数の最初の呼び出し時に、検索項目を含む全文書の内部リストが作成されます。2 回目以降の関数の呼び出しでは、関係する文書がそのリスト内にあるかどうか判别されます。

ヒント

DB2 テキスト・エクステンダーの関数を使用して表の中を検索する場合、テキスト列ではなく関数に必ずハンドル列を渡してください。テキスト列の中を検索しようとする、SQL は次のようなメッセージで応答し、データ・タイプが誤っていることを示します。

```
No function by the name "CONTAINS" having compatible arguments was found in the function path.
```

ENABLE TEXT TABLE または ENABLE TEXT COLUMN コマンドを出した直後にテキストを検索する場合、そのコマンドにより作成中の索引がまだ存在しないことを示すエラー RC_SE_EMPTY_INDEX が発生することがあります。索引の作成に要する時間は、索引を作成する文書の数や、索引の作成を実行するシステムのパフォーマンスなどの要素によって決まります。作成時間は、数分から数時間の範囲です。それで、深夜など、システムの負荷が軽い時間帯に実行するべきです。

このメッセージが出されたら、後で検索を再試行するか、GET INDEX STATUS を使用して、索引付けエラーが起きていないかをチェックします。

照会の作成

次の SQL の例は、CONTAINS 関数がハンドル名によって識別される文書内のテキストをどのように検索するかを示しています。テキストが検索引き数を満たす場合は 1 が戻され、満たさない場合は 0 が戻されます。

```
SELECT DATE, SUBJECT
       FROM DB2TX.SAMPLE
       WHERE DB2TX.CONTAINS (COMMENTHANDLE, '"compress"') = 1
```

この例では、COMMENTHANDLE 列内のハンドルにより参照されるテキストの中で compress という用語を検索します。COMMENTHANDLE 列の中のハンドルは、COMMENT テキストが索引付けされる位置を指示します。

ヒント

表または列中に大文字小文字混在の ID を作成した場合は、これらの名前を二重引用符で囲む必要があります。例:

```
SELECT DATE, SUBJECT
      FROM "db2tx.sample"
      WHERE DB2TX.CONTAINS (COMMENTHANDLE, "compress") = 1
```

DB2 UDB for iSeries の select ステートメントをコマンド行より指定すると、オペレーティング・システムのコマンド行パーサーにより、コマンド・ストリングから二重引用符などの特殊文字が除去されるため、これらの特殊記号をマスクするためには、¥ 記号を使用する必要があります。例:

```
WHERE DB2TX.CONTAINS (COMMENTHANDLE, '¥"compress¥") = 1
```

ヒント

照会の結果がデータベース・タイムアウト・エラーになる場合は、QQAINI ファイル設定の UDF_TIME_OUT 値を増やす必要があります。

検出された一致数の検索と入手

各テキスト文書中に検索基準が何回検出されるかを判別するには、NO_OF_MATCHES 関数を使用してください。

NO_OF_MATCHES は整数値を返します。

検出テキスト文書のランクの検索と入手

RANK は、検出された文書が検索基準を満たす度合いを、他の検出文書と比べて示す絶対値です。この値は、文書内で検出された一致数と、その文書サイズを示します。RANK 関数を使用することによって、検出された文書のランクを入手できます。

次に例を示します。

RANK は、0 ~ 1 の DOUBLE 値を返します。

検索引き数の指定

CONTAINS、NO_OF_MATCHES、および RANK では、検索引き数が使用されます。このセクションでは、CONTAINS 関数を使用して、DB2 テキスト・エクステンダー関数での検索引き数のさまざまな例を示します。

複数の用語の検索

検索引き数に複数の用語を指定することができます。複数の検索項目を結合する方法として、次のようにコンマを使ってそれらを結合する方法があります。

検索引き数の指定

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
'("compress", "compiler", "pack", "zip", "compact")') = 1
```

このような検索引き数の書式では、それらの検索項目を含むテキストであれば、どれでも検出されます。論理項の場合、検索項目は OR 演算子でつなぎます。

ブール演算子 AND および OR を使用した検索

(68 ページの『ブール演算子 NOT を使用した検索』も参照してください。)

ブール演算子「&」(AND) および「!」(OR) を使用すると、ある検索項目を別の検索項目と結合することができます。例:

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
'"compress" ! "compiler"') = 1
```

ブール演算子を使用して複数の用語を結合することができます。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
'"compress" ! "compiler" & "DB2"') = 1
```

複数のブール演算子を使用すると、DB2 テキスト・エクステンダーはそれらを左から右へ評価しますが、論理 AND 演算子 (&) は論理 OR 演算子 (!) よりも優先的にバインドされます。たとえば、次の例で括弧を付けません。

```
"DB2" & "compiler" ! "support" & "compress"
```

これは次のように評価されます。

```
("DB2" & "compiler") ! ("support" & "compress")
```

したがって、次の例では括弧を付ける必要があります。

```
"DB2" & ("compiler" ! "support") & "compress"
```

ブール演算子をコンマ区切り文字を使用して 1 つにつなげた検索項目と結合する場合は、次のようにします。

```
("compress", "compiler") & "DB2"
```

コンマは、次のようにブール OR 演算子として解釈されます。

```
("compress" ! "compiler") & "DB2"
```

用語の変化形の検索

一致語句索引を使用している場合、DB2 テキスト・エクステンダーはユーザーが入力したとおりの用語を検索します。たとえば、media という用語からは「media」を含んだテキストしか検出されません。単数形の「medium」を含むテキストは検出されません。

基本語形索引を使用している場合、DB2 テキスト・エクステンダーは名詞の複数形や動詞の別の時制などの用語の変化形も検索します。

たとえば、drive という用語は、「drive」、「drives」、「driving」、「drove」、および「driven」という語を含むテキストを検出します。

用語の一部 (マスキング文字) の検索

マスキング文字 (または「ワイルドカード」文字と呼ぶ) を使用すると、検索をより柔軟に行うことができます。マスキング文字とは、検索項目の前部、中央部、および後部で任意指定される文字のことを指します。この文字を使用すると、検索の結果検出されるテキスト文書の数が増加します。

ヒント

マスキング文字を使用した場合、SYNONYM FORM OF キーワードを使用することはできません。

マスキング文字は、一致語句索引の場合に用語の変化形を検出するのに特に便利です。基本語形索引の場合、マスキング文字を用いて検出される変化形の多くは、マスキング文字を用いなくても検出されます。

語の断片 (ワイルドカード文字でマスクされた語) は、基本形に変形できないという点に注意してください。したがって、`pass%` を検索すると、「`passes`」や「`passed`」という語は索引の中で基本形「`pass`」に変形されるため、検出されません。これらの語を検出するには、`pass%` により検索しなければなりません。

DB2 テキスト・エクステンダーは、2 つのマスキング文字、パーセント (%) と下線 (_) を使用します。

- % は、**任意の個数の任意の文字**を表します。次に示すのは、% をマスキング文字として検索項目の前に使用した例です。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE, '%"name"') = 1
```

この検索項目は、「`username`」、「`file_name`」、および「`table-name`」などを含んだテキスト文書を検出します。

- % で**語全体**を表すことも可能です。次の例では、「`graphic function`」および「`query function`」などの句を含んだテキスト文書が検出されます。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE, '%" function"') = 1
```

- _ は、検索用語中の **1 文字**を示します。次の例では、「`CLOB`」および「`BLOB`」を含むテキスト文書を検出しています。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE, '%_LOB') = 1
```

マスキング文字をすでに含む用語の検索

「%」文字または「_」文字を含む用語を検索したい場合は、その文字の前になんらかのエスケープ文字を付けてください。そして、ESCAPE キーワードを使用してそのエスケープ文字を指定します。

検索引き数の指定

たとえば、「100% interest」を検索するには、次のように指定します。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                      '"100!% interest" ESCAPE "!"') = 1
```

この例でのエスケープ文字は "!" です。

任意の順序での用語の検索

次の例で示されているように「hard disk」を検索すると、使用している索引タイプにかかわらず、この 2 つの用語が隣接してその順序で並んでいる場合にのみ検出されます。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE, '"hard disk"') = 1
```

たとえば、「data disks and hard drives」のように任意の順序で並んだ用語を検索するには、コンマを使って用語を分離してください。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE, '("hard", "disk"') = 1
```

同じ文または段落中の用語の検索

次に示すのは、指定した複数の検索項目が同じ文中にあるテキスト文書を検出する検索引き数の例です。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                      '"compress" IN SAME SENTENCE AS "decompress"') = 1
```

また、2 つ以上の語を一緒に検索することもできます。次の例では、同じ段落にいくつかの用語が見つかる文書が検索されます。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                      '"compress" IN SAME PARAGRAPH AS "decompress"
                      AND "encryption"') = 1
```

構造化文書のセクションでの用語の検索

次に示すのは、検索項目 Williams が、構造化文書のセクション play のサブセクション author 中にあるテキスト文書を検出する検索引き数の例です。文書構造は、文書モデル・ファイルで記述されたモデル play によって指定されます。詳細については、37 ページの『構造化文書 (セクション・サポート) での作業』を参照してください。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                      'MODEL play SECTIONS (play/author) "williams"') = 1
```

用語の同義語の検索

基本語形索引または併用索引の場合、指定の検索項目だけでなく、類似した意味を持つ語も探すことにより、より柔軟な検索が可能になります。たとえば、「book」という語を検索する場合、その同義語も検索すると便利ことがあります。これを行うには、次のように指定します。

```
SELECT DATE, SUBJECT
       FROM DB2TX.SAMPLE
       WHERE DB2TX.CONTAINS (COMMENTHANDLE, 'SYNONYM FORM OF "book"') = 1
```

SYNONYM FORM OF を使用すると、用語の同義語が論理 OR 演算子でつながれていると見なされます。つまり、検索引き数が次のように解釈されます。

```
"book" ! "article" ! "volume" ! "manual"
```

同義語は、DB2 テキスト・エクステンダーに備えられている辞書の中にあります。同義語用に使用されるデフォルトの辞書は常に US_ENGLISH で、テキスト構成設定値で指定された言語ではありません。

別の言語を指定して特定の照会用に辞書を変更することができます。次に例を示します。

```
SELECT DATE, SUBJECT
       FROM DB2TX.SAMPLE
       WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                             'SYNONYM FORM OF UK_ENGLISH "programme"') = 1
```

ヒント

SYNONYM キーワードは、検索項目の中にマスキング文字がある場合や、NOT が検索引き数と共に使用されている場合には使用できません。

基本語形検索の実行

DB2 テキスト・エクステンダーには、ユーザー指定の検索項目に基づいた検索を行うための、高性能の言語的な処理があります。言語的な機能は、索引が基本語形索引である場合に適用されます。言語関数についての説明は 197 ページの『第 15 章 基本語形索引と一致語句索引の言語的な処理』に記載しています。

これはたとえば、「utilities」などの複数形の語を検索して、「utility」を検出するような場合です。この複数形は、検索前に、英語の辞書を使用して、基本形である utility に変形されます。

しかし、英語の辞書には、他の言語の用語の変化形を基本形に戻すための情報がありません。他言語の用語の複数形を検索するには、その言語の辞書を使用する必要があります。

たとえば、GERMAN を指定した場合は、「geflogen」（「飛ぶ」の過去分詞）を検索し、その基本形「fliegen」（飛ぶ）のすべてのバリエーションを検出します。つまり、「geflogen」だけでなく、「fliege」、「fliegt」なども検出されます。

```
SELECT DATE, SUBJECT
       FROM DB2TX.SAMPLE
       WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                             'STEMMED FORM OF GERMAN "geflogen"') = 1
```

ヒント

米国英語以外の文書で検索を行う際は、デフォルトの言語に関係なく 検索引き数で言語を指定してください。

検索項目の変化形ではなく基本形を常に指定する場合は、言語を指定する必要はありません。

その理由を理解するには、データベース内のテキストが索引付けされる際に何が起るかを考えてください。基本語形索引または併用索引を使用している場合、用語の変化形は基本形に変形されてから索引に保管されます。つまり、DB2TX.SAMPLE表を例にとると、「decompress」という語が COMMENT 列の最初の項目にあり、「decompression」という語が 2 番目の項目にある場合でも、索引には基本形「decompress」しか含まれず、しかもこの用語 (またはその変化形) が両方の項目にあるものとして識別されます。

その後、基本形「decompress」を検索すると、その変化形がすべて検出されます。しかし、「decompression」などの変化形を検索しても、その語を直接検出することはできません。適切な検索用の辞書を指定する必要があるので、変化形は最初にその基本形に変換されます。

ブール演算子 NOT を使用した検索

ブール演算子 NOT を使用すると、特定のテキスト文書を検索から除外することができます。例:

```
("compress", "compiler") & NOT "DB2"
```

この場合、「DB2」という用語を含んだテキスト文書はすべて、「compress」または「compiler」の検索からは除外されます。

NOT 演算子は、66 ページの『同じ文または段落中の用語の検索』で説明した IN SAME SENTENCE AS または IN SAME PARAGRAPH AS と組み合わせて使用したり、67 ページの『用語の同義語の検索』で説明した SYNONYM FORM OF とともに使用することはできません。

NOT 演算子は、基本検索語 (search-primary) としてしか使用できません。つまり、&、!、および NOT 演算子を自由に組み合わせることはできません (166 ページの『検索引き数構文』を参照してください)。

NOT の誤った使用例:

```
NOT("compress" & "compiler")
```

正しい使用例:

```
NOT("compress" , "compiler")
```

ファジー検索

「ファジー」検索では、検索項目と類似したつづりの語が検索されます。これは、Ngram 索引についても適用可能です。

例:

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                      'FUZZY FORM OF 2 "compress") = 1
```

この検索では、つづりが誤りである語 compress を検出することができます。

この例の一致レベル“2”は正確度を示します。これは5レベルまでサポートされますが、レベル1は最も正確度が低く、約20%の一致を示し、レベル5は最も高い正確度で、約90%の一致を示します。光学式文字認識装置または音声による入力によく起こることがありますが、文書内につづりの誤りがあると考えられる場合は、ファジー検索を使用します。

語句の境界の考慮

韓国語用に「Bound」検索が開発されました。これは、DB2 テキスト・エクステンダーが検索時に、必ず語の境界を考慮するようにするものです。例:

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                      'BOUND "korean-expression") = 1
```

発音の似た語の検索

「Sound」検索では、検索引き数と発音の似た語を検出します。これは、発音は似ているが、つづりが異なる語が文書に含まれている可能性がある場合に役立ちます。たとえば、my-er と発音するドイツ語の名前にはいくつかのつづりがあります。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
                      'SOUNDS LIKE "Meyer") = 1
```

この検索では、「Meyer」、「Mayer」、および「Maier」が検出できます。

シソーラス検索

シソーラス検索は、DB2 テキスト・エクステンダーのもう1つの高機能な検索項目拡張機能です。検索される追加の用語は、ユーザー自身が作成したシソーラスから取得されるため、それらを直接制御できます。たとえば、「database」を検索して、「repository」や「DB2」といった用語を検出できます。

この検索タイプは、頻繁に検索を行う、関心のある特定の領域向けのものです。つまり、きわめて有効な検索結果を生成するためにシソーラス (類語事典) を作成する時間をかける価値がある領域です。

シソーラスの作成方法の詳細と説明については、206ページの『シソーラスの概念』を参照してください。206ページの図17の例は、データベースの対象のシソーラスからの小規模な抜粋です。これは、シソーラスの拡張を使用する構文を示す次の例で使用されます。

この例は、「object relational database management system」という用語を選び、それにシソーラス「myterms」で検出されるこの用語のすべてのインスタンスを追加して拡張します。ここでは、「DB2」が検索に追加されます。

検索引き数の指定

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
    'THESAURUS "myterms"
    EXPAND "INST"
    TERM OF "object relational database
    management system") = 1
```

次の例は、「document management system」という用語を選び、それにすべての同義語を追加して、それを拡張します。ここでは、1つの同義語である「library」が追加されます。

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
    'THESAURUS "myterms"
    EXPAND "SYN"
    TERM OF "document management system") = 1
```

フリー・テキストと混成検索

「フリー・テキスト検索」では、検索項目が自由形式テキストで表されます。句または文が、検索の対象を自然言語で記述しています。フリー・テキスト照会では、語順は関係ありません。さらに、語いの類縁性 これは検索では、特定の最小頻度、かつ特定の最短距離が指定されたフリー・テキスト照会用語に出現する語と文書集合内に出現する語のあるペアです。英語の文書の最短距離は 5 語です。

フリー・テキスト引き数内の検索ストリングに対しては、文字または語のマスキングがサポートされないことに注意してください。

例:

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
    'IS ABOUT "everything related to AIX installation") = 1
```

混成検索は、ブール検索とフリー・テキスト検索の組み合わせです。例:

```
SELECT DATE, SUBJECT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE,
    '"DB2" & IS ABOUT "everything related to
    AIX installation") = 1
```

直前の検索の詳細化

検索引き数によって検出されるものが多過ぎるときは、最初の検索引き数と 2 番目の検索引き数をブール AND の関係で組み合わせて検索範囲を狭くする、つまり検索を詳細化すると便利な場合がよくあります。

検索結果の詳細化は、その結果を表に保管し、次にその表と対照して検索を行うことにより、REFINE 関数を使わなくても行うことができます。しかし、この方式は、修飾用語の数によっては最新の検索引き数を保管して REFINE を使用する方式よりも効果的でない場合があります。

以下のステップでは、検索後に REFINE 関数を使用してその結果を詳細化する方法を示します。REFINE 関数は、2つの入力パラメーターをブール AND で組み合わ

せた検索引き数を戻します。REFINE により戻される検索引き数の組み合わせは、LONG VARCHAR タイプの値となります。

1. 古い検索引き数用の表を作成します。

すでに実行した検索の検索引き数を保持するには、表 PREVIOUS_SEARCHES を作成してください。

```
CREATE TABLE PREVIOUS_SEARCHES (step INT,
                                searchargument LONG VARCHAR)
```

PREVIOUS_SEARCHES

STEP	SEARCHARGUMENT

2. 最初の検索引き数を検索します。

サンプル表の中で「compress」という語を検索します。

```
SELECT COMMENT
FROM DB2TX.SAMPLE
WHERE DB2TX.CONTAINS (COMMENTHANDLE, "compress") = 1
```

その後のステップで使用するために、検索引き数を PREVIOUS_SEARCHES 表に挿入してください。

```
INSERT INTO PREVIOUS_SEARCHES
VALUES (1, "compress")
```

PREVIOUS_SEARCHES

STEP	SEARCHARGUMENT
1	"compress"

3. 検索を詳細化します。

検索により戻されるテキスト文書が多過ぎると仮定し、REFINE 関数を使用して直前の検索項目を「compiler」という語と組み合わせて検索を詳細化します。

```
WITH LAST_STEP(STEP_MAX)
AS (SELECT MAX(STEP)
    FROM PREVIOUS_SEARCHES),
LAST_SEARCH(LAST_SEARCH)
AS (SELECT SEARCHARGUMENT
    FROM PREVIOUS_SEARCHES, LAST_STEP
    WHERE STEP = STEP_MAX)
SELECT COMMENT
FROM DB2TX.SAMPLE, LAST_SEARCH
WHERE DB2TX.CONTAINS(COMMENTHANDLE,
                    DB2TX.REFINE(LAST_SEARCH, "compiler")) = 1
```

その後のステップで使用するために、詳細化された検索引き数を PREVIOUS_SEARCHES 表に挿入してください。

```
INSERT INTO PREVIOUS_SEARCHES
WITH LAST_STEP(STEP_MAX)
AS (SELECT MAX(STEP)
    FROM PREVIOUS_SEARCHES)
SELECT STEP_MAX+1, DB2TX.REFINE(SEARCHARGUMENT, "compiler")
FROM PREVIOUS_SEARCHES, LAST_STEP
```

直前の検索の詳細化

PREVIOUS_SEARCHES

STEP	SEARCHARGUMENT
1	"compress"
2	"compress" と "compiler"

このステップは、検出されるテキスト文書数が十分少なくなるまで繰り返すことができます。

ハンドル内の情報の設定と抽出

ハンドルには、CCSID、形式、およびそれらのテキスト文書の言語が含まれます。外部ファイルのハンドルにはさらに、その外部ファイルへのポインターが含まれます。このようなハンドルは、テキスト列または外部ファイルを使用可能にすると作成されます。

以下に説明する DB2 テキスト・エクステンダー関数を使用すると、ハンドル内のテキスト情報の設定や抽出を行うことができます。

ハンドルからの情報の抽出



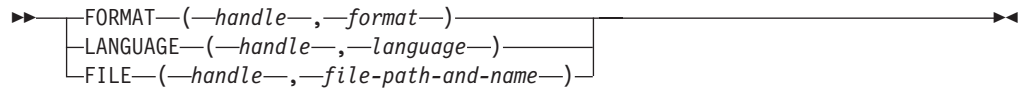
次の例は、ハンドルから CCSID を抽出します。

```
SELECT DISTINCT DB2TX.CCSID(COMMENTHANDLE)
FROM DB2TX.SAMPLE
```

同じ方法を使って、テキスト文書の形式や言語、または外部ファイルの位置をハンドルから抽出することができます。次に示す内容は、FORMAT 関数の使用法について説明した例です。この関数は、ASCII (TDS) 文書の数をサンプル表の中に戻します。

```
SELECT COUNT(*)
FROM DB2TX.SAMPLE
WHERE DB2TX.FORMAT(COMMENTHANDLE) = 'TDS'
```

ハンドル内の情報の変更



FORMAT および LANGUAGE 関数は、ハンドル内のそれぞれに対応する仕様を変更することもできます。これらの関数は、変更されたハンドルを DB2TEXTH タイプ、または DB2TEXTFH タイプの値として戻します。

次の例では、テキスト文書の言語設定を変更する方法について示します。

```
UPDATE DB2TX.SAMPLE
  SET COMMENTHANDLE = DB2TX.LANGUAGE(COMMENTHANDLE, 'FRENCH')
  WHERE ...
```

LANGUAGE 関数を再び使用すると、変更が行われたことを確認することができます。

```
SELECT DISTINCT DB2TX.LANGUAGE(COMMENTHANDLE)
  FROM DB2TX.SAMPLE
```

検索パフォーマンスの向上

SEARCH_RESULT 関数では、DB2 の表の値に基づく関数の概念が活用されています。関数は、SQL ステートメントの FROM 文節内で使用されて、指定された検索ストリングの検索結果の入った中間表を戻します。検索ストリングの構文は、165 ページの『第 12 章 検索引き数の構文』で説明するものと同じです。CONTAINS または RANK と比較した場合のこの関数の利点は、ラージ表が含まれている場合にかなりパフォーマンスの向上が見られるという点です。

戻される表の構造は次のとおりです。

Column Name	Data type
HANDLE	DB2TX.DB2TEXTH
NUMBER_OF_MATCHES	INTEGER
RANK	DOUBLE

例:

```

WITH REPHANDLE (MYDOCHANDLE)
  AS (SELECT PROTOTYPEHANDLE
      FROM DB2TX.TEXTCOLUMNS
      WHERE TABLESCHEMA = 'DB2TX'    AND
          TABLENAME = 'SAMPLE'      AND
          HANDLENAME = 'H_L'
      )
SELECT DOCID, NUMBER_OF_MATCHES, RANK, HANDLE
FROM DB2TX.SAMPLE T1, REPHANDLE T2,
TABLE (DB2TX.SEARCH_RESULT(CAST(T2.MYDOCHANDLE AS DB2TX.DB2TEXTH),
    "VisualAge")) AS T3
WHERE T1.H_1 = T3.HANDLE

```

SELECT NUMBER_OF_MATCHES, RANK, HANDLE では、3 つの項目がすべて戻されますが、3 つの項目を任意の組み合わせで指定することもできます。たとえば、RANK を省略して、それに関連した集中処理を回避することもできます。

HANDLE 値のみが必要な場合は、単に SELECT COUNT(*) を使用することができます。

第 7 章 管理

この章では、テキスト索引の保守方法、有用な情報の入手方法、およびテキスト準備プロセスを取り消す方法について説明します。

AS/400 オペレーション・ナビゲーターを使用しての管理コマンドの実行

IBM AS/400 Client Access Express の一部である AS/400 オペレーション・ナビゲーターを使用することによって、すべての AS/400 コマンドを Windows ワークステーションから実行することができます。このツールを使用して DB2 テキスト・エクステンダー管理コマンドを実行するには、以下の手順を実行してください。

1. Windows ワークステーションで AS/400 オペレーション・ナビゲーターを開始する。
2. コマンドを実行する AS/400 を選択し、マウス・ボタンを右クリックする。メニューから「実行 (Run)」コマンドを選択します。
3. 管理コマンドを入力する。たとえば以下のようになります。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET STATUS')
```

「了解 (OK)」ボタンをクリックします。

コマンドが正常に開始された後、「管理制御 (Management Control)」コマンドを強調表示し、オペレーション・ナビゲーターのメニューから「タスク・アクティビティ (Task Activity)」を選択することによって、経過と結果を確認することができます。

注

オペレーション・ナビゲーターのある AS/400 システムには、DB2 テキスト・エクステンダーのユーザー・プロファイルを使用してログオンしなければなりません。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

テキスト索引の保守

索引の保守タスクには、以下のようなものがあります。

- 索引の更新
- 索引の設定の変更
- 索引状況のリセット
- 索引イベントの削除
- 索引の再編成

これらのタスクは、いつでもどの順序でも実行することができます。

索引の更新

要約

実行時 周期的な索引付けが行われるのを待たずに、即時に索引を更新する必要がある場合。(周期的な索引付けに関する情報は、49ページの『テキスト列の使用可能化』を参照してください。)

コマンド

```
UPDATE INDEX
```

権限 テキスト・エクステンダーのユーザー権限。14ページの『ユーザーに権限を提供する』を参照してください。

次の例は、共通索引表に関する索引を更新します。

```
CALL PGM(QDB2TX/DB2TX) PARM('UPDATE INDEX db2tx.mytable')
```

次の例は、複数の索引表の列に関する索引を更新します。

```
CALL PGM(QDB2TX/DB2TX) PARM('UPDATE INDEX db2tx.mytable HANDLE commenthandle')
```

次の周期的な索引付けが自動的に行われるのを待たずに、索引を即時に更新する場合には、このコマンドを使用してください。この方法は、データベースに複数のテキスト文書を追加した直後にこれらの文書を検索する場合に便利です。

DB2 テキスト・エクステンダーは、挿入または変更が加えられたこの列 (または表の中のすべての列) のテキスト文書を索引付けし、削除された文書の用語を索引から除去します。索引に関連したログ表には、挿入、更新、および削除が行われた文書に関する情報が含まれています。

外部ファイルの索引の更新

文書を同じ絶対パス名をもつ新しいバージョンで置き換えたりすることなどにより、ユーザーに索引付けされている可能性のあるどの外部ファイル (53ページの『外部テキスト・ファイルの使用可能化』) の変更に関する情報も自動的にログ表に入れられることはありません。DB2 テキスト・エクステンダーは、ログ表内で、このようなファイルに対する更新をモニターすることはできません。このような更新は、DB2 UDB for iSeries の効力範囲外で起きるためです。

外部ファイルにおける更新が DB2 テキスト・エクステンダー索引に反映されるようにするには、以下のことを実行できます。

1. 「変更」項目に対応するハンドル列で、実際には何も行わない更新ステートメントを出すことによって、その「変更」項目をログ表に強制的に入れる。

```
UPDATE table
SET filehandlecol = filehandlecol
WHERE DB2TX.FILE(filehandlecol) = filename
```

ここで、*filename* は更新された外部ファイルの絶対パス名です。

2. UPDATE INDEX を実行して索引を更新し、外部ファイルに適用された変更を組み込む。

索引の設定の変更

要約

実行時 索引の更新頻度を変更する場合。

コマンド

CHANGE INDEX SETTINGS

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

このコマンドを使用して、索引の更新頻度を変更してください。

更新頻度

詳細は、35 ページの『索引の更新』を参照してください。更新頻度を指定しない場合は、現行の設定はこのまま変更されません。

テキスト文書の索引付けを即時に開始するかどうか、また次回いつ周期的な索引付けをスケジュールするかを決めるには、UPDATEINDEX キーワードを使用してください。このキーワードを使用しない場合は、現行の設定はこのまま変更されません。

例

最少 100 のテキスト文書が待ち行列にある場合に月曜から金曜の 12:00 または 15:00 に索引が更新されるよう索引の更新頻度を変更するには、次のようにします。

```
CALL PGM(QDB2TX/DB2TX) PARM('CHANGE INDEX SETTINGS db2tx.mytable
                             HANDLE      commenthandle
                             UPDATEFREQ min(100) d(1,2,3,4,5) h(12,15) m(00)')
```

索引の周期的な更新を停止するには次のようにします。

```
CALL PGM(QDB2TX/DB2TX) PARM('CHANGE INDEX SETTINGS db2tx.mytable
                             HANDLE      commenthandle
                             UPDATEFREQ none')
```

索引状況のリセット

要約

実行時 索引をこれ以上検索または更新できない場合

コマンド

RESET INDEX STATUS

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

索引の検索や更新ができない状態が生じる場合があります。82 ページの『索引状況の表示』では、このようなイベントが生じた場合の判別方法を説明しています。RESET INDEX STATUS は、索引を再度使用できるように再活動化します。

テキスト索引の保守

次の例は、共通索引表の索引の状況をリセットします。

```
CALL PGM(QDB2TX/DB2TX) PARM('RESET INDEX STATUS db2tx.mytable')
```

構文を使用すると、特定のテキスト列の索引状況をリセットできます。次の例は、複数索引表列の索引の状況をリセットします。

```
CALL PGM(QDB2TX/DB2TX) PARM('RESET INDEX STATUS db2tx.mytable  
HANDLE commenthandle')
```

索引イベントの削除

要約

実行時 索引のログ表内のメッセージがもう必要なくなった場合

コマンド

```
DELETE INDEX EVENTS
```

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

何らかの原因で索引の検索や更新ができない場合や、文書が索引付けできない場合、この状態は索引付けイベントと呼ばれます。索引付けイベントに関する情報は、その索引のログ表の中に保管されます。これは、問題の原因を判別する際に役立ちます。これらのメッセージは、必要なくなったときには削除できます。

次の例は、共通索引表からメッセージを削除します。

```
CALL PGM(QDB2TX/DB2TX) PARM('DELETE INDEX EVENTS db2tx.mytable')
```

構文を使用しても、特定のテキスト列の索引付けイベントを削除できます。次の例は、複数索引表列の索引に対するメッセージを削除します。

```
CALL PGM(QDB2TX/DB2TX) PARM('DELETE INDEX EVENTS db2tx.mytable  
HANDLE commenthandle')
```

索引の再編成

要約

実行時 GET INDEX STATUS で、索引を手操作で再編成しなければならないことが示される場合

コマンド

```
REORGANIZE INDEX
```

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

テキスト列が頻繁に更新される場合、索引の検索効率が下がる可能性があります。検索の効率を再び上げるには、索引を再編成する必要があります。DB2 テキスト・エクステンダーは索引を再編成する必要を認識し、バックグラウンドで自動的に再編成を行いますが、REORGANIZE INDEX を使用して手操作で再編成しなければ

ばならない場合もあります。索引を再編成する必要があるかどうかを調べるには、コマンド `GET INDEX STATUS` を使用します。

`REORGANIZE INDEX` の実行中には索引の検索は行えますが、索引の更新は行えません。

次の例は、共通索引表の索引を再編成します。

```
CALL PGM(QDB2TX/DB2TX) PARM('REORGANIZE INDEX db2tx.mytable')
```

次の例は、複数索引表列の索引を再編成します。

```
CALL PGM(QDB2TX/DB2TX) PARM('REORGANIZE INDEX db2tx.mytable  
HANDLE commenthandle')
```

有用な情報の入手

この説では、以下のものに関する情報を表示するためのコマンドについて説明します。

- データベース、表、列、およびファイルの使用可能状況
- 環境変数の設定
- テキスト構成設定値
- 索引状況
- エラー・イベント
- 索引設定値
- 列のテキスト設定値

使用可能状況情報の表示

要約

実行時 データベース、表、およびテキスト列、または外部ファイルの使用可能状況についての情報が必要な場合。

コマンド

`GET STATUS`

権限 テキスト・エクステンダーのユーザー権限。 14 ページの『ユーザーに権限を提供する』を参照してください。

次のように入力します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET STATUS')
```

次の内容は `GET STATUS` によって表示される出力の例です。これは、データベースの使用可能状況とデータベースに含まれる使用可能表、テキスト列、またはテキスト・ファイルの状況を示します。

```
Database is enabled for Text Extender

Table DB2TX.MYTABLE is enabled as a common-index table

Table DB2TX.MYTABLE is enabled as a common-index table
TextColumnName      HandleColumnName
```

```

-----
COMMENT          COMMENTHANDLE

Table DB2TX.TEST is enabled as a multi-index table
TextColumnName   HandleColumnName
-----
ABSTRACT1        ABSTRACT1HANDLE
ABSTRACT2        ABSTRACT2HANDLE
    
```

テキスト構成設定値の表示

要約

実行時 テキスト、索引、および処理情報のデフォルト設定値が必要な場合

コマンド

```
GET TEXT CONFIGURATION
```

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

これらの設定値については、10 ページの『テキスト構成設定値』で説明しています。これらを変更するには、11 ページの『テキスト構成の変更』を参照してください。

テキスト構成を表示するには、次のように入力します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET TEXT CFG')
```

次の内容は GET TEXT CONFIGURATION によって表示される出力の例です。この出力により、現行のテキスト構成設定値が示されます。

```

Coded character set ID      (CCSID) = 500
Language                    (LANGUAGE) = US_ENGLISH
Format                      (FORMAT) = TDS

Index type                  (INDEXTYPE) = LINGUISTIC
Update frequency           (UPDATEFREQ) = NONE
Index directory             (DIRECTORY) = user1/db2tx/indexes

Update index option        (UPDATEINDEX) = UPDATE
Commit count               (COMMITCOUNT) = 10 000
Tablespace name            (TABLESPACE) = TXLOG
    
```

索引状況の表示

要約

実行時 索引が検索可能か、更新可能かを判別する必要がある場合

コマンド

```
GET INDEX STATUS
```

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

索引の検索や更新ができない状態が生じる場合があります。このような状態では、メッセージが索引のログ表内に保管され、原因を判別するうえで役立ちます。したがって、索引の状況を検査して、参照可能なメッセージの有無を調べるために利用できます。

次の例は、共通索引表の索引の状況を表示します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET INDEX STATUS db2tx.mytable')
```

構文を使用すると、特定のテキスト列の索引状況を表示できます。次の例は、複数索引表列の索引の状況を入力します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET INDEX STATUS db2tx.mytable  
HANDLE commenthandle')
```

次の内容は GET INDEX STATUS によって表示される出力の例です。

```
Node 0
Search status           = Search available
Update status          = Update available
Reorg status           = started 13.55
Scheduled documents    = 0
Indexed documents      = 187000
Primary index documents = 130000
Secondary index documents = 57000
Error events           = No error events
```

索引がいくつかのノードに分割される場合には、ノード情報はノードごとに表示されます。

検索状況 (Search status)

指定されたハンドル列を使用して索引を検索できるかどうかを示します。検索が可能でない場合は、示されている理由コードをチェックし、その状況が起きた原因に関する詳細な情報がないかを調べてから、RESET INDEX STATUS を使用して、その索引を再度処理できるようにします。243 ページの『第 20 章 エラー・イベント理由コード』を参照してください。

更新状況 (Update status)

指定された表または列の索引を更新できるかどうかを示します。索引更新関数が利用可能でない場合は、示されている理由コードをチェックして、その状況が起きた原因に関する詳細な情報がないかを調べてから、RESET INDEX STATUS を使用して、その索引を再度処理できるようにします。

再編成状況 (Reorg status)

指定された表または列の索引を再編成できるかどうかを示します。再編成が可能でない場合は、示されている理由コードをチェックし、その状態が起きた原因に関する詳細な情報がないかを調べます。再編成ができない理由として一般的なものは、現在索引を更新中であることです。

スケジュールされた文書 (Scheduled documents)

索引付けのために (あるいは、索引からの削除のために) 待ち行列内にリストされた文書の数を示します。

索引付け文書 (Indexed documents)

すでに、スケジュール済み文書の待ち行列から索引付けされている文書の数を示します。

有用な情報の入手

1 次索引文書 (Primary index documents)

1 次索引内の文書の数を示します。

2 次索引文書 (Secondary index documents)

2 次索引内の文書の数を示します。

エラー・イベント (Error events)

索引のログ表内で使用可能なイベントの数を示します。この情報は、『エラー・イベントの表示』の説明にしたがって表示することができます。この情報が必要なくなったら、80 ページの『索引イベントの削除』の説明に従って、削除することができます。

エラー・イベントの表示

索引付けの間に、索引付け用にスケジュールされた文書が見つからないといった問題が生じた場合は、エラー・イベント と呼ばれるイベントが索引のログ表に書き込まれます。

イベント戻りコードについては、243 ページの『第 20 章 エラー・イベント理由コード』に説明してあります。

このエラー・イベントは、db2tx.LOGIXnnnnnn と呼ばれるログ表の視点でアクセスすることができます。ここで、IXnnnnnn は索引の名前で、カタログ・ビューから入手できます。

索引の名前を知るには、対話式 SQL セッションで以下のステートメントを実行します。

```
SELECT TABLENAME,  
       HANDLENAME,  
       INDEXNAME  
FROM   DB2TX.TEXTCOLUMNS
```

エラー・イベント視点のレイアウトは次のとおりです。

```
UPDATESTATUS  SMALLINT  
EVENTREASON   INTEGER  
EVENTMESSAGE  VARCHAR(1024)  
UPDATETIME    TIMESTAMP  
HANDLE        DB2TEXTH or DB2TEXTFH  
NODENUMBER    INTEGER
```

次の例は、索引ログから情報をアクセスする方法を示します。

```
SELECT EVENTREASON,  
       EVENTMESSAGE,  
       UPDATETIME,  
       HANDLE  
FROM   DB2TX.LOGIXNNNNNN
```


索引設定の表示

要約

実行時 索引の設定に関する情報が必要な場合

コマンド

GET INDEX SETTINGS

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

次の例は、共通索引表の索引の設定を入手します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET INDEX STATUS db2tx.mytable')
```

次の例は、複数索引表列の索引の索引設定値を入手します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET INDEX SETTINGS db2tx.mytable  
HANDLE commenthandle')
```

複数の索引表として使用可能になっている表の場合は、このコマンドは表の中のすべての使用可能なテキスト列の索引設定値を表示します。

次の内容は共通索引表に対する GET INDEX SETTINGS によって表示される出力の例です。複数索引表の出力は、索引ごとに同じ情報を示します。構文を使用すると、特定のテキスト列の索引設定を要求できます。

Current index settings:

```
Index type           (INDEXTYPE) = LINGUISTIC
Update index option  (UPDATEINDEX) = UPDATE
Update frequency     (UPDATEFREQ) = NONE
Node 0
Index directory      (DIRECTORY) = /home/user1/db2tx/indices
```

索引がいくつかのノードに分割される場合には、ノード情報は索引ディレクトリーごとに表示されます。

列のテキスト設定の表示

要約

実行時 列のテキスト設定に関する情報が必要な場合

コマンド

GET TEXT INFO

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

次の例は、共通索引表の索引のテキスト情報を入手します。

```
CALL PGM(QDB2TX/DB2TX) PARM('GET TEXT INFO db2tx.mytable')
```

次の例は、複数索引表の索引のテキスト情報を入手します。

有用な情報の入手

```
CALL PGM(QDB2TX/DB2TX) PARM('GET TEXT INFO db2tx.mytable HANDLE
commenthandle')
```

構文を使用すると、表名とハンドル列名を指定できます。

表名のみをコマンド中で指定した場合は、その表の中の使用可能な各列のテキスト情報が表示されます。ハンドル列名も指定した場合は、その列のテキスト情報のみが表示されます。

次の例は、複数索引表に対するこのコマンドによって表示される内容を示したものです。

```
Text information for column ABSTRACT1
with handle column ABSTRACT1HANDLE:
Coded character set ID (CCSID) = 500
Language              (LANGUAGE) = US_ENGLISH
Format                (FORMAT) = TDS
```

```
Text information for column ABSTRACT2
with handle column ABSTRACT2HANDLE:
Coded character set ID (CCSID) = 500
Language              (LANGUAGE) = US_ENGLISH
Format                (FORMAT) = TDS
```

DB2 テキスト・エクステンダー・カタログ・ビューの処理

DB2 テキスト・エクステンダーはサブシステムごとに DB2TX.TEXTINDEXES というカタログ・ビューを作成し、保守します。これは、ENABLE SERVER の実行時に作成されます。カタログ・ビューには、DB2 テキスト・エクステンダーで使用可能にされている表および列に関する情報が含まれています。

表、列、または外部ファイルが使用可能であれば必ず、DB2TX.TEXTINDEXES 内に新規項目が作成されます。項目は、CHANGE INDEX SETTINGS コマンドを使用して索引設定を修正するたびに更新されます。列または表が使用不能になると、項目は削除されます。

カタログ・ビュー中のデータは、通常の SQL 照会機能によって利用できます。しかし、カタログ・ビューの修正は、通常の SQL データ操作コマンドでは行えません。カタログ・ビューを明示的に作成または消去することはできません。表 6 は、カタログ・ビューの内容を示します。

表 6. DB2 テキスト・エクステンダーのカタログ・ビュー

列名	データ・タイプ	ヌル可能	説明
TABLESCHEMA	CHAR(8)	不可	この項目を適用する表のスキーマです。
TABLENAME	VARCHAR(18)	不可	この項目を適用する表の名前です。
COLUMNNAME	VARCHAR(18)	可	この表の中で使用可能になった列の名前です。表が使用可能になっても、すべての列が使用不能であればこの値はヌルになります。
HANDLENAME	VARCHAR(18)	可	ハンドル列の名前。TABLESCHEMA.TABLENAME 表の中に使用可能な列がない場合は、この値はヌルになります。
INDEXNAME	CHAR(8)	不可	テキスト表またはテキスト列を使用可能にする際に作成されるテキスト索引の名前です。

DB2 テキスト・エクステンダー・カタログ・ビューの処理

表 6. DB2 テキスト・エクステンダーのカタログ・ビュー (続き)

列名	データ・タイプ	ヌル可能	説明
LOGTABLE	VARCHAR(18)	不可	INDEXNAME 索引のログ表の名前です。表 DB2TX.LOGTABLE には、テキスト索引の次の更新用にスケジュールされたテキスト文書がどれであるかとエラー・イベントについての情報が含まれます。
INDEXTYPE	VARCHAR(30)	不可	索引タイプ: LINGUISTIC、PRECISE、NGRAM
MINIMUM	INTEGER	可	索引更新の実行前に必要な、索引の更新要求回数の最小数です。 35 ページの『索引の更新』を参照してください。更新頻度が NONE に設定されている場合は、この値はヌルになります。
DAYS	VARCHAR(15)	可	更新をスケジュールする日です。 35 ページの『索引の更新』を参照してください。更新頻度が NONE に設定されている場合は、この値はヌルになります。
HOURS	VARCHAR(75)	可	索引更新をスケジュールする時間です。 35 ページの『索引の更新』を参照してください。更新頻度が NONE に設定されている場合は、この値はヌルになります。
MINUTES	VARCHAR(185)	可	更新をスケジュールする分です。 35 ページの『索引の更新』を参照してください。更新頻度が NONE に設定されている場合は、この値はヌルになります。
INDEXDIRECTORY	VARCHAR(254)	不可	テキスト索引を保管する、ファイル・システム中のディレクトリーの名前です。
UPDATEONCREATE	VARCHAR(10)	不可	値 “update” または “noupdate”。 ENABLE TEXT TABLE または ENABLE TEXT COLUMN の UPDATEINDEX オプションで、または最後の CHANGE INDEX SETTINGS で指定された値です。
COMMONINDEX	VARCHAR(4)	不可	TABLESCHEMA.TABLENAME 表が共通索引表である場合は、「可」です。 TABLESCHEMA.TABLENAME 表が複数の索引表である場合は、「不可」です。
CCSID	SMALLINT	可	テキスト列を使用可能にするコマンドで指定した、TEXTCOLUMN テキスト列の CCSID です。TEXTCOLUMN がヌルである場合は、この値はヌルになります。
LANGUAGE	VARCHAR(30)	可	TEXTCOLUMN テキスト列を処理する際に使用するディレクトリーの名前です。 TEXTCOLUMN がヌルである場合は、この値はヌルになります。
FORMAT	VARCHAR(30)	可	TEXTCOLUMN テキスト列に指定した形式です。TEXTCOLUMN がヌルである場合は、この値はヌルになります。
FUNCTIONSCHEMA	CHAR(8)	可	ENABLE TEXT COLUMN 中で FUNCTION オプションを使用して指定したアクセス関数のスキーマです。FUNCTION オプションが指定されていない場合は、この値はヌルになります。
FUNCTIONNAME	VARCHAR(18)	可	ENABLE TEXT COLUMN 中で FUNCTION オプションを使用して指定したアクセス関数の名前です。FUNCTION オプションが指定されていない場合は、この値はヌルになります。

DB2 テキスト・エクステンダー・カタログ・ビューの処理

表 6. DB2 テキスト・エクステンダーのカタログ・ビュー (続き)

列名	データ・タイプ	ヌル可能	説明
PROTOTYPEHANDLE	VARCHAR(60)	可	パフォーマンス関数で使用するハンドル。これには、テキスト列全体に共通な索引名だけが含まれています。
INDEXOPTION	VARCHAR(30)	可	索引作成時のオプション: CASE_ENABLED
INDEXPROPERTY	VARCHAR(30)	可	索引作成時のプロパティ: SECTIONS_ENABLED
NODENUMBER	INTEGER	不可	表区画のノード番号。

テキスト準備処理の取り消し

DB2 テキスト・エクステンダーで使用するためにテキストを作成すると、特定の管理上の変更が行われます。このセクションでは、このようなプロセスを取り消す際に役立つ機能について説明します。

テキスト列を使用不可にする

要約

実行時 テキスト列でのテキスト検索をこれ以上行わない場合

コマンド

DISABLE TEXT COLUMN

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

例:

```
CALL PGM(QDB2TX/DB2TX) PARM('DISABLE TEXT COLUMN db2tx.mytable  
HANDLE commenthandle')
```

テキスト列を使用不可にすると、以下のことが起こります。

- 複数索引表の場合、すなわち列に独自のテキスト索引とログ表がある場合は、索引、ログ表、およびログ表トリガーが削除されます。
- 共通索引の場合、すなわち、1 つの索引がすべてのテキスト列によって共用される場合は、この列の文書の用語が共通索引から除去されます。これが表の中に残された唯一の使用可能テキスト列である場合は、索引、ログ表、およびログ表トリガーが削除されます。

テキスト・ファイルを使用不可にする

要約

実行時 外部テキスト・ファイル内でテキスト検索をこれ以上行わない場合

コマンド

DISABLE TEXT FILES

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

例:

```
CALL PGM(QDB2TX/DB2TX) PARM('DISABLE TEXT FILES db2tx.mytable
HANDLE commenthandle')
```

外部テキスト・ファイルを使用不可にすると、以下のことが起こります。

- このハンドル列の索引が削除されます。
- ログ表とトリガーが削除されます。

テキスト表を使用不可にする

要約

実行時 テキスト表でのテキスト検索をこれ以上行わない場合

コマンド

DISABLE TEXT TABLE

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

例:

```
CALL PGM(QDB2TX/DB2TX) PARM('DISABLE TEXT TABLE db2tx.mytable')
```

テキスト表を使用不可にすると、以下のことが起こります。

- 表のテキスト列に関する共通索引がある場合は、この索引が削除される。あるいは、テキスト列ごとに個別の索引がある場合は、テキスト列に関する索引がすべて削除される。
- どのテキストを索引付けするかを自動的に記録するのに使用される共通ログ表が削除される。あるいは、テキスト列ごとに個別のログ表がある場合は、すべてのログ表が削除される。
- ログ表の保守に使用されるトリガーが削除される。
- ハンドル列の内容がヌル設定になる。

サーバーを使用不可にする

要約

実行時 このデータベースでのテキスト検索をこれ以上行わない場合

コマンド

```
DISABLE SERVER for DB2TEXT
```

権限 テキスト・エクステンダーのユーザー権限。14 ページの『ユーザーに権限を提供する』を参照してください。

接続しているデータベースを使用不可にするには、次のように入力します。

```
CALL PGM(QDB2TX/DB2TX) PARM('DISABLE SERVER for DB2TEXT')
```

データベースを使用不可にすると、以下のオブジェクトが削除されます。

- サーバーを使用可能にした際に作成された、DB2 テキスト・エクステンダーのカタログ・ビュー
- DB2 テキスト・エクステンダーの SQL 関数 (UDF) の宣言、およびこのデータベースの DB2 テキスト・エクステンダー特殊タイプ (UDT)
- このデータベースのテキスト表またはテキスト列に関連したすべての索引
- どのテキスト文書を索引付けするかを自動的に記録するために使用されるログ表と、それらを保守するのに使用されるトリガー

ハンドル列は特殊タイプの 1 つですが、削除できないため、特殊タイプの中には削除されないものもあります。

この呼び出しによって影響を受けるのは、接続しているデータベースのみです。

第 8 章 検索およびブラウズのための API 関数の使用法

この章では、DB2 テキスト・エクステンダー API の検索関数とブラウズ関数の使用方法について説明します。これらの関数の詳細については、175 ページの『第 13 章 検索およびブラウズのための API 関数』を参照してください。API 関数を使用したプログラムの例は、195 ページの『第 14 章 サンプル API プログラム』に示します。同じ章の中で、サンプル・ブラウズ関数 `DesBrowseDocument` について説明しています。

ヒント

検索する前には、31 ページの『検索のタイプ』をお読みください。使用する索引タイプによっては、検索結果が異なる場合があります。GET INDEX SETTINGS を使用して、検索するテキストと関連したテキスト索引タイプを調べてください。

アプリケーションのセットアップ

DB2 テキスト・エクステンダー API を使用するアプリケーション・プログラムは、DB2 CLI アプリケーションです。その理由は、API 関数の中にはデータベース接続ハンドルを入力用として必要とするものがあるためです。したがって、DB2 CLI アプリケーションに対して考慮すべき規則が、DB2 テキスト・エクステンダー API を使用するアプリケーションにも適用されます。

使用中のアプリケーションに、DB2 テキスト・エクステンダーの製品ライブラリー「QDB2TX」の「H」ファイルにある `des_ext.h` を組み込んでください。コンパイル・ステップで、コンパイル・オプション「`_OS400_`」も設定しなければなりません。

DB2 テキスト・エクステンダー API を用いてアプリケーション・プログラムを使用するには、そのプログラムを API にリンクしてください。

アプリケーションのリンク

サービス・プログラム `desclapi` を使用中のアプリケーションにリンクする必要があります。このサービス・プログラムは、DB2 テキスト・エクステンダー製品ライブラリー QDB2TX にあります。

API 関数の概説

以下に示すのは、検索関数およびブラウズ関数です。最初の関数は検索関数で、残りはブラウズ関数です。

- DesGetSearchResultTable
- DesGetBrowseInfo
- DesStartBrowseSession

API 関数の概説

DesOpenDocument
DesGetMatches
DesCloseDocument
DesEndBrowseSession
DesFreeBrowseInfo

ヒント

API 関数の多くには接続ハンドル (hdbc) が必要です。このハンドルは SQLConnect 関数を使用して指定しなければなりません。それによって組み込み SQL プログラムから DB2 テキスト・エクステンダーを呼び出せなくなることはありません。「DB2 CLI の手引きおよび解説書」では、CLI ステートメントと組み込み SQL ステートメントとの組み合わせ方法が説明されています。

API 関数を使用して、以下に示す作業を実行することができます。

• テキストの検索

このシナリオでは、検索関数 DesGetSearchResultTable のみが必要です。この関数は、入力用として検索引き数とハンドル列名をとります。そして、テキストを検索し、事前に作成されていた結果表の中へ文書に関する情報を入れます。

この関数は、93 ページの『テキストの検索』で詳しく説明されています。
195 ページの『第 14 章 サンプル API プログラム』も参照してください。

• テキストのブラウズ

示されている順序に従って、以下の関数を使用してください。

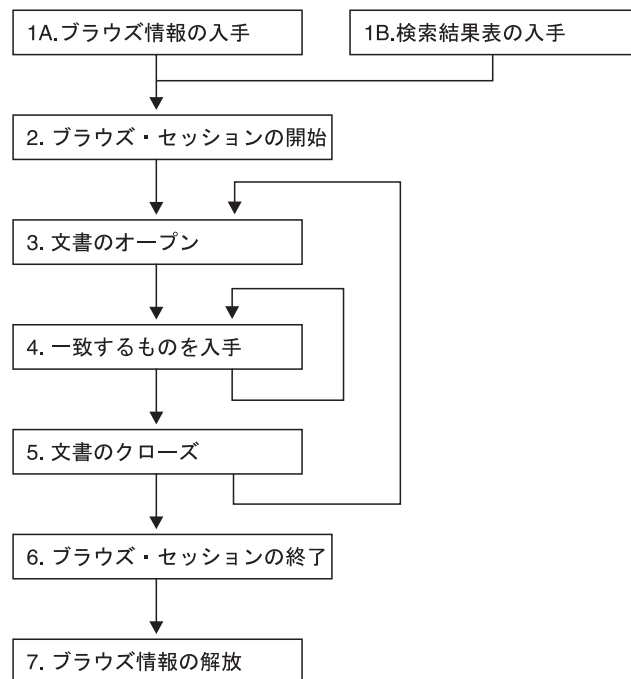


図 15. API 関数を使用する順序

これらの関数は、強調表示情報を入手した後、ブラウザ・セッションを開始してテキスト文書を表示し、そこにある用語を強調表示します。

ブラウザ・セッションでは、同じ強調表示情報を使用している文書をさらにオープンして表示することができます。以上の関数は、94 ページの『テキストのブラウザ』でさらに詳しく説明されています。

テキストの検索

テキスト検索のための API 関数は、DesGetSearchResultTable 関数の 1 つだけです。

検索結果表の入手 (DesGetSearchResultTable)

DesGetSearchResultTable 関数は、特定のテキスト列中にあるテキスト文書を検索するための検索引き数を受け取り、結果を表中に保管します。結果の表には、検出された個々の文書のハンドルが含まれます。また、指定する検索オプションによっては、ランク情報と一致数を含めることもできます。

RANK および NO_OF_MATCHES 関数を使って、この情報を入手することもできます。次に例を示します。

```
INSERT INTO RESULT
SELECT COMMENTHANDLE,
      RANK(COMMENTHANDLE, "stored procedures"),
      NO_OF_MATCHES(COMMENTHANDLE, "stored procedures")
FROM DB2TX.SAMPLE
WHERE CONTAINS(COMMENTHANDLE, "stored procedures") = 1
```

DesGetSearchResultTable は基礎表にしか使用できませんが、テキスト専用照会の場合には DB2 テキスト・エクステンダー関数を使用するよりも速く処理されます。つまり、DB2 テキスト・エクステンダー・サーバーに直接アクセスしてランク情報と一致数を入手し、検出された一致文書の数の回数分のみループします。一方、UDF の例では、表中の行ごとに CONTAINS 関数が呼び出されてから、修飾行ごとに RANK および NO_OF_MATCHES 関数が呼び出されます。検出文書にはそれぞれ、3 つの別々の検索が行われます。

入力

入力パラメーターは以下のとおりです。

- データベース接続用のハンドル
- 検索される表
- 検索されるテキスト列に関連したハンドル列の名前
- 検索引き数
- 検索オプション
- ブラウズ・オプション (ブラウズ情報を戻す)
- 結果が保管される表の名前

出力

ブラウズ・オプションを指定すると、この関数はブラウズ情報に対するポインターを戻します。

テキストのブラウズ

92 ページの図 15 の中のこの関数のグループは、強調表示されている用語を検出します。次に、ブラウズ・セッションを開始して文書をオープンし、ご使用のブラウザーを呼び出すアプリケーション・プログラムにより分析されるデータ・ストリームの書式で一致情報を入手します。

ブラウズ情報の入手 (DesGetBrowseInfo)

DesGetBrowseInfo 関数は、検索引き数とハンドルを受け取ります。そして DesStartBrowseSession が必要とするブラウズ情報に対するポインターを戻します。ブラウズ情報には、強調表示される全用語のリストが含まれます。

ブラウズ情報を入手する別の方法として、DesGetSearchResultTable 関数に Browse オプションを指定する方法もあります。

入力

入力パラメーターは以下のとおりです。

- データベース接続用のハンドル
- ハンドル
- 検索引き数

出力

この関数は、ブラウズ情報に対するポインターを戻します。

ブラウズ・セッションの開始 (DesStartBrowseSession)

DesStartBrowseSession 関数はブラウズ・セッションを開始し、テキスト文書のブラウズおよびその一致情報の強調表示に必要な環境を設置します。この関数は、DesGetBrowseInfo または DesGetSearchResultTable のいずれかより、ブラウズ情報に対するポインターを受け取り、他のブラウズ関数が使用するためのブラウズ・セッション・ハンドルを戻します。

入力

入力パラメーターは以下のとおりです。

- DesGetBrowseInfo または DesGetSearchResultTable からのブラウズ情報に対するポインター
- ユーザー ID
- パスワード

出力

この情報は、ブラウズ・セッション・ハンドルを戻します。

文書のオープン (DesOpenDocument)

DesOpenDocument 関数はブラウズ・セッション・ポインター、ハンドル、および検出された用語を強調するために使用される言語処理タイプを示すオプションである DES_FAST または DES_EXTENDED を受け取ります。205 ページの『ステージ 2: 拡張マッチング』を参照してください。DES_FAST は、辞書を使用せずに基本テ

キスト分析を使用して、どの用語を強調表示するかを判別します。
DES_EXTENDED は、拡張一致情報を使用します。

DesOpenDocument は、文書テキストおよび強調表示情報を入手するためのハンドルに対応するテキスト文書を作成し、DesGetMatches の反復呼び出しに使用される文書ハンドルを戻します。

入力

入力パラメーターは以下のとおりです。

DesStartBrowseSession からのブラウズ・セッション・ハンドル
テキスト・ハンドル

一致オプション: DES_FAST または DES_EXTENDED

出力

この関数は、DesGetMatches および DesCloseDocument が使用する文書ハンドルを戻します。

一致情報の入手 (DesGetMatches)

DesGetMatches 関数は、文書ハンドルに記述されたテキスト文書の強調表示情報に対するポインターを戻します。強調表示情報はデータ・ストリームです。これには、テキスト文脈 (最低 1 つの段落) と、その文脈内のテキストを強調表示させるための情報が含まれています。データ・ストリームについては、183 ページの『データ・ストリーム構文』で説明します。アプリケーション・プログラムは、データ・ストリームを解析し、ユーザー独自のブラウザーを使用して、それを処理することができます。

DesGetMatches は、データ・ストリームの一部だけを戻します。これは、その出力構造体の一部の長さを示します。

DesGetMatches を連続して呼び出すと、テキスト文書全体の内容を入手できます。テキスト文書の最後に達すると標識が戻されます。

入力

入力パラメーターは以下のとおりです。

ブラウズ・セッション・ハンドル

DesOpenDocument からの文書ハンドル

出力

この関数は、データ・ストリームの一部およびその長さを含んだ構造体に対するポインターを戻します。

文書のクローズ (DesCloseDocument)

DesCloseDocument 関数は、DesOpenDocument がオープンしたテキスト文書をクローズし、文書テキストおよび強調表示情報の戻り時に割り当てられるストレージを解放します。

テキストのブラウズ

入力

入力パラメーターは以下のとおりです。

ブラウズ・セッション・ハンドル

DesOpenDocument からの文書ハンドル

出力

ありません。

ブラウズ・セッションの終了 (DesEndBrowseSession)

DesEndBrowseSession 関数は、DesStartBrowseSession が開始したブラウズ・セッションを終了し、ブラウズ・セッションに割り当てられたストレージを解放します。

入力

入力パラメーターは以下のとおりです。

ブラウズ・セッション・ハンドル

出力

ありません。

ブラウズ情報の解放 (DesFreeBrowseInfo)

DesFreeBrowseInfo 関数は、DesGetBrowseInfo によってブラウズ情報に割り当てられたストレージを解放します。

入力

入力パラメーターは以下のとおりです。

ブラウズ情報に対するポインター

出力

ありません。

第 2 部 参照情報

第 9 章 クライアント用のテキスト準備コマンドと管理コマンド

コマンド	用途	ページ
CHANGE INDEX SETTINGS	索引の特性を変更する。	101
CHANGE TEXT CONFIGURATION	テキスト構成設定値を変更する。	103
CONNECT	サブシステムに接続する。	106
DELETE INDEX EVENTS	索引イベントをログ表から削除する。	107
DISABLE SERVER FOR DB2TEXT	サーバーを DB2 テキスト・エクステンダーで使用できないようにする。	108
DISABLE TEXT COLUMN	テキスト列を DB2 テキスト・エクステンダーで使用できないようにし、それに関連した索引を削除する。	109
DISABLE TEXT FILES	テキスト・ファイルを DB2 テキスト・エクステンダーで使用できないようにし、それらに関連した索引を削除する。	110
DISABLE TEXT TABLE	表を DB2 テキスト・エクステンダーで使用できないようにし、表に関連した索引を削除する。	111
ENABLE SERVER FOR DB2TEXT	DB2 テキスト・エクステンダーで使用するためにサーバーを準備する。	112
ENABLE TEXT COLUMN	DB2 テキスト・エクステンダーで使用するためのテキスト列を作成し、列に対して個別のテキスト索引を作成する。	113
ENABLE TEXT FILES	DB2 テキスト・エクステンダーで使用するためのテキスト・ファイルを作成し、そのファイルに対して個別のテキスト索引を作成する。	120
ENABLE TEXT TABLE	表の共通テキスト索引を作成する。	123
GET INDEX SETTINGS	索引の特性を表示する。	127
GET INDEX STATUS	索引のための状況情報を表示する。	129
GET STATUS	データベース、表、および列の使用可能状況を表示する。	130
GET TEXT CONFIGURATION	テキスト構成設定値を表示する。	131
GET TEXT INFO	テキスト列のテキスト情報を表示する。	132
QUIT	DB2 テキスト・エクステンダー・コマンド行プロセッサ・モードを終了する。	133
REORGANIZE INDEX	索引を再編成して検索の効率を高める。	134
RESET INDEX STATUS	索引状況を再度使用できるようにリセットする。	135
UPDATE INDEX	テキスト索引を更新する。	136

この章では、クライアント用のテキスト準備コマンドと管理コマンドの構文について説明します。これらのコマンドの使用方法については、43 ページの『第 5 章 テキストを検索可能にする』および 77 ページの『第 7 章 管理』で説明しています。

これらのコマンドを使用する前に、コマンド `db2tx` を入力して DB2 テキスト・エクステンダー・コマンド行プロセッサを開始してください。これにより対話式入力モードに入り、このモードでは後続のコマンドはすべて DB2 テキスト・エク

テンダー・コマンドとして解釈されます。あるいは、77 ページの『AS/400 オペレーション・ナビゲーターを使用しての管理コマンドの実行』で説明されているように、DB2 テキスト・エクステンダー管理コマンドを実行することができます。

このモードを終了するには QUIT と入力します。

注

DB2TX プログラムを開始するには、テキスト・エクステンダーの管理者権限またはテキスト・エクステンダーのユーザー権限が必要です。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

CHANGE INDEX SETTINGS

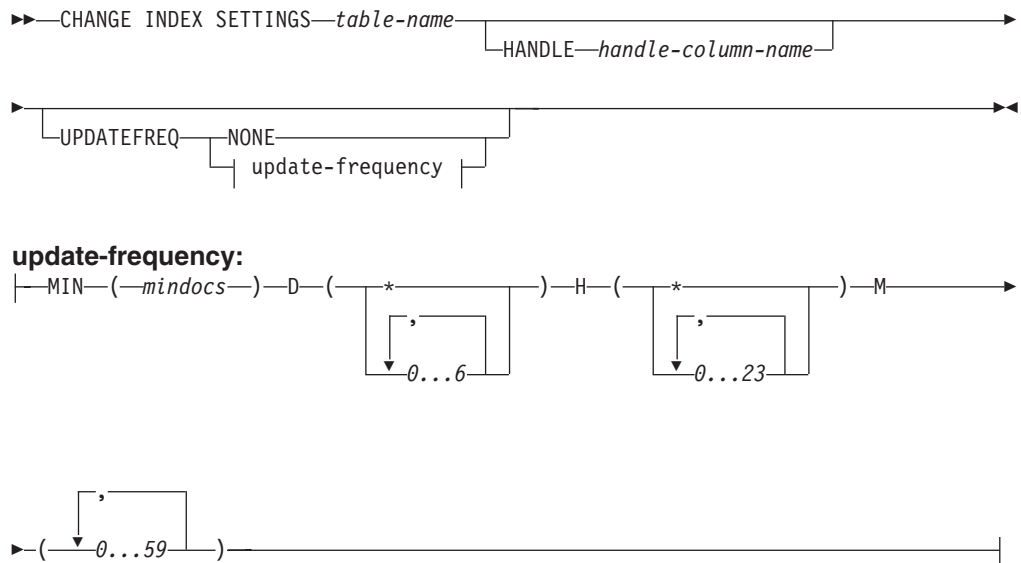
このコマンドは、データベースが使用可能になった**後**で、索引の特性を変更します。

データベースを最初に使用可能にする際に使用されるデフォルト設定値を変更するには、103 ページの『CHANGE TEXT CONFIGURATION』を使用します。

許可

表に関する所要のオブジェクト権限が付与されていなければなりません。詳しくは、14 ページの『DB2 テキスト・エクステンダーの SQL 表を準備する』を参照してください。

コマンド構文



コマンド・パラメーター

table-name

変更する索引更新頻度を持つテキスト列が含まれる、接続データベース内のテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

HANDLE handle-column-name

変更する索引更新頻度を持つハンドル列の名前。テキスト列が独自の索引を持っている場合、すなわち ENABLE TEXT COLUMN コマンドを使用して索引を作成した場合に、このパラメーターは必須です。

ただし、ENABLE TEXT TABLE コマンドを使用して索引を作成した場合、すなわち表がすべてのテキスト列について 1 つのテキスト索引を持っている場合には、このキーワードは無視されます。

UPDATEFREQ update-frequency

更新を実行するとき、およびログ表で待ち行列に入れるテキスト文書の最少数に関する索引の更新頻度です。所定の日時にログ表中に十分な数のテキスト文書がなければ、索引は更新されません。

CHANGE INDEX SETTINGS コマンド

構文については、35 ページの『索引の更新』で説明しています。

NONE 以後索引の更新を行いません。これは、これ以上変更を行わないテキスト列のためのものです。

UPDATEFREQ キーワードを指定しない場合は、頻度設定はそのまま変更されません。

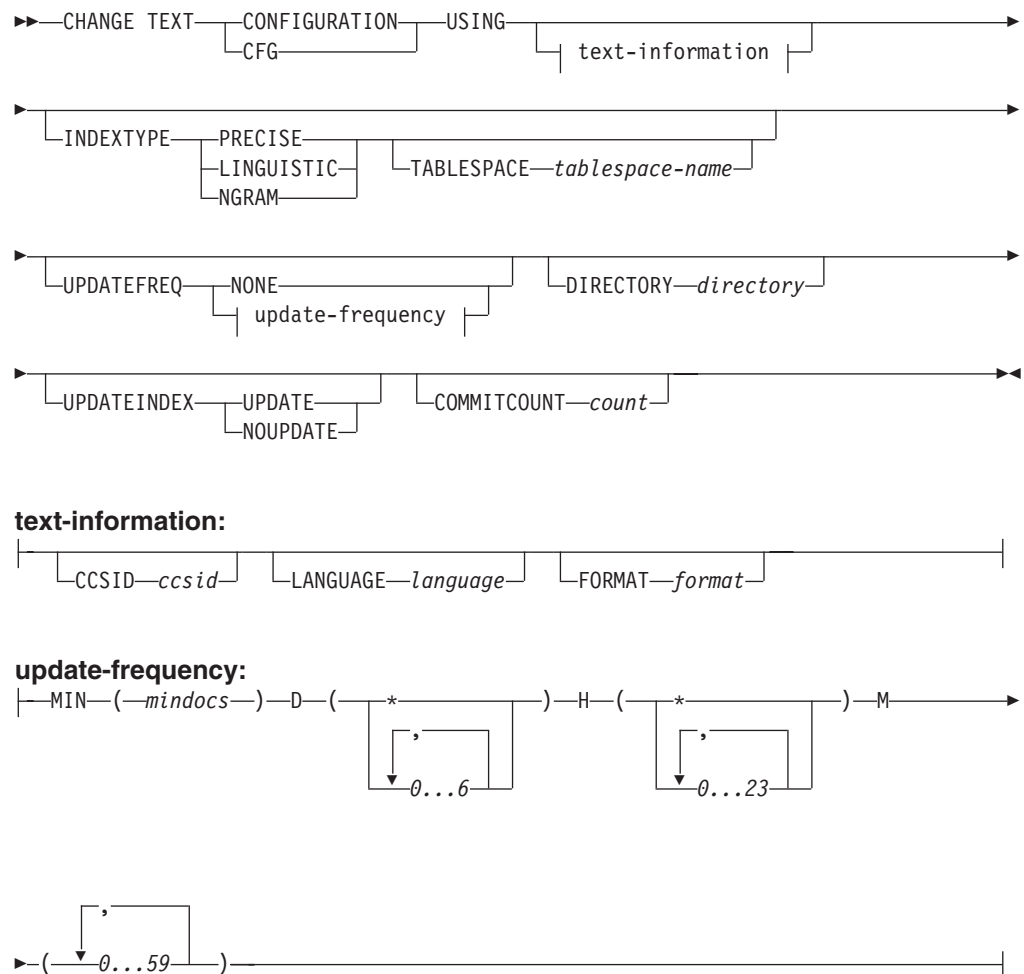
CHANGE TEXT CONFIGURATION

このコマンドは、データベースを使用可能にする際に使用されるテキスト構成のデフォルト設定値を変更します。これらはテキスト構成 設定値です。DB2 テキスト・エクステンダーがインストールされた際のこの表の初期テキスト構成設定値については、10 ページの『テキスト構成設定値』で説明しています。

許可

ありません。

コマンド構文



コマンド・パラメーター

INDEXTYPE

デフォルト索引タイプを変更するには、次のいずれかを選択します。詳細については、31 ページの『検索のタイプ』を参照してください。

PRECISE

用語はテキスト文書に出てくるとおりに索引付けされ、検索されます。

CHANGE TEXT CONFIGURATION コマンド

LINGUISTIC

用語は索引付けの前に言語的な処理が行われます。また、検索用語も検索開始前に言語的な処理が行われます。

NGRAM

用語は、辞書を使用するのではなく、一連の文字を解析することによって索引付けされます。索引付けしている文書に DBCS 文字が含まれている場合は、この索引タイプが必須ですが、SBCS 文書に対して Ngram 索引を使用することもできます。

TABLESPACE tablespace-name

既存の表スペースの名前を指定してください。表スペースは、DB2 テキスト・エクステンダーにより作成される索引固有の表 (ログ表など) を保持するために使用されます。ラージ表の場合には、別個の表スペースを使用してください。表スペースを指定しない場合には、表は DB2 デフォルト表スペースに作成されます。

UPDATEFREQ update-frequency

更新を実行するとき、およびログ表で待ち行列に入れるテキスト文書の最少数に関する索引の更新頻度です。所定の日時にログ表中に十分な数のテキスト文書がなければ、索引は更新されません。

構文については、35 ページの『索引の更新』で説明しています。

NONE 以後索引の更新を行いません。これは、これ以上変更を行わないテキスト列のためのものです。

DIRECTORY directory

テキスト索引を保管するディレクトリーです。

UPDATEINDEX

UPDATEFREQ で設定された次の周期的索引付けを待たず、このオプションを使用するコマンドの完了後すぐに、テキスト文書を索引付けするかどうかを決めるキーワードです。これらのコマンドは ENABLE TEXT COLUMN、および ENABLE TEXT FILES です。

UPDATE

このコマンドの完了後すぐに、テキスト文書の索引付けが行われます。

NOUPDATE

このコマンドの UPDATEFREQ またはテキスト構成設定値によって指定された更新頻度設定値に設定された時間に、索引付けが行われます。

COMMITCOUNT count

挿入または更新の数を示す 500 ~ 1 000 000 の値で、この値を超えると、DB2 UDB for iSeries により中間コミット・ステートメントが出されなければなりません。この値を使用することにより、ラージ表、列、または多数の外部ファイル用のログ・スペースが不足するという状態が起きるのを避けることができます。

CCSID ccsid

テキスト文書を索引付けする際に使用するコード化文字セット ID です。

CHANGE TEXT CONFIGURATION コマンド

サポートされる CCSID に関する情報については、26 ページの『CCSID』を参照してください。

LANGUAGE language

テキストを書き込む言語です。これにより、テキスト文書を索引付けするときやテキスト文書内を検索するとき使用される辞書が決まります。

197 ページの『第 15 章 基本語形索引と一致語句索引の言語的な処理』で、辞書の使用方法を説明しています。

サポートされる言語は、40 ページの表 4 に記載されています。

FORMAT format

保管されるテキスト文書のタイプで、WordPerfect または ASCII など。

DB2 テキスト・エクステンダーでは、文書を索引付けするときに、この情報が必要です。サポートされる文書形式は、23 ページの『サポートされる文書形式』に列挙されています。

使用上の注意

特定のデータベースが使用可能にされた後で、これらの設定値を変更するには、101 ページの『CHANGE INDEX SETTINGS』を使用します。

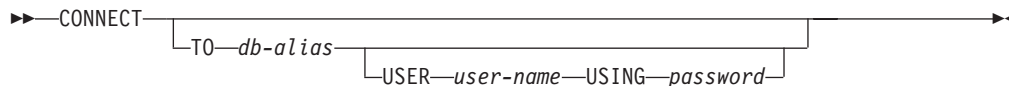
CONNECT

このコマンドは DB2 テキスト・エクステンダーをサブシステムに接続します。

許可

ありません。

コマンド構文



コマンド・パラメーター

TO database

接続するデータベース。

USER user-name

ユーザー名を指定しない場合、オペレーティング・システムから検索されます。

USING password

パスワードは、ユーザー名を指定した場合に限り必須です。

CONNECT no operand

オペランドを指定せず、サブシステムに接続されていない場合は、CONNECT はその時点においてアクティブな名前・スペースに暗黙に接続します。オペランドを指定せず、サブシステムに接続している場合は、CONNECT は現行のサブシステムに関する情報を表示します。

使用上の注意

1 度に接続できるサブシステムは 1 つのみであり、これを現行のサブシステムといいます。対話モードでは、別の CONNECT TO ステートメントでサブシステムを変更するか、または QUIT コマンドを発行するまで接続は継続されます。コマンド行モードでは、CONNECT コマンドは無効となります。

DELETE INDEX EVENTS

このコマンドは、所定のハンドル列または表用の索引のログ表から、索引付けイベントを削除します。

許可

表に関する所要のオブジェクト権限が付与されていなければなりません。詳しくは、14 ページの『DB2 テキスト・エクステンダーの SQL 表を準備する』を参照してください。

コマンド構文

```
▶▶—DELETE INDEX EVENTS—table-name—┬──────────────────────────┬──────────▶▶  
                                └──HANDLE—handle-column-name──┘
```

コマンド・パラメーター

table-name

ログ表からエラー・イベントが削除される、接続しているデータベース内のテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

HANDLE handle-column-name

ログ表からメッセージが削除されるハンドル列の名前。

使用上の注意

ハンドル列名を指定した場合は、その列の索引付けイベントのみが削除されます。

ログ表をクリーンアップする場合、索引イベントの削除を実行するのは、イベントの原因を調べ、可能であればエラーの原因を除去してからにしてください。

DISABLE SERVER FOR DB2TEXT

このコマンドは、DB2 テキスト・エクステンダーにより実行されたサーバーに関するあらゆる準備作業をリセットし、すべてのテキスト表を DB2 テキスト・エクステンダーで使用できないようにします。

許可

ありません。

コマンド構文

▶▶—DISABLE SERVER FOR DB2TEXT—▶▶

コマンド・パラメーター

ありません。

使用上の注意

このコマンドは、接続しているサーバーをリセットし、以後これを DB2 テキスト・エクステンダーで検索できないようにします。すなわち、サーバー内のすべての DB2 テキスト・エクステンダーのテキスト表とテキスト列を使用できないようにします。これには、DB2 テキスト・エクステンダーのテキスト表およびテキスト列を使用可能にするためにサーバー内で行われたすべての変更が含まれます。すべてのテキスト列および外部ファイルがリセットされ、すべての関連するテキスト索引が削除され、サーバー内の DB2 テキスト・エクステンダー・カタログ・ビュー TEXTCOLUMNS およびすべての DB2 テキスト・エクステンダー・トリガーが削除されます。

DISABLE TEXT COLUMN

このコマンドは、テキスト列を DB2 テキスト・エクステンダーで使用できないようにします。

許可

表に関する所要のオブジェクト権限が付与されていないとなりません。詳しくは、14 ページの『DB2 テキスト・エクステンダーの SQL 表を準備する』を参照してください。

コマンド構文

```
►►—DISABLE TEXT COLUMN—table-name—HANDLE—handle-column-name—◄◄
```

コマンド・パラメーター

table-name

使用不可にする列が含まれた、接続しているデータベース内のテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

HANDLE handle-column-name

DB2 テキスト・エクステンダーで使用できないようにするハンドル列の名前。

使用上の注意

索引が削除されます。

ハンドル列内の変更 (挿入、更新、および削除) を記録するログ表が削除されます。

ログ表に項目を書き込むトリガーが削除されます。

ハンドル列は変更されません。

DISABLE TEXT FILES

このコマンドは、DB2 テキスト・エクステンダーに使用される外部ファイルの集合を使用不可にします。

許可

表に関する所要のオブジェクト権限が付与されていなければなりません。詳しくは、14 ページの『DB2 テキスト・エクステンダーの SQL 表を準備する』を参照してください。

コマンド構文

```
▶▶—DISABLE TEXT FILES—table-name—HANDLE—handle-column-name—▶▶
```

コマンド・パラメーター

table-name

使用不可にする外部テキスト・ファイルのハンドル列を含んだ、接続しているデータベース内のテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

HANDLE handle-column-name

DB2 テキスト・エクステンダーで使用できないようにするハンドル列の名前。

使用上の注意

索引が削除されます。

ハンドル列内の変更 (挿入、更新、および削除) を記録するログ表が削除されます。ログ表に項目を書き込むトリガーも削除されます。

DISABLE TEXT TABLE

このコマンドは、表中のすべてのテキスト列を DB2 テキスト・エクステンダーで使用できないようにします。

許可

表に関する所要のオブジェクト権限が付与されていなければなりません。詳しくは、14 ページの『DB2 テキスト・エクステンダーの SQL 表を準備する』を参照してください。

コマンド構文

```
▶—DISABLE TEXT TABLE—table-name—▶
```

コマンド・パラメーター

table-name

使用不可にする列が含まれた、接続しているデータベース内のテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

使用上の注意

このコマンドは、表中のすべてのテキスト列を DB2 テキスト・エクステンダーで使用できないようにします。

この表中のテキスト列を ENABLE TEXT COLUMN によって個別に使用可能にした場合は、それらの列に関連したテキスト索引がすべて削除されます。(テキスト列を個別に使用不能にし、それらの列に関連したテキスト索引を個別に削除するには、DISABLE TEXT COLUMN コマンドを使用します。) ENABLE TEXT TABLE も使用してこの表中のテキスト列を使用可能にした場合は、すべてのテキスト列について 1 つの共通索引があります。このコマンドは共通索引を削除します。

テキスト列での変更 (挿入、更新、および削除) を記録するために使用されるログ表が削除されます。ログ表に項目を書き込むトリガーが削除されます。

ENABLE SERVER FOR DB2TEXT

このコマンドは、DB2 テキスト・エクステンダーが現行のデータベースを使用できるようにします。

許可

ありません。

コマンド構文

▶—ENABLE SERVER FOR DB2TEXT—▶

コマンド・パラメーター

ありません。

使用上の注意

このコマンドを発行するには、データベースに明示的または暗黙的に接続されていなければなりません (106 ページの『CONNECT』を参照)。

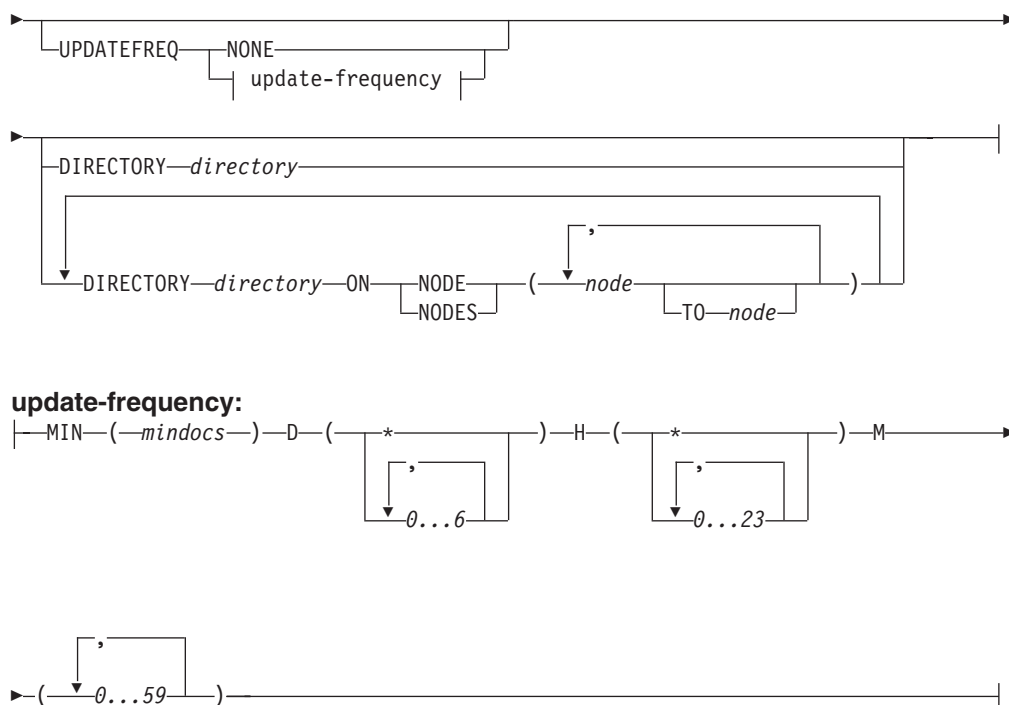
このコマンドは、接続しているデータベースを DB2 テキスト・エクステンダーで使用するよう準備します。これは、DB2 テキスト・エクステンダーのテキスト表またはテキスト列をデータベース内で使用できるようにするための必須ステップです。ENABLE SERVER は、DB2TX.TEXTINDEXES という DB2 テキスト・エクステンダー・カタログ・ビューを作成します。これについては、86 ページの『DB2 テキスト・エクステンダー・カタログ・ビューの処理』で説明しています。さらに、DB2TX.TEXTCOLUMNS というカタログ・ビューも作成します。

1 つのシステム / ユーザー ASP、あるいは 1 つまたは複数の独立 ASP を使用可能にできます。すなわち、複数の ASP を使用可能にする場合は、その ASP はすべて独立 ASP でなければなりません。

また、このコマンドは、10 ページの『テキスト構成設定値』で説明されているテキスト構成設定値も作成します。

DB2 テキスト・エクステンダー特殊タイプおよび DB2 テキスト・エクステンダー関数の宣言などの、他の作業も実行されます。

ENABLE TEXT COLUMN コマンド



コマンド・パラメーター

table-name

使用可能にする列を含む、接続しているデータベース内のテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

text-column-name

DB2 テキスト・エクステンダーで使用できる列の名前。この列は次のいずれかのタイプです。CHAR、VARCHAR、LONG VARCHAR、CLOB、DBCLOB、GRAPHIC、VARGRAPHIC、または LONG VARGRAPHIC。文書タイプがこれらのいずれでもない場合は、FUNCTION を使用して、文書タイプを変換してください。

注

テキスト列のデータ・タイプとして LONG VARCHAR を使用する場合、テキスト表を作成する際に、ハンドル列 (複数可) を作成しなければなりません。

ハンドル列のタイプは、db2tx.DB2TextH (テキスト列を使用可能化する場合)、または db2tx.DB2TextFH (外部ファイルを使用可能化する場合) のいずれかでなければなりません。

LONG VARCHAR 列を含む表にエラー処理列 (複数可) を作成しないと、ENABLE TEXT COLUMN コマンドの実行時に「SQL ステートメントが長すぎるか複雑すぎます (SQL statement too long or complex)」などのエラー・メッセージが戻されます。

FUNCTION function-name

DB2 テキスト・エクステンダーが、以下のいずれのタイプでもない列の中のテキスト文書をアクセスするために使用するユーザー定義関数の名前。CHAR、VARCHAR、LONG VARCHAR、CLOB、DBCLOB、GRAPHIC、VARGRAPHIC、または LONG VARGRAPHIC。詳細については、52 ページの『サポートされないデータ・タイプのテキスト列の使用可能化』を参照してください。

HANDLE handle-column-name

DB2 テキスト・エクステンダーの関数で使用するために表に追加するハンドル列の名前。

CCSID ccsid

テキスト文書を索引付けする際に使用するコード化文字セット ID。

Ngram 索引のテキスト列を使用可能にする際に、CCSID を指定する場合には、サブシステムの CCSID と同じ CCSID を指定しなければならず、検索の際にもこの CCSID (サブシステムの CCSID) と同じものを使用しなければなりません。デフォルトの CCSID を調べるには、以下のコマンドを使用します。

```
db2tx get text cfg
```

インストール時のデフォルトは、サブシステムの CCSID。

キーワードを指定しない場合は、テキスト構成設定値内に指定された CCSID が使用されます。後でテキスト構成設定値に対して行う変更は無視されるため、使用される値はテキスト文書を索引付けする際の値ではなく、列が使用可能になったときの値となります。

サポートされる CCSID に関する情報については、26 ページの『CCSID』を参照してください。

LANGUAGE language

テキストを書き込む言語。この値により、テキスト文書を索引付けする際やテキスト文書内を検索する際に使用される辞書が決まります。197 ページの『第 15 章 基本語形索引と一致語句索引の言語的な処理』で、辞書を使用方法を説明しています。

このキーワードでは、1 度に列全体に言語を指定します。

キーワードを指定しない場合は、テキスト構成設定値内に指定された言語が使用されます。後でテキスト構成設定値に対して行う変更は無視されるため、使用される値はテキスト文書を索引付けする際の値ではなく、列が使用可能になった時点での値となります。

サポートされる言語は、40 ページの表 4 に記載されています。

FORMAT format

保管されるテキスト文書のタイプで、WordPerfect または ASCII など。

DB2 テキスト・エクステンダーでは、文書を索引付けする際に、この情報が必要です。サポートされる文書形式は、23 ページの『サポートされる文書形式』に列挙されています。

ENABLE TEXT COLUMN コマンド

構造化文書用にサポートされている文書形式は以下のとおりです。

- ASCII_SECTIONS

ASCII_SECTIONS 形式の文書には、ネストされたセクションを含めることはできません。(ネスト・セクションについては、37 ページの『構造化文書 (セクション・サポート) での作業』を参照してください。) セクションの開始タグは、次の開始タグで終わります。

- HTML

サンプルの文書モデル・ファイルは HTML 文書用のものです。この文書には修正可能な、標準的 HTML 定義のサブセットが含まれています。HTML 文書には、ネストされたセクションを含めることはできません。

- XML

XML 文書の処理には、文書タイプ定義 (DTD) の評価が含まれます。文書に割り当てられたモデルは、DTD と比較してチェックされます。文書モデル・ファイル中で定義されているタグが DTD で定義されていない場合、その文書は索引付けされません。認識されている DTD にモデルが定義されていない場合、その文書は索引付けされません。XML 文書にはネストされたセクションを含めることができます。

XML は、Text Search Engine リリース 4 (FMID HIMN230) を含む、OS/390® リリース 2.9 および z/OS のみで使用できます。

これらの形式については、文書モデル・ファイルに構造情報を指定しなければなりません。37 ページの『構造化文書 (セクション・サポート) での作業』を参照してください。形式 TDS と INDEXPROPERTY SECTION_ENABLED を指定した場合、文書形式は ASCII_SECTIONS であると見なされます。

モデル・ファイル中で定義されていないタグは、索引タイプに従って通常の方法で索引付けされます。

このキーワードは、1 度に列全体に形式を指定します。INSERT ステートメントで INIT_TEXT_HANDLE 関数を使うと、別々に挿入されるテキスト文書ごとにこの値を変更することができます。

このキーワードを指定しない場合は、テキスト構成設定値内に指定された形式が使用されます。後でテキスト構成設定値に対して行う変更は無視されるため、使用される値はテキスト文書を索引付けするときの値ではなく、列が使用可能になったときの値となります。

INDEXTYPE

作成する索引のタイプです。詳細については、31 ページの『検索のタイプ』を参照してください。

PRECISE

用語はテキスト文書に出てくるとおりに索引付けされ、検索されません。

LINGUISTIC

用語は索引付けの前に言語的な処理が行われます。また、検索用語も検索開始前に言語的な処理が行われます。

NGRAM

用語は、辞書を使用するのではなく、一連の文字を解析することによ

ENABLE TEXT COLUMN コマンド

って索引付けされます。索引付けしている文書に DBCS 文字が含まれている場合は、この索引タイプが必須ですが、SBCS 文書に対して Ngram 索引を使用することもできます。

INDEXTYPE キーワードを指定しない場合、テキスト構成設定値の値が使用されます。

Ngram 索引では、XML 形式の文書はサポートされていません。

INDEXOPTION

索引作成時に使用されるオプションです。

CASE_ENABLED

このオプションは、**Ngram 索引に対してのみ**使用可能です。通常、Ngram 索引では、大文字小文字を区別する検索は行われません。CASE_ENABLED を指定することによって、文書は必ず、大文字小文字を区別する検索が可能となるように索引付けされます。詳細については、33 ページの『ファジー検索または DBCS 文書の検索』を参照してください。

NORMALIZED

このオプションは、**一致語句索引に対してのみ**使用可能です。正規化された一致語句索引は、以下の点で一致語句索引と異なります。

- 大文字小文字の区別がない。すべて大文字の場合を除き、すべての語は小文字に変換されます。
- すべて大文字の語は停止ワードのフィルター操作の対象にならない。たとえば、省略語 UK は索引付けされます。
- 英語の検索用語を拡張して、発見的手法アルゴリズムを使用する lemmatization 形式を含めることができる。house を検索すると houses も検索されます。

INDEXPROPERTY SECTIONS_ENABLED DOCUMENTMODEL(S) model

選択した索引タイプのプロパティ。

SECTIONS_ENABLED を指定すると、選択した索引タイプに、文書構造に関する情報を含めることができます。

DOCUMENTMODEL/DOCUMENTMODELS *model* は、索引付けされる文書にデフォルトとして関連付けられる 1 つまたは複数のモデルを指定します。モデル名を指定しなければならないのは、索引プロパティ SECTIONS_ENABLED を使用している場合です。モデルのリストを指定すると、最初のモデルが索引のデフォルトのモデルとして使用されます。モデルに対する参照が文書にない場合や、検索の際にモデルを指定しない場合は、デフォルトのモデルが使用されます。

モデル名に使用できる文字は、a ~ z、A ~ Z、および 0 ~ 9 です。

モデル定義ファイル `desmodel.ini` 中のモデル定義に対応するモデル名を指定しなければなりません。モデル名には大文字小文字の区別があるという点に注意してください。

索引に関連付けられている 1 つまたは複数のモデルを変更するには、以下の手順を実行します。

1. DISABLE TEXT COLUMN を使用して索引を使用不能にする。

ENABLE TEXT COLUMN コマンド

2. ENABLE TEXT COLUMN を使用し、異なる文書モデル名を指定して文書モデル名を索引付けし直す。

UPDATEFREQ update-frequency

更新を実行する際、およびログ表で待ち行列に入れるテキスト文書の最少数に関する索引の更新頻度です。所定の日時にログ表中に十分な数のテキスト文書が存在しない場合、索引は更新されません。

構文については、35 ページの『索引の更新』で説明しています。

UPDATEFREQ を指定しない場合は、テキスト構成設定値内に指定されたデフォルトの頻度が使用されます。

ヒント

表が多数存在する場合は、デフォルト値を使用しないことを考慮してください。表ごとに個別の更新頻度を設定すると、すべての表を同時に索引付けし、その結果システム・リソース上で不必要にロードが延長されることを防ぐことができます。

NONE 以後索引の更新を行いません。これは、これ以上変更を行わないテキスト列のためのものです。

これらの更新頻度設定は、すでに ENABLE TEXT TABLE によって表全体について設定済みである場合は無視されます。

DIRECTORY directory

テキスト索引を保管するディレクトリー・パスです。指定されたパスは、“txins nnn ” と連結されます。ここで、 nnn はノード番号です。

これは、DB2 テキスト・エクステンダー・サーバーが稼働しているシステム上のディレクトリーです。ディレクトリーが存在していない場合は、作成されます。

DIRECTORY キーワードを指定しない場合は、テキスト構成設定値内の DIRECTORY 設定の値が使用されます。

設定は、ENABLE TEXT TABLE によってすでに設定済みの場合は無視されます。

ON NODE node [TO node]

ディレクトリー・パス名が割り当てられるノードまたはノードの範囲の番号。

UPDATEINDEX

UPDATEFREQ で設定された次の周期的索引付けを待たず、このコマンドの完了後すぐに、このハンドル列に関連したテキスト文書を索引付けするかどうかを指定するキーワードです。

UPDATE

このコマンドの完了後すぐに、テキスト文書の索引付けが行われます。

NOUPDATE

このコマンドの UPDATEFREQ またはテキスト構成設定値によって指定された更新頻度設定値に設定された時間に、索引付けが行われます。

このキーワードを指定しない場合は、テキスト構成設定値内の値が採用されます。

TABLESPACE tablespace-name

ハンドル列に内部的に作成される索引の表スペースの名前。表スペースは事前に作成済みでなければなりません。

COMMITCOUNT count

挿入または更新の数を示す 500 ~ 1 000 000 の値で、この値を超えると、DB2 UDB for iSeries により中間コミット・ステートメントが出されなければなりません。この値を使用することにより、ラージ表、列、または多数の外部ファイル用のログ・スペースが不足するという状態の発生を避けることができます。

使用上の注意

このコマンドは、指定の DB2 表にハンドル列を追加します。各ハンドル列はテキスト列に関連付けられており、DB2 テキスト・エクステンダーの関数で使用されます。

この表がまだ使用可能になっておらず、共通索引を作成できない場合は、このテキスト列に関連した索引が作成されます。

また、ログ表もデータベース内に作成されます。ログ表は、テキスト列に対する変更内容、すなわち挿入、更新、および削除を記録するために使用されます。ログ表を自動的に最新の状態に維持するために、テキスト列に関する挿入、更新、および削除トリガーが定義されます。

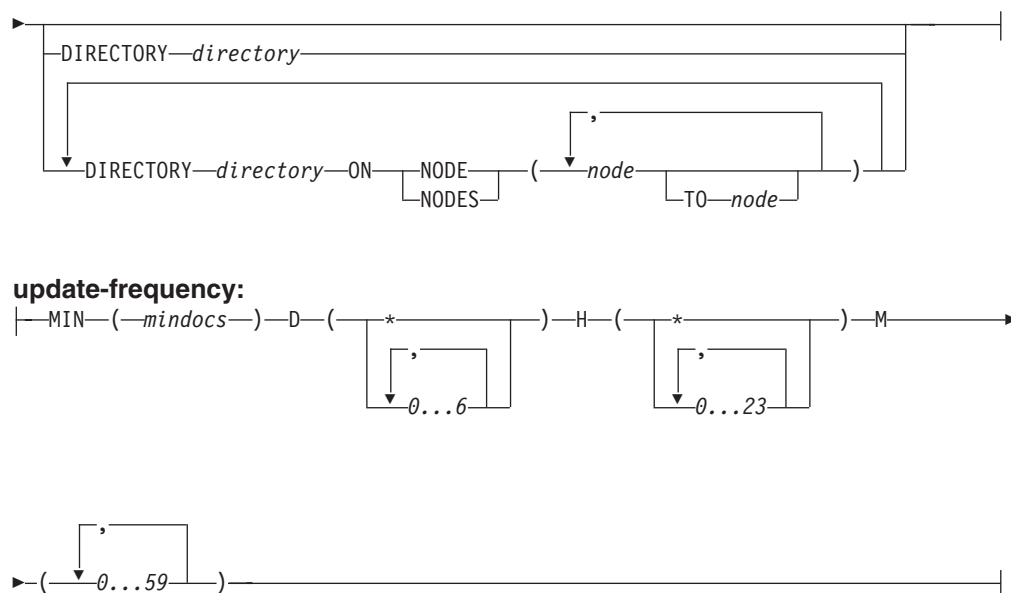
使用可能にしているテキスト列がマルチノード・ノードグループの一部である表に属している場合には、指定する索引ディレクトリーは、すべての物理ノードで使用できなければなりません。テキスト構成で指定されるデフォルト・ディレクトリーを使用する場合には、ノード・グループのすべてのノードでパスを使用できることを確認してください。この方法が適切でない場合には、それぞれのノードについて特定のパスを ENABLE TEXT COLUMN コマンドに指定することができます。

DB2 テキスト・エクステンダーに使用可能にされた表を含むノード・グループのノード構成を変更するには、表を索引付けし直さなければなりません。

ヒント

このステップで、ログ用スペースを使い尽くしてしまった場合は、51 ページの『ラージ表でのテキスト列の使用可能化』を参照して、可能な解決方法を見つけてください。

ENABLE TEXT FILES コマンド



コマンド・パラメーター

table-name

索引付けされる外部テキスト・ファイルに関連付けられる、接続しているデータベース内のテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

その他のコマンド・パラメーターの説明については、113 ページの『ENABLE TEXT COLUMN』を参照してください。

使用上の注意

このコマンドは、指定の DB2 表にハンドル列を追加します。各ハンドル列はテキスト列に関連付けられていて、DB2 テキスト・エクステンダーの関数で使用されます。これらのファイルに関連付けられる索引が作成されます。

ハンドル列名は、ENABLE TEXT FILES で以前に使用されていた場合には、テキスト列のハンドル列を識別するために再使用することはできません。

ログ表は、ファイルに対する変更を記録するために作成されますが、DB2 UDB for iSeries は、それらのファイルを制御しないため、そのような変更を認知できません。したがって、変更を記録するためには、手操作で、トリガーを活動化させる必要があります。これを行う方法については、78 ページの『外部ファイルの索引の更新』を参照してください。

このステップで、ログ用スペースを使い尽くしてしまった場合は、51 ページの『ラージ表でのテキスト列の使用可能化』を参照して、可能な解決方法を見つけてください。

ENABLE TEXT FILES コマンド

注

索引付けの際に起こる可能性がある CCSID 変換の問題を防ぐには、ENABLE TEXT FILES コマンドを実行する前に外部ファイルの CCSID を設定しなければなりません。

ENABLE TEXT TABLE コマンド

コマンド・パラメーター

table-name

接続しているデータベース内で使用可能にするテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

INDEXTYPE

作成する索引のタイプ。詳細については、31 ページの『検索のタイプ』を参照してください。

PRECISE

用語はテキスト文書に出てくるとおりに索引付けされ、検索されます。

LINGUISTIC

用語は索引付けの前に言語的な処理が行われます。また、検索用語も検索開始前に言語的な処理が行われます。

NGRAM

用語は、辞書を使用するのではなく、一連の文字を解析することによって索引付けされます。索引付けしている文書に DBCS 文字が含まれている場合は、この索引タイプが必須ですが、SBCS 文書に対して Ngram 索引を使用することもできます。

INDEXTYPE キーワードを指定しない場合は、テキスト構成が使用されます。

INDEXOPTION

索引作成時に使用されるオプション。

CASE_ENABLED

このオプションは、Ngram 索引に対してのみ使用可能です。通常、Ngram 索引では、大文字小文字を区別する検索は行われません。CASE_ENABLED を指定することによって、文書は必ず、大文字小文字を区別する検索が可能であるように索引付けされます。詳細については、33 ページの『ファジー検索または DBCS 文書の検索』を参照してください。

INDEXPROPERTY SECTIONS_ENABLED DOCUMENTMODEL(S) model

選択した索引タイプのプロパティ。

SECTIONS_ENABLED を指定すると、選択した索引タイプに、文書構造に関する情報を含めることができます。

DOCUMENTMODEL/DOCUMENTMODELS model は、索引付けされる文書にデフォルトとして関連付けられる 1 つまたは複数のモデルを指定します。モデル名を指定しなければならないのは、索引プロパティ SECTIONS_ENABLED を使用している場合です。モデルのリストを指定すると、最初のモデルが索引のデフォルトのモデルとして使用されます。モデルに対する参照が文書にない場合や、検索の際にモデルを指定しない場合は、デフォルトのモデルが使用されます。

モデル名に使用できる文字は、a ~ z、A ~ Z、および 0 ~ 9 です。

ENABLE TEXT TABLE コマンド

モデル定義ファイル `desmodel.ini` 中のモデル定義に対応するモデル名を指定しなければなりません。

索引に関連付けられている 1 つまたは複数のモデルを変更するには、以下のようになります。

1. `DISABLE TEXT TABLE` を使用して索引を使用不能にする。
2. `ENABLE TEXT TABLE` を使用し、異なる文書モデル名を指定して文書モデル名を索引付けし直す。

UPDATEFREQ update-frequency

更新を実行するとき、およびログ表で待ち行列に入れるテキスト文書の数に関する索引の更新頻度。所定の日時にログ表中に十分な数のテキスト文書が存在しない場合は、索引は更新されません。

構文については、35 ページの『索引の更新』で説明しています。

`UPDATEFREQ` を指定しない場合は、テキスト構成設定値内に指定されたデフォルトの頻度が使用されます。

NONE 以後索引の更新を行いません。これは、これ以上変更を行わないテキスト列のためのものです。

ヒント

表が多数存在する場合は、デフォルト値を使用しないことを考慮してください。表ごとに個別の更新頻度を設定すると、すべての表を同時に索引付けし、その結果システム・リソース上で不必要にロードが延長されることを防ぐことができます。

DIRECTORY directory

テキスト索引を保管するディレクトリー・パス。指定されたパスは、“`txins nnn` ” と連結されます。ここで、 nnn はノード番号です。

これは、DB2 テキスト・エクステンダー・サーバーが稼働しているシステム上のディレクトリーです。ディレクトリーが存在していない場合は、作成されます。

`DIRECTORY` キーワードを指定しない場合は、テキスト構成設定値内の `DIRECTORY` 設定の値が使用されます。

ON NODE node [TO node]

ディレクトリー・パス名が割り当てられるノードまたはノードの範囲の番号。

使用上の注意

この表中のすべてのテキスト列に関連した新規のテキスト索引が作成されます。テキスト列ごとに別々の索引を作成するのではなく、表のすべてのテキスト列について 1 つの共通索引を作成したい場合にこれを実行します。

表を使用可能にした場合は、検索したいテキスト列ごとに `ENABLE TEXT COLUMN` を実行しなければなりません。

ENABLE TEXT TABLE コマンド

ログ表がデータベース内に作成されます。この表は、後で使用可能にするテキスト列での変更内容、すなわち挿入、更新、および削除を記録するために使用されます。

テキスト列が使用可能になるとトリガーが作成され、このトリガーがテキストに対する変更をモニターし、索引付けが必要な文書のログ表に自動的に記録を保持します。

DB2 テキスト・エクステンダーは、ログ表にリストされたテキスト文書を UPDATEFREQ キーワードで指定されたように周期的に索引付けします。

GET INDEX SETTINGS コマンド

Update frequency (UPDATEFREQ) = NONE
Node 0
Index directory (DIRECTORY) = /QIBM/UserData/DB2Extenders/Text/indices

GET INDEX STATUS

このコマンドは、指定されたハンドル列または表について、以下の索引の状況情報を表示します。

- 検索関数を使用できるかどうか
- 索引更新関数を使用できるかどうか
- 再編成関数を使用できるかどうか
- スケジュールされた文書の数
- 索引付けされた文書の数
- 1 次索引中の索引付けされた文書の数
- 2 次索引中の索引付けされた文書の数
- エラー・イベント

許可

表に関する所要のオブジェクト権限が付与されていなければなりません。詳しくは、14 ページの『DB2 テキスト・エクステンダーの SQL 表を準備する』を参照してください。

コマンド構文

```

▶▶ GET INDEX STATUS table-name ───────────────────────────────────▶▶
    └── HANDLE handle-column-name ───────────────────────────────────┘
  
```

コマンド・パラメーター

table-name

状況を表示するテキスト列を含む、接続しているデータベース内のテキスト表の名前。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

HANDLE handle-column-name

状況を表示したいハンドル列の名前。

使用上の注意

複数索引表については、ハンドル列の名前を指定する必要があります。

GET STATUS

このコマンドは、サブシステム、表、またはテキスト列の使用可能状況に関する情報を表示します。

許可

ありません。

コマンド構文

▶▶ GET STATUS ◀◀

コマンド・パラメーター

ありません。

使用上の注意

このコマンドは、サーバーが使用可能であるかどうか、サブシステム内の使用可能なテキスト表の名前、使用可能なテキスト列とそれらに関連付けられているハンドル列の名前、および外部ファイルのハンドル列の名前を表示します。

GET TEXT CONFIGURATION

このコマンドは、接続しているデータベースのテキスト構成のデフォルト設定値を表示します。

これらのデフォルト設定値を変更するには、103 ページの『CHANGE TEXT CONFIGURATION』を使用してください。

許可

ありません。

コマンド構文

▶▶ GET TEXT CONFIGURATION

```
▶▶ GET TEXT CONFIGURATION
```

The diagram shows the command structure. It starts with '▶▶ GET TEXT' followed by 'CONFIGURATION'. Below 'CONFIGURATION' is a box containing 'CFG'. A long arrow points to the right from the end of 'CONFIGURATION'.

コマンド・パラメーター

ありません。

使用上の注意

テキスト構成情報の例については、82 ページの『テキスト構成設定値の表示』を参照してください。

QUIT

このコマンドは、DB2 テキスト・エクステンダー・コマンド行プロセッサを停止し、オペレーティング・システムに制御を戻します。

許可

ありません。

コマンド構文

▶▶—QUIT—◀◀

コマンド・パラメーター

ありません。

使用上の注意

データベースへの接続は終了します。

UPDATE INDEX

このコマンドは、即時に索引付けを開始します。これは索引を最新のものにし、索引に関連したテキスト列 (複数の場合もある) の現行の内容を反映します。

外部ファイルの更新を索引に反映させるためには、対応するハンドル列で更新ステートメントを出すことによって、その「変更」項目をログ表に強制的に入れます。この例については、78 ページの『外部ファイルの索引の更新』を参照してください。

許可

表に関する所要のオブジェクト権限が付与されていなければなりません。詳しくは、14 ページの『DB2 テキスト・エクステンダーの SQL 表を準備する』を参照してください。

コマンド構文

```

▶▶ UPDATE INDEX table-name [HANDLE handle-column-name]
[COMMITCOUNT count]

```

コマンド・パラメーター

table-name

更新する索引のあるテキスト列を含む、接続しているデータベース内のテキスト表の名前。また、これは共通索引表の名前とすることもできます。スキーマ名がユーザー ID と異なる場合は、表名に明示のスキーマ名 (修飾子) を含めなければなりません。

HANDLE handle-column-name

これが共通索引表である場合は、*handle-column-name* は不要であり、無視されます。更新された索引は、個別のテキスト列にではなく、表全体に関連したものです。

これが複数索引表である場合は、*handle-column-name* は索引を更新したいハンドル列の名前です。

COMMITCOUNT count

挿入または更新の数を示す 500 ~ 1 000 000 の値で、この値を超えると、DB2 UDB for iSeries により中間コミット・ステートメントが出されなければなりません。この値を使用することにより、ラージ表、列、または多数の外部ファイルを使用可能化する際に、ログ・スペースが不足する事態を避けることができます。

第 10 章 サーバー用の管理コマンド

この章では、サーバー用の管理コマンドの構文を説明します。サーバー管理は、ユーザーが実行できる、DB2 テキスト・エクステンダー・サーバーの開始、停止、およびその状況のチェックと、サンプル・データベースおよびサンプル表の作成の各タスクから構成されています。これらのコマンドの使用方法については、11 ページの『DB2 テキスト・エクステンダー・サーバーの設定および保守』で説明しています。

コマンド	用途	ページ
TXICRT	DB2 テキスト・エクステンダー・インスタンスを作成する。	138
TXIDROP	DB2 テキスト・エクステンダー・インスタンスを除去する。	139
TXSAMPLE	サンプル表を作成して使用可能にする。	140
TXSTART	DB2 テキスト・エクステンダー・サービスを開始する。	141
TXSTATUS	検索サービスの状況を表示する。	142
TXSTOP	DB2 テキスト・エクステンダー・サービスを停止する。	143
IMOTHESC	シソーラス定義ファイルをコンパイルする。	144
IMOTHESN	Ngram シソーラス定義ファイルをコンパイルする。	146
IMOTRACE	トレース情報を生成する。	148
TXVERIFY	インストーラが正常に完了したか検査する。	152

TXICRT

このコマンドは、DB2 テキスト・エクステンダー・インスタンスを作成します。

許可

テキスト・エクステンダーの管理者権限を持っていないとなりません。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

コマンド構文

▶—CALL—PGM(QDB2TX/TXICRT)—▶

コマンド・パラメーター

ありません。

使用上の注意

テキスト表またはテキスト列は、DB2 テキスト・エクステンダー・インスタンスが作成されている場合にのみ、使用可能とすることができます。

TXIDROP

このコマンドは、DB2 テキスト・エクステンダー・インスタンスとその索引すべてを一緒に除去します。

許可

テキスト・エクステンダーの管理者権限を持っていないければなりません。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

コマンド構文

```
▶▶—CALL—PGM(QDB2TX/TXIDROP)—▶▶
```

コマンド・パラメーター

ありません。

使用上の注意

インスタンスを除去する前に、それが使用可能であるデータベースはいずれも使用不可にしておきます。

TXSAMPLE

このコマンドは、サンプル表を作成し、英語文書のサンプルを COMMENT 列にロードし、そのテキスト列を使用可能にします。9 ページの『インストール検査のためにサンプル・データベースを準備する』を参照してください。

許可

テキスト・エクステンダーの管理者権限またはテキスト・エクステンダーのユーザー権限が必要です。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

コマンド・パラメーター

subsystem-name

すでに DB2 テキスト・エクステンダーによって使用可能にされているサブシステムの名前。ここにサンプル表が作成されます。

user-id

これは、クライアント・ワークステーションで作業している場合にのみ必要です。

password

これは、クライアント・ワークステーションで作業している場合にのみ必要です。

使用上の注意

データベースの CCSID が 500 ではない場合、Ngram 索引タイプの使用可能化ステップの例を参照し、使用しているデータベースの CCSID に合わせて CCSID を変更してください。

ヒント

このコマンドは、クライアント・ワークステーションにおいても使用できます。

TXSTART

このコマンドは DB2 テキスト・エクステンダーの検索サービスを開始します。

許可

テキスト・エクステンダーの管理者権限を持っていないとできません。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

コマンド構文

▶▶—CALL—PGM(QDB2TX/TXSTART)—◀◀

コマンド・パラメーター

ありません。

使用上の注意

データへの索引付けやデータに基づいた検索を行う前に、DB2 テキスト・エクステンダーの検索サービスを開始しなければなりません。

TXSTATUS

このコマンドは、DB2 テキスト・エクステンダーが起動され、実行中であるかどうかを表示します。

許可

テキスト・エクステンダーの管理者権限またはテキスト・エクステンダーのユーザー権限が必要です。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

コマンド構文

▶▶—CALL—PGM(QDB2TX/TXSTATUS)—▶▶

コマンド・パラメーター

ありません。

TXSTOP

このコマンドは DB2 テキスト・エクステンダー・サービスを停止します。

許可

テキスト・エクステンダーの管理者権限を持っていないとできません。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

コマンド構文

▶—CALL—PGM(QDB2TX/TXSTOP)—▶

コマンド・パラメーター

ありません。

使用上の注意

このコマンドは、DB2 自体は停止させません。

IMOTHESC

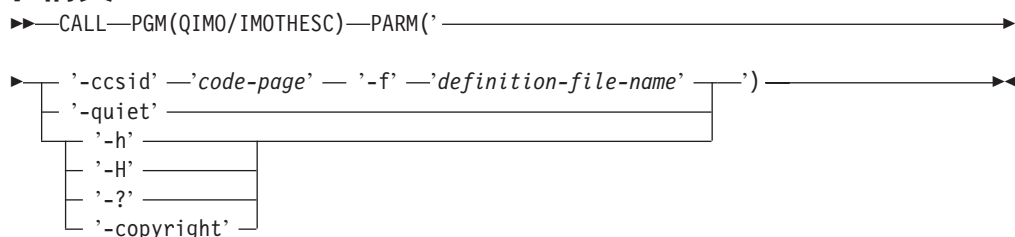
このコマンドは、シソーラス定義ファイルをコンパイルします。このシソーラスは、一致語句索引または基本語形索引の検索にのみ使用できます。

Ngram シソーラス定義ファイルのコンパイルについては、146 ページの『IMOTHESN』を参照してください。

許可

ありません。

コマンド構文



コマンド・パラメーター

-f *definition-file-name*

シソーラス定義の入っている SGML ファイルの名前。このファイル名には、ファイルに対する絶対パスか相対パスが含まれていなければなりません。

シソーラス辞書は、定義ファイルと同じディレクトリーに生成されます。この辞書は定義ファイルと同じ名前になりますが、拡張子は th1 ~ th6 になります。

ヒント

同じ名前のシソーラス・ファイルがあると上書きされるため、シソーラスごとに別個のディレクトリーを使用してください。

-ccsid *code-page*

シソーラス定義ファイルが作成される際のコード・ページ。現時点では、コード・ページ 850 のみがサポートされています。

-quiet 出力情報は表示されません。

-copyright

製品の内部製造番号を戻します。この番号は、問題を報告する際に使用してください。

-h、-H、または -?

ヘルプ情報を表示します。

使用上の注意

このコマンドは、標準シソーラス定義ファイルを 2 進数形式にコンパイルするのに使
用します。定義ファイルは SGML 形式でなければなりません。

コンパイル済みのシソーラス・ファイルを使用するには、サーバー・インスタ
ンスの辞書ディレクトリーに移動して、検索時にファイルの場所を指定してくださ
い。

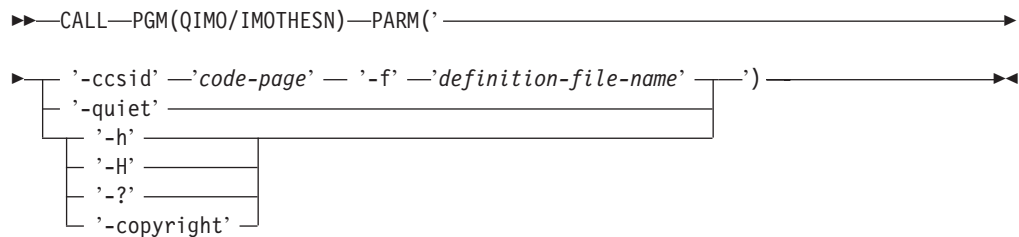
IMOTHESN

このコマンドは、Ngram シソーラス定義ファイルをコンパイルします。このシソーラスは、Ngram 索引の検索にのみ使用できます。

許可

ありません。

コマンド構文



コマンド・パラメーター

-f *definition-file-name*

シソーラス定義の入っているファイルの名前。このファイル名には、ファイルに対する絶対パスか相対パスが含まれていなければなりません。ファイル名は 8+3 文字に制限されています。拡張子は任意指定です。

シソーラス辞書は、定義ファイルと同じディレクトリーに生成されます。この辞書は定義ファイルと同じ名前になりますが、拡張子は wdf、wdv、grf、grv、MEY、ROS、NEY、SOS、および 1kn (*n* は数字) になります。

ヒント

同じ名前のシソーラス・ファイルがあると上書きされるため、シソーラスごとに別個のディレクトリーを使用してください。

-ccsid *code-page*

シソーラス定義ファイルが作成される際のコード・ページ。サポートされているコード・ページのリストについては、26 ページの『CCSID』を参照してください。

-quiet 出力情報は表示されません。

-copyright

製品の内部製造番号を戻します。この番号は、問題を報告する際に使用してください。

-h、-H、または -?

ヘルプ情報を表示します。

使用上の注意

このコマンドは、シソーラス定義ファイルを 2 進数形式にコンパイルするために使用します。定義ファイルは、213 ページの『Ngram シソーラスの作成』で説明されている形式でなければなりません。

コンパイル済みのシソーラス・ファイルを使用するには、サーバー・インスタンスの辞書ディレクトリーに移動して、検索時にファイルの場所を指定してください。

辞書ディレクトリーは、次のディレクトリーです。

```
/QIBM/ProdData/imo/dict
```


dump | dmp

トレース情報を 2 進数形式でファイルに書き込みます。

off トレース機能を停止します。

format | fmt

2 進数形式のトレース情報をフォーマット設定します。トレース機能がオフに切り替えられている際に、ダンプ・ファイルの名前とフォーマット設定済みトレース情報を保持するファイルの名前を指定することによって、ダンプ・ファイルをフォーマット設定することができます。

change | chg

トレース用マスク `maxSysErrors` または `maxRecordSize` を変更します。

clear | clr

トレースを消去します。

flow | flw

トレースの制御フローを表示します。

information | info | inf

トレースに関する情報を入手します。

options

トレース機能情報がオンである際 (これは共用メモリーの使用率を減らします)、またはトレース情報をフォーマット設定する際に、トレース情報にフィルター操作を適用します。トレース情報が非常に大量でない限り、通常はすべてのトレース情報を書き込んでから、フォーマット設定時にそれをフィルター操作する方が簡単です。

-m 「マスク」を追加して、トレースに組み込むイベント、コンポーネント、および関数を指定します。デフォルトには、すべてのトレースが行われます。マスクは `2.2-6.1,3.*` のように、それぞれがピリオドで区切られた 4 つの部分からなります。範囲を示す区切り記号として “-” を使用することができ、また、リストを示す区切り記号として “;” を使用することができます。たとえば、`2-6` は、ID の範囲が 2 から 6 までのイベントのみを組み込みます。コンポーネント `2` および `6` のみを組み込む場合は、`2,6` と指定します。

products

プロダクト ID。DB2 テキスト・エクステンダーのプロダクト ID は “2” です。TextMiner のプロダクト ID は “3” です。

events トレースに組み込むイベント・タイプのセット。次のとおりです。

0	system_error
1	system_error
2	system_error
3	non-fatal_error
4	non-fatal_error
5	api_errcode
7	fnc_errcode
8	trap_error

IMOTRACE コマンド

10	api_entry
11	api_exit
13	api_retcode
15	api_data
30	fnc_entry
31	fnc_exit
33	fnc_retcode
35	fnc_data

components

トレース対象のコンポーネント。

DB2 テキスト・エクステンダーのコンポーネント ID は次のとおりです。

1	COMMAND_LINE_INTERFACE
2	UDF
3	STORED_PROCEDURES
4	ADMINISTRATION
5	INDEX_CONTROL
6	LIBRARY_SERVICES
7	DES_PARSER
8	DES_DEMON
9	DES_API
10	SERVICES

TextMiner のコンポーネント ID は次のとおりです。

1	automachine
2	bgproc (バックグラウンド・プロセス)
3	cluster
4	common
5	commsrvc (共通サービス)
6	communic (通信)
7	daemon
8	dsclient
9	environ (環境)
10	glue
11	idxcomm (索引作成、共通部分)
12	libsrv (ライブラリー・サービス)
13	search
14	trace
15	guru
16	indexbld (索引作成、tm のみ)
17	indexeng (索引エンジン、tm のみ)
18	smsearch
19	(検索エンジン、tm のみ)
20	tmsearch
21	gtrcm (gtr、共通部分)
22	gtrsrch (検索、gtr のみ)
23	gtridx (索引作成、gtr のみ)

functions

アスタリスク (*)。トレースする関数のセット。IBM サポート・センターから他の指示がないかぎり、アスタリスク (*) を使用して、すべての関数をトレースします。

-rc *return-code*

return-code をシステム・エラーとして扱います。

-e *max-system-errors*

整数。この数のエラーが発生したら、トレースを停止します。デフォルトは 1 で、これは、最初にシステム・エラーが起きたときに、それ以降に発生するより低い重大度のイベントをすべて抑止することを示します。これは、最初に起きる重大エラーのみを重視する場合には受け入れることができますが、最初のシステム・エラー発生後の完全なトレースが見たい場合には、さらに大きい数 (-e 50 など) を指定する必要があります。トレース宛先は共用メモリーです。

-r *max-record-size*

整数。この数のレコードがトレース・ファイルに書き込まれた後で、トレースを停止します。デフォルトは 16 KB です。

-i *initial-buffer-size*

整数。トレースの先頭からこの数のレコードを保持します。-i を指定した場合、デフォルトは 16 KB です。UNIX システムでは、推奨バッファ・サイズは 2 MB です。

-i または -i のいずれも指定しない場合、デフォルトは -1 になります。

-i を指定すると、折り返しは行われません。すなわち、レコードの量が *max-record-size* の値を超えた場合、すべてのトレース項目を消去しても、それ以上のトレース項目は書き込まれません。新規トレース項目を書き込むためには、バッファ・サイズを増やし、トレース機能をオフにしてから、再度オンにします。

-l *last-buffer-size*

整数。トレースの終わりからこの数のレコードを保持します。デフォルトは 16 KB です。UNIX システムでは、推奨バッファ・サイズは 2 MB です。

path 対応するファイルが保管されるディレクトリー。

dump-filename

2 進数形式のトレース情報を含むファイルの名前。

formatted-filename

定様式トレース情報を含むファイルの名前。

例

13 ページの『障害のトレース』を参照してください。

TXVERIFY

このコマンドは、データベース・サーバーを使用可能化し、いくつかの管理コマンドを実行することによって、DB2 テキスト・エクステンダーのインストールが正常に完了したかどうかを検査します。

許可

テキスト・エクステンダーの管理者権限またはテキスト・エクステンダーのユーザー権限が必要です。詳細については、14 ページの『ユーザーに権限を提供する』を参照してください。

コマンド構文

▶—CALL PGM(QDB2TX/TXVERIFY)—▶

コマンド・パラメーター

使用上の注意

このコマンドは、クライアント・ワークステーションにおいても使用できます。

第 11 章 検索関数

DB2 テキスト・エクステンダーには、DB2 UDB for iSeries に保管されたテキスト文書を検索してその結果を処理する、SQL 関数が備わっています。この関数のパラメーターの中には、DB2 テキスト・エクステンダーに備わっている特殊タイプと呼ばれるデータ・タイプのものがあります。

この章では、DB2 テキスト・エクステンダーの SQL 関数と特殊タイプについて説明します。

DB2 テキスト・エクステンダーの QDB2TX ライブラリーの SAMPLES 物理ファイルには、TXSAMPLE というサンプル・ファイル・メンバーが用意されています。これには、9 ページの『インストール検査のためにサンプル・データベースを準備する』で説明されているサンプル表に対して実行される DB2 テキスト・エクステンダー関数の例が含まれています。このファイルは、テキスト作成関数および検索関数の構文の例を調べるのにご使用ください。

DB2 テキスト・エクステンダー特殊タイプ

特殊タイプ	ソース・データ・タイプ	コメント
DB2TEXTH	VARCHAR(60) FOR BIT DATA	<p>テキスト・ハンドル。 テキスト列に保管されるテキスト文書の索引付けに必要な情報を含む可変長ストリングです。ハンドル内の情報には、文書 ID、テキストが索引付けされるサーバーの名前、索引名、およびテキスト文書に関する情報が入っています。</p> <p>ハンドルは、DB2 テキスト・エクステンダーが作成する列の中に保管されて、個々のテキスト列と関連付けられます。</p>
DB2TEXTFH	VARCHAR(210) FOR BIT DATA	<p>ファイル・ハンドル。 DB2 UDB for iSeries の制御外に保管されたファイルである外部ファイルの索引付けに必要な情報を含む可変長ストリングです。テキスト・ハンドル内の情報には、文書 ID、テキストが索引付けされるサーバーの名前、索引名、テキスト・ファイルに関する情報、およびファイルの場所に関する情報が入っています。</p> <p>ファイル・ハンドルは、DB2 テキスト・エクステンダーが作成する列の中に保管されて、外部ファイルの個々のグループと関連付けられます。</p>

DB2 テキスト・エクステンダー特殊タイプ

特殊タイプ	ソース・データ・タイプ	コメント
DB2TEXTHLISTP	VARCHAR(16) FOR BIT DATA	ハンドル・リスト・ポインター。検索で検出されたテキスト文書に関連付けられたハンドルのリストに対するポインターです。 関数 HANDLE_LIST は、このデータ・タイプを返します。
DB2TEXTFHLISTP	VARCHAR(16) FOR BIT DATA	ハンドル・リスト・ポインター。検索で検出された外部ファイルに関連付けられたハンドルのリストに対するポインターです。

DB2 テキスト・エクステンダー関数の要約

検索関数	用途	ページ
CCSID	ハンドルから CCSID を戻す。	155
CONTAINS	特定の文書でテキストの検索を行う。	156
FILE	既存のハンドル内のファイルのパスまたは名前を戻すか、変更する。	157
FORMAT	ハンドル内の文書形式の設定を戻す、または変更する。	158
LANGUAGE	ハンドル内の言語設定値を戻す、または変更する。	159
NO_OF_DOCUMENTS ¹	ハンドル・リスト中にリストされている文書数を戻す。	160
NO_OF_MATCHES	検出された一致数を検索して戻す。	161
RANK	検出テキスト文書のランク値を検索して戻す。	162
REFINE	検索引き数および詳細化した検索引き数を受け取り、結合された検索引き数を戻す。	163
SEARCH_RESULT	指定された検索ストリングの検索結果の入った中間表を戻す。	164

DB2 テキスト・エクステンダー関数の使用例は、57 ページの『第 6 章 検索方法』に示しています。

1. これらの検索関数は、DB2 テキスト・エクステンダーの以前のリリースの機能です。これらの関数は互換性を保つために引き続きサポートされていますが、これらの機能は SEARCH_RESULT 検索関数に取って替わられています。

CCSID

CCSID 関数は、ハンドルから CCSID (データ・タイプ SMALLINT) を戻します。これは、対応するテキスト文書の索引付けに使用される CCSID パラメーターです。これについては、26 ページの『CCSID』で説明されています。個々のテキスト列に関する CCSID を設定するには、ENABLE TEXT COLUMN コマンドを使用します。

関数の構文

▶▶—CCSID—(—*handle*—)————▶▶

関数のパラメーター

handle

結果が DB2TEXTFH または DB2TEXTH タイプの値となる式です。通常は、戻される CCSID 設定が含まれているハンドル列の名前です。

CONTAINS

CONTAINS 関数は、特定のテキスト文書内のテキストを検索します。その文書の中にテキストが含まれている場合、INTEGER 値 1 が戻されます。含まれていない場合は 0 が戻されます。

関数の構文

▶▶—CONTAINS—(—*handle*—,—*search-argument*—)—▶▶

関数のパラメーター

handle

結果が DB2TEXTFH または DB2TEXTH タイプの値となる式です。通常は、検索したいテキスト文書のハンドルが含まれているハンドル列の名前です。

search-argument

検索する用語を含んだ LONG VARCHAR タイプのストリングです。
165 ページの『第 12 章 検索引き数の構文』を参照してください。

FILE

FILE 関数は以下のいずれかを実行します。

- パスまたはファイル名をハンドル内に戻す。
- ハンドル内のパスまたはファイル名を変更して、そのパスとファイル名を戻す。

戻されるハンドルは、DB2TEXTFH タイプの値です。

関数の構文

```
▶▶FILE—(—handle—)—————▶▶  
▶▶FILE—(—handle—, —file-name—)—————▶▶
```

関数のパラメーター

handle

結果が DB2TEXTFH タイプの値となる式です。通常は、戻されるファイル名が含まれているハンドル列の名前です。

file-name

ハンドルに関連付けられる外部ファイルの新しい絶対パスおよびファイル名を指定する VARCHAR(150) タイプのストリングです。このパスは、たとえば、LAN ドライブまたは NFS 取り付けドライブなどです。ファイル・アクセス許可により、DB2 UDB for iSeries インスタンス所有者によるファイルのアクセスを許可する必要があります。

FORMAT

FORMAT 関数は、以下のいずれかを実行します。

- ハンドル内に指定された文書形式を戻す。
- 文書のハンドル内の形式指定を変更し、その変更されたハンドルを戻す。

戻される文書形式は、VARCHAR(30) タイプのストリングです。戻されるハンドルは DB2TEXTFH または DB2TEXTH タイプです。

これは、対応するテキスト文書の索引付けに使用される形式パラメーターです。サポートされる文書形式は、23 ページの『サポートされる文書形式』に列挙されています。

関数の構文

(1)
 ►► `FORMAT` (*—handle—*)

注:

- 1 VARCHAR(30) タイプの形式値を戻します。

(1)
 ►► `FORMAT` (*—handle—, —format—*)

注:

- 1 DB2TEXTFH または DB2TEXTH タイプのハンドルを戻します。

関数のパラメーター

handle

結果が DB2TEXTFH または DB2TEXTH タイプの値となる式です。通常は、形式設定が戻されるハンドル列の名前か、または形式設定がセットされるハンドル列の名前です。

format

データ・タイプ VARCHAR(30) の新規の文書形式の設定です。

format を指定すると、ハンドル内に文書形式が設定されます。この場合、形式設定の代わりに、ハンドルが戻されます。

LANGUAGE

LANGUAGE 関数は、以下のいずれかを実行します。

- ハンドル内に指定されている文書言語を戻す。
- 文書のハンドル内の言語指定を変更し、その変更されたハンドルを戻す。

戻される文書言語は、VARCHAR(30) タイプのstringです。戻されるハンドルは DB2TEXTFH または DB2TEXTH タイプです。

これは、対応するテキスト文書の索引付けに使用される言語パラメーターです。サポートされる言語は、40 ページの表 4 に記載されています。

関数の構文

(1)
 ►► LANGUAGE (—*handle*—) ►►

注:

- 1 VARCHAR(30) タイプの言語値を戻します。

(1)
 ►► LANGUAGE (—*handle*—, —*language*—) ►►

注:

- 1 DB2TEXTFH または DB2TEXTH タイプのハンドルを戻します。

関数のパラメーター

handle

結果が DB2TEXTFH または DB2TEXTH タイプの値となる式です。通常は、戻される言語設定が含まれているハンドル列の名前です。

language

データ・タイプ VARCHAR(30) の新規の文書言語の設定です。

language を指定すると、ハンドル内にこの文書言語が設定されます。この場合、言語設定の代わりにハンドルが戻されます。

NO_OF_DOCUMENTS

NO_OF_DOCUMENTS 関数は、検索により検出されたテキスト文書のリストにある項目数を示す INTEGER 値を返します。戻される値は、ハンドルのリスト内で見つかる項目の数です。

ヒント

この関数は、DB2 テキスト・エクステンダーの以前のリリースの機能です。これは互換性を保つために引き続きサポートされていますが、この機能は SEARCH_RESULT 検索関数に取って替わられています。

関数の構文

▶▶—NO_OF_DOCUMENTS—(—*handle-list*—)————▶▶

関数のパラメーター

handle-list

結果が DB2TEXTHLISTP または DB2TEXTFHLISTP タイプの値となる式です。関数 HANDLE_LIST によって戻されます。

検索で検出された文書のハンドルのリストに対するポインターです。

HANDLE_LIST および NO_OF_DOCUMENTS 関数は、同じ SQL ステートメント内になければなりません。リストはステートメントの効力範囲内でのみ存在するためです。

NO_OF_MATCHES

NO_OF_MATCHES は、テキスト文書内を検索し、その結果として文書ごとに一致した数を示す INTEGER 値を戻すことができます。

関数の構文

▶▶—NO_OF_MATCHES—(—handle—,—search-argument—)————▶▶

関数のパラメーター

handle

結果が DB2TEXTFH または DB2TEXTH タイプの値となる式です。通常は、検索したいテキスト文書のハンドルが含まれているハンドル列の名前です。

search-argument

検索する用語を含んだ LONG VARCHAR タイプの文字列です。
165 ページの『第 12 章 検索引き数の構文』を参照してください。

RANK

RANK は、テキスト文書内を検索し、検出された個々の文書のランク値を戻すことができます。この値は、検索引き数により検出文書が記述される程度を示します。

RANK は、0 ~ 1 の DOUBLE 値を戻します。ランク値は、検出された文書が検索基準を満たす度合いを、他の検出文書と比べて示す絶対値です。この値は、文書内での一致検出数を文書サイズ比で示すものです。

関数の構文

▶▶—RANK—(—*handle*—,—*search-argument*—)—————▶▶

関数のパラメーター

handle

結果が DB2TEXTFH または DB2TEXTH タイプの値となる式です。通常は、検索したいテキスト文書のハンドルが含まれているハンドル列の名前です。

search-argument

検索する用語を含んだ LONG VARCHAR タイプのストリングです。
165 ページの『第 12 章 検索引き数の構文』を参照してください。

REFINE

REFINE 関数は、2 つの検索引き数を取り、それらをブール演算子 AND で 1 つに結合した LONG VARCHAR タイプの検索引き数を戻します。

関数の構文

▶▶—REFINE—(—*search-argument*—,—*search-argument*—)————▶▶

関数のパラメーター

search-argument

検索する用語を含んだ LONG VARCHAR タイプのストリングです。
165 ページの『第 12 章 検索引き数の構文』を参照してください。

検索引き数に、検索パラメーター IS ABOUT、THESAURUS、または EXPAND を含めることはできません。

SEARCH_RESULT

SEARCH_RESULT 関数は、検索結果を中間表に戻します。この関数は、SQL ステートメントの FROM 文節で使用できます。

戻される表の構造は次のとおりです。

列名	データ・タイプ
HANDLE	DB2TX,DB2TEXTH
NUMBER_OF_MATCHES	INTEGER
RANK	DOUBLE

中間表の、選択された列の値のみが生成されます。Select count(*) を使用すると、HANDLE 列のみが生成されます。ランク値の計算では多くのシステム・リソースが消費されるため、ランク値が必要ない場合は中間表のランク値を選択しないでください。

ラージ表の処理は、この関数の方が CONTAINS または RANK より高速です。

関数の構文

▶▶SEARCH_RESULT(—handle—,—search-argument—)▶▶

関数のパラメーター

handle

検索される文書を含む列に対応したハンドル列の名前。

search-argument

検索する用語を含んだ LONG VARCHAR タイプのストリングです。
165 ページの『第 12 章 検索引き数の構文』を参照してください。

例

例については、75 ページの『検索パフォーマンスの向上』を参照してください。

第 12 章 検索引き数の構文

検索引き数とは、テキスト文書内の用語を検索する場合に指定する条件です。検索引き数は、1 つまたは複数の検索項目と検索パラメーターとから成ります。

検索引き数の例は、63 ページの『検索引き数の指定』に記載してありますが、これは `txsample.udf` と呼ばれるファイルに格納されています。これには、9 ページの『インストール検査のためにサンプル・データベースを準備する』で説明されているサンプル表に対して実行される DB2 テキスト・エクステンダー関数の例が含まれています。

検索引き数を使用する DB2 テキスト・エクステンダー関数には、以下のものがあります。

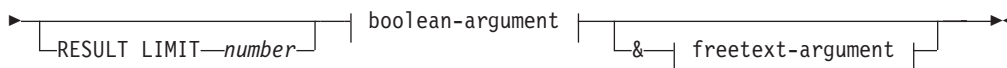
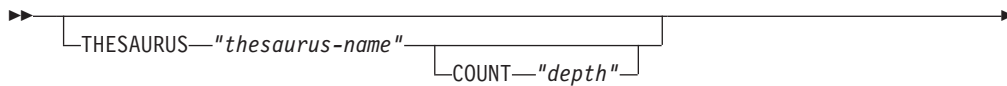
- **CONTAINS**。この関数は、検索引き数を使って特定のテキスト文書にあるテキストを検索します。その文書の中にテキストが含まれている場合、INTEGER 値 1 が戻されます。含まれていない場合は 0 が戻されます。
- **NO_OF_MATCHES**。この関数は、検索引き数を使ってテキスト文書中を検索します。この関数により、1 文書当たりの一致数を示す INTEGER 値が戻されます。
- **RANK**。この関数は、検索引き数を使ってテキスト文書中を検索します。これは検出された文書ごとに、その文書が検索引き数により記述される程度を示す値を戻します。
- **REFINE**。この関数は、2 つの検索引き数を結合してタイプ `LONG VARCHAR` の検索引き数を返します。これは、元の 2 つの検索引き数をブール演算子 `AND` で連結したものです。

検索引き数を使用する API 関数には、以下のものがあります。

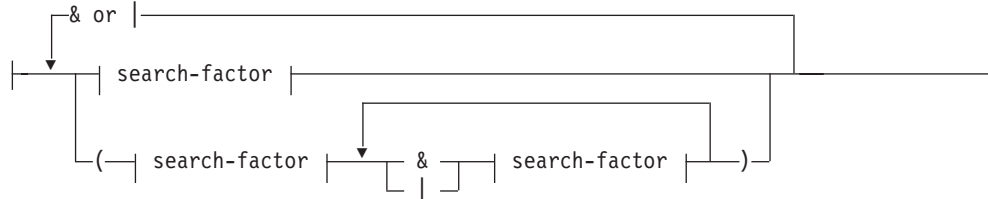
- **DesGetBrowseInfo**。この関数は、検索引き数を使って、ハンドルによって識別されるテキスト中を検索します。これは、`DesStartBrowseSession` が用語を強調表示するために必要となるブラウザ情報に対するポインターを戻します。
- **DesGetSearchResultTable**。この関数は、テキスト列によって識別されるテキスト文書を検索するために検索引き数を使用します。検出されたテキスト項目のハンドル・データは、結果表に書き込まれます。また、ランクや一致数に関するブラウザ情報も結果表に書き込むことができます。

検索引き数

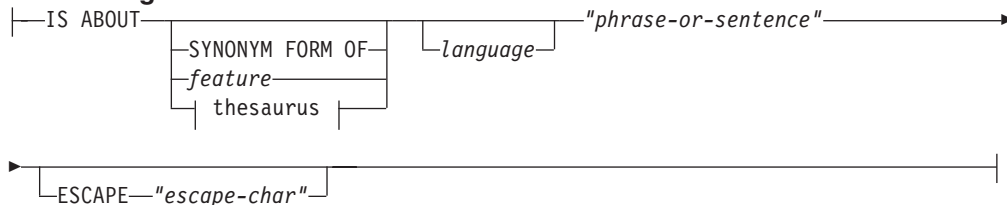
検索引き数構文



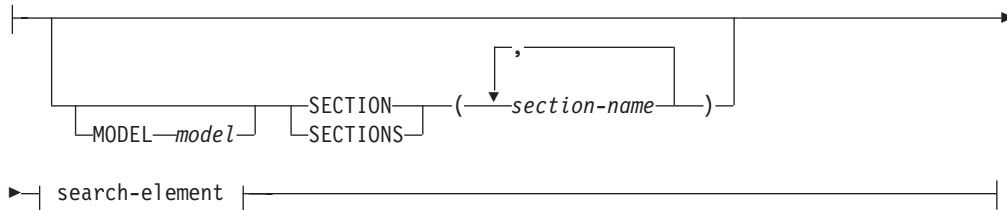
boolean-argument:



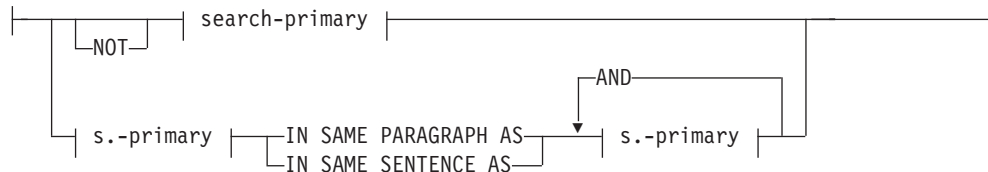
freetext-argument:



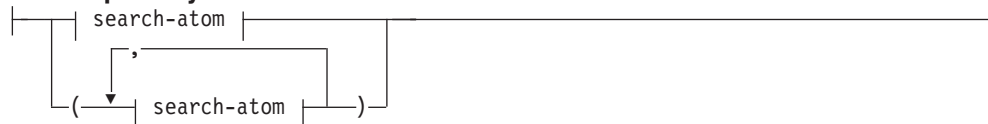
search-factor:



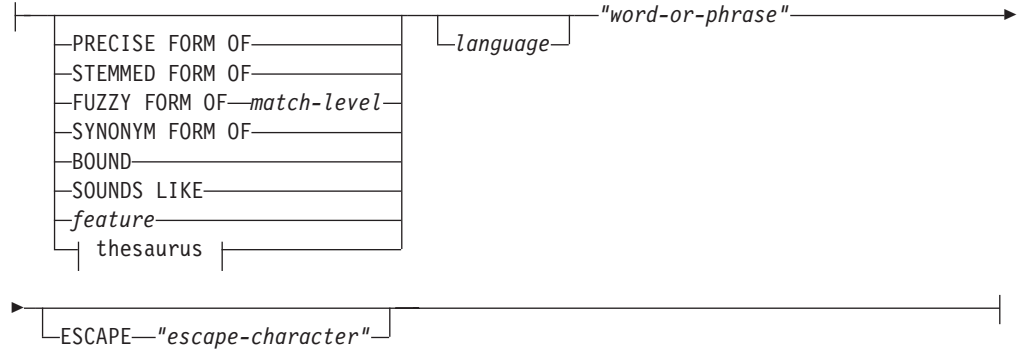
search-element:



search-primary:



search-atom:



thesaurus (THESAURUS が指定されている場合):



例

検索引き数の例は、63 ページの『検索引き数の指定』に記載されています。

検索パラメーター

IS ABOUT

ユーザーがフリー・テキスト検索引き数、すなわち、検出したい概念を記述する自然言語の句または文を指定できるオプションです。70 ページの『フリー・テキストと混成検索』を参照してください。

MODEL *model*

検索用語に使用する文書モデルの名前を指定するキーワードです。文書モデルは文書の構造を記述したものです。記述される文書には識別可能なセクションが含まれており、これらのセクションの内容を個別に検索できます。

モデル名は、37 ページの『構造化文書 (セクション・サポート)』での作業』で説明されている文書モデル・ファイル中に指定しなければなりません。ワイルドカード文字を使用してモデル名をマスクできます。

モデルを指定しない場合は、索引作成時に指定されたデフォルトのモデルが使用されます。

SECTION(S) *section-name*

検索が限定される 1 つまたは複数のセクションを指定するキーワードです。セクション名は、37 ページの『構造化文書 (セクション・サポート)』での作業』で説明されている文書モデル・ファイル中のモデル中に指定しなければなりません。ワイルドカード文字 % および _ を使用してセクション名をマスクできます。Ngram 索引の場合、ネストされていないセクション名にはワイルドカード検索しか使用できない点に注意してください。

セクションは、次のように他のセクション内にネストできます。

play/Act/Title=play/act/title

検索引き数の構文

制約事項: ネストされたセクションを検索できるのは、XML 形式が利用可能な列に保管された文書のみです。Ngram 索引の場合は、1 つのセクション名のみを検索でき、XML 形式はサポートされていません。

attribute-value

先行する比較演算子と一緒に使用される、その前のセクション・リストにリストされている属性に関係する値。属性の値が特定の範囲内に収まるよう要求する照会の場合は、1 つの属性条件中に 2 つの比較演算子を使用できません。

同一の条件中に同じ種類の比較演算子を組み合わせて使用する (最初に \geq を使用し、次に $>$ を使用するなど) ことは無効です。2 つの比較演算子と $=$ 演算子を指定することも無効です。

- $=$ 索引付き文書の属性と、この演算子の後に指定した属性値との等価比較を要求します。
- \geq 索引付き文書の属性と、この演算子の後に指定した属性値との「より大か等しい」比較を要求します。
- $>$ 索引付き文書の属性と、この演算子の後に指定した属性値との「より大きい」比較を要求します。
- \leq 索引付き文書の属性と、この演算子の後に指定した属性値との「より小か等しい」比較を要求します。
- $<$ 索引付き文書の属性と、この演算子の後に指定した属性値との「より小さい」比較を要求します。

セクションは、次のように他のセクション内にネストできます。

```
play/Act/Title=play/act/title
```

制約事項: ネストされたセクションを検索できるのは、XML 形式が利用可能な列に保管された文書のみです。Ngram 索引の場合は、1 つのセクション名のみを検索でき、XML 形式はサポートされていません。

THESAURUS *thesaurus-name*

検索用語の拡張に使用するシソーラスの名前を指定するのに使用されるキーワードです。シソーラス名は、シソーラス・コンパイラ TXTHESC または TXTHESN を使用してコンパイルされたシソーラスのファイル名 (拡張子なし) です。サンプル・ディレクトリーには、デフォルト・シソーラス *dessthes* および *desnthes* (*desnthes* は Ngram シソーラス) が保管されています。また、そのファイルのパス名も指定できます。デフォルト・パス名は辞書のパスです。

COUNT *depth*

シソーラスに含まれる用語のレベル数 (深さ) を指定するために使用されるキーワードです。これらのレベルは、指定された関係の検索用語を拡張するために使用されます。このキーワードを指定しない場合は、カウント 1 と想定されます。

RESULT LIMIT *number*

結果リストに戻される項目の最大数を指定するために使用されるキーワードです。 *number* は、1 から 32767 の値です。フリー・テキスト検索が使用される場合には、検索結果リストは、完全な検索結果リストに関してのみランクされます。その他の場合には、限定検索結果がリストの項目からのみランクされます。

EXPAND *relation*

INSTANCE などのように、TERM OF に指定された検索用語と、その検索用語を拡張するために使用されるシソーラス内の用語との関係の指定に使用されるキーワードです。関係の名前はシソーラス内で使用されている関係に対応していなければなりません。 206 ページの『シソーラスの概念』を参照してください。

Ngram シソーラスの場合は、 213 ページの『Ngram シソーラスの作成』で説明されている member-relation 名を使用してください。ユーザー定義のメンバー関係の場合は、 :RELATION *n* (*n* は :RELATED (*number*) で指定されているメンバー関係番号) を使用してください。

TERM OF "*word-or-phrase*"

シソーラスから他の検索用語が追加される検索用語または複数語検索用語です。

search-factor

他のオペランドと結合して検索引き数を形成することができるオペランドです。評価は左から右へと行われます。論理 AND (&) 演算子を使用すると、論理 OR (|) 演算子の場合よりも強いバインドが行われます。例:

"passenger" & "vehicle" | "transport" & "public"

これは次のように評価されます。

("passenger" & "vehicle") | ("transport" & "public")

検索を行うには次のようにします。

"passenger" & ("vehicle" | "transport") & "public"

この場合、上記のように括弧を付けなければなりません。

NOT search-primary

特定の用語を含むテキスト文書を検索から除外するための演算子です。

検索項目 (search factor) 内に NOT を使用する場合は、SYNONYM FORM OF キーワードは使用できません。

search-primary IN SAME PARAGRAPH AS search-primary

同一段落内にある用語の組み合わせを検索するためのキーワードです。

次の検索引き数を指定すると、「air」という用語が同一段落内にある場合のみ、「traffic」を含むテキスト文書が検出されます。

"traffic" IN SAME PARAGRAPH AS "air"

IN SAME PARAGRAPH AS キーワードは、検索項目に NOT を使用する場合は指定できません。

検索引き数の構文

search-primary IN SAME SENTENCE AS search-primary

同一文にある用語の組み合わせを検索するためのキーワードです。これは IN SAME PARAGRAPH AS と類似しています。

AND search-primary

同一文または同一段落内で検索を行うために、複数の基本検索語を結合するキーワードです。

次の検索引き数では、同一文で「forest」、「rain」、「erosion」、および「land」が検索されます。

```
"forest" IN SAME SENTENCE AS "rain" AND "erosion" AND "land"
```

search-atom

一連の検索アトム (search atom) の間にコンマを入れると、検索アトムのいずれか 1 つで用語が検出された場合に検索が正常に行われます。どの検索アトムにも語または句が最低 1 つは含まれていなければなりません。

次のステートメントは、1 つまたは複数の検索引き数が検出された場合に真となります。

```
CONTAINS (mytexthandle, '( "text",  
                           "graphic",  
                           "audio",  
                           "video" )' ) = 1
```

PRECISE FORM OF、STEMMED FORM OF、FUZZY FORM OF、SYNONYM FORM OF、BOUND

171 ページの表 7 は、種々のタイプの索引に該当するオプションを示しています。たとえば、基本語形索引の場合、PRECISE FORM OF 以外のすべてのオプションが有効です。PRECISE FORM OF を指定すると、これは無視されてデフォルト値が使用されます。

検索用語処理については、171 ページの表 8 で詳しく説明しています。

表 7. 基本語形オプション

検索アトム・キーワード	索引タイプ				
	基本語形	一致語句	正規化された一致語句	Ngram	大文字小文字が区別される Ngram
PRECISE FORM OF		X	X		O
STEMMED FORM OF	X			O	O
FUZZY FORM OF				O	O
IS ABOUT	O	O	O		
SYNONYM FORM OF	O	O	O		
EXPAND	O	O	O		
SOUNDS LIKE	O	O	O		
IN SAME SENTENCE AS	O	O	O	O	O
IN SAME PARAGRAPH AS	O	O	O	O	O
BOUND				O	O

X=デフォルト設定 O=使用可能な機能

表 8. Ngram 索引の検索用語オプション

検索アトム・キーワード	用語検索処理				
	大文字小文字		語幹形への 変換	一致	
	区別あり	区別なし		完全	ファジー
PRECISE FORM OF	大文字小文字が区別される場合	X		X	
STEMMED FORM OF		X	X		
FUZZY FORM OF		X			X

X=デフォルト設定

索引タイプにおいて使用可能でないキーワードを使用すると、それは無視され、代わりにデフォルト・キーワードが使用されるか、メッセージが戻されます。

PRECISE FORM OF

PRECISE FORM OF に続く語 (または句内のすべての語) について、まずそれを語幹形式に変換するのではなく、入力されたとおりの語として検索を行います。一致語句索引の場合、この形式の検索では、大文字小文字が区別されます。つまり、大文字を使うか小文字を使うかが重要な意味を持ちます。たとえば、mouse を検索する場合、「Mouse」は検出されません。

これは、一致語句索引でのデフォルト・オプションです。正規化された一致語句索引の場合、デフォルトの検索形式では大文字小文字が区別されません。このキーワードを基本語形索引で指定すると、無視されて STEMMED FORM OF が使用されます。

STEMMED FORM OF

STEMMED FORM OF に続く語 (または句内のすべての語) を語幹

検索引き数の構文

形式に変換してから、検索を行います。この方式の検索では、大文字小文字の区別は行われません。たとえば `mouse` を検索すると、「`Mouse`」も検出されます。

語を語幹形に戻す方法は、言語によって異なります。

例: `programming computer systems` は、米国英語用辞書を使用すると `program compute system` に置き換えられ、英国英語用辞書を使用すると、`programme compute system` に置き換えられます。

この検索句では、「`programmer computes system`」、「`program computing systems`」、「`programming computer system`」などを検出することができます。

これは基本語形索引の場合のデフォルト・オプションです。このキーワードを一致語句索引で指定すると、無視されて代わりに `PRECISE FORM OF` が使用されます。

FUZZY FORM OF

検索用語と類似した用語の検索を行う「ファジー」検索を行うためのキーワードです。これは、光学式文字認識 (OCR) プログラムによって作成された文書の検索には特に有効です。そのような文書には、つづりを誤った語が含まれていることが多いためです。たとえば、`economy` という語は、OCR プログラムでは、`econony` として認識される可能性があります。 *match-level*: 類似の程度を示す 1 ~ 5 の整数です。5 は 1 より類似度が高いことを示します。

SYNONYM FORM OF

`SYNONYM FORM OF` に続く語または句がその同義語と一緒に検索されるようにするキーワードです。同義語は、*language* で指定された辞書か、またはデフォルトの辞書に備えられています。

句の同義語とは、元の句を成す各語をその同義語の 1 つと置き換えることによって得られる同義語の組み合わせをすべて含む別の句です。このとき、語順は元の句と同じです。

このキーワードを一致語句索引で指定すると、無視されて代わりに `PRECISE FORM OF` が使用されます。

このキーワードは、検索項目内に `NOT` を使用した場合、または検索する語または句にマスキング文字が含まれている場合には指定できません。

BOUND

韓国語 `CCSID` を使用する文書を検索するためのキーワードです。語句の境界を重視する検索を行います。 *language* を指定しても、それは無視され、韓国語と想定されます。

language

索引付けや検索の最中に、テキスト文書の言語的な処理で使用する辞書を指定する変数です。この変数は、基本語形索引ばかりでなく、一致語句索引にも適用されます。これは、一致語句索引が停止ワードを処理するための辞書を使用するためです。

言語的な処理には、同義語処理と語幹処理が含まれます。

サポートされる言語は、40 ページの表 4 に列挙されています。

注: 米国英語で書かれていない文書を検索する場合は、検索引き数 *regardless of the default language* に言語を指定しなければなりません。

"word-or-phrase"

検索する語または句です。1 つの語の中で使用できる文字は、言語によって異なります。また、語を分離文字で分離する必要があるかどうか、言語によって異なります。英語やほとんどの言語の場合、句内のそれぞれの語は空白文字で分離しなければなりません。

一致語句検索または基本言語検索。DB2 テキスト・エクステンダーは、語または句の一致語句書式かその変化形のどちらかを使用して検索することができます。171 ページの表 7 に記載されているオプションの 1 つを指定しない場合は、使用している索引のタイプに応じてデフォルトの基本語形オプションが使用されます。

二重引用符を含む文字ストリングを検索するには、二重引用符を 2 回入力してください。たとえば、テキスト "wildcard" character を検索するには、次のように指定します。

```
""wildcard"" character"
```

マスキング文字。 語には以下のマスキング文字を含めることができます。

_ (下線)

任意の単一文字を表します。

% (パーセント)

任意の個数の任意の文字を表します。語が 1 つの % から成っている場合、それは任意の長さのオプション語を表します。

% を 1 つ使用してオプション語を表す場合を除き、語をマスキング文字のみで構成することはできません。

マスキング文字を使用する場合は、SYNONYM FORM OF、*feature*、または THESAURUS を使用することはできません。

ESCAPE "escape-character"

エスケープ文字とは、次に続く文字を検索対象の文字として識別し、マスキング文字として使用する文字ではないと識別するための文字です。

例: エスケープ文字 が \$ である場合、\$%、\$_、および \$\$ は、それぞれ %、_、および \$ を表します。前に \$ が付かない % および _ 文字は、マスキング文字を表しています。また、エスケープ文字は、単一文字でなければならぬ点にも注意してください。

規則と制約事項の要約:

ブール演算

OR の後で NOT を使用することはできません。

FUZZY FORM OF

先頭の 3 文字は一致しなければなりません。検索アトム内の語にマスキング文字が含まれている場合は使用できません。NOT と一緒に使用することはできません。Ngram 索引でのみ使用できます。

IN SAME PARAGRAPH AS

検索係数内で NOT を指定する場合は使用できません。

IN SAME SENTENCE AS

検索係数内で NOT を指定する場合は使用できません。

基本語形索引

PRECISE FORM OF は使用できません。デフォルトとして STEMMED FORM OF を使用します。マスキング文字を使用できます。検索では、大文字小文字が区別されません。

マスキング文字

SYNONYM FORM OF と THESAURUS が使用されないようにします。

Ngram 索引

マスキング文字を使用できますが、前に英数字以外の文字を付けることはできません。索引で大文字小文字の区別が可能でない場合および PRECISE FORM OF が使用されない場合は、検索では大文字小文字を区別しません。

NOT SYNONYM FORM OF、IN SAME PARAGRAPH AS、および IN SAME SENTENCE AS が使用されないようにします。

PRECISE FORM OF

基本語形索引では無視されます。

一致語句索引

STEMMED FORM OF と SYNONYM FORM OF が使用されないようにします。デフォルトとして PRECISE FORM OF を使用します。マスキング文字を使用できます。検索では、大文字小文字が区別されます。

STEMMED FORM OF

一致語句索引の場合は無視されますが、英語の文書を含む正規化された一致語句索引の場合は使用できます。

SYNONYM FORM OF

検索アトム内の語にマスキング文字が含まれている場合は使用できません。NOT と一緒に使用することはできません。一致語句索引では使用できません。

第 13 章 検索およびブラウズのための API 関数

DB2 テキスト・エクステンダーには、テキスト文書を検索するための C 関数、および検出された文書をブラウズ (表示) するための C 関数があります。これらの関数は、DB2 テキスト・エクステンダー・アプリケーション・プログラミング・インターフェース (API) を構成しています。この章では、API 関数をアルファベット順に説明します。

91 ページの『第 8 章 検索およびブラウズのための API 関数の使用法』では、この関数の概要について説明し、さらにこの関数を組み合わせた使用法について述べています。

関数	用途	ページ
DesCloseDocument	DesOpenDocument によって割り当てられたストレージを解放します。	176
DesEndBrowseSession	ブラウズ・セッションをクローズし、DesStartBrowseSession によって割り当てられたストレージを解放します。	177
DesFreeBrowseInfo	DesGetBrowseInfo によって割り当てられたストレージを解放します。	178
DesGetBrowseInfo	検索引き数を使用しているテキストを文書内で検索し、ブラウズ情報を作成します。	179
DesGetMatches	文書ハンドルで記述されたテキスト文書の強調表示情報に対するポインターを戻します。強調表示情報はデータ・ストリームです。これには、テキスト文脈 (最低 1 つの段落) と、その文脈内のテキストを強調表示させるための情報が含まれています。	182
DesGetSearchResultTable	指定されたテキスト列内でテキスト文書を検索するための検索引き数を取得し、その結果をユーザーが作成した表に保管します。ブラウズ情報を戻すことも可能です。	187
DesOpenDocument	ブラウズ・セッション・ポインター、ハンドル、および検出された用語を強調するのに使用される言語処理タイプを示す DES_FAST または DES_EXTENDED を受け取ります。文書テキストおよび強調表示情報を入手するためのハンドルに対応するテキスト文書を作成し、DesGetMatches を反復呼び出しするための文書ハンドルを戻します。	191
DesStartBrowseSession	DesGetBrowseInfo のブラウズ情報を使用してブラウズ・セッションをオープンし、他のブラウズ関数が使用するブラウズ・セッション・ハンドルを戻します。	193

ヒント

API 関数の多くには接続ハンドル (hdbc) が必要です。このハンドルは SQLConnect 関数を使用して指定しなければなりません。それによって組み込み SQL プログラムから DB2 テキスト・エクステンダーを呼び出せなくなることはありません。「DB2 CLI の手引きおよび解説書」では、CLI ステートメントと組み込み SQL ステートメントとの組み合わせ方法が説明されています。

DesCloseDocument

目的

DesOpenDocument でオープンされたテキスト文書をクローズし、文書テキストと強調表示情報が戻される際に、割り当てられていたストレージを解放します。

構文

```
DESRETURN  
DesCloseDocument  
(DESBROWSESESSION    BrowseSession,  
 DESHANDLE             DocumentHandle);
```

関数引き数

表 9. DesCloseDocument 引き数

データ・タイプ	引き数	用法	説明
DESBROWSESESSION	<i>BrowseSession</i>	入力	ブラウザ・セッション・ハンドルです。
DESHANDLE	<i>DocumentHandle</i>	入力	オープンされたテキスト文書を識別する DesOpenDocument により戻されたハンドルです。

戻りコード

RC_SUCCESS

RC_INVALID_PARAMETER

RC_INVALID_SESSION

RC_SE_INCORRECT_HANDLE

RC_SE_IO_PROBLEM

RC_SE_LS_FUNCTION_FAILED

RC_SE_NOT_ENOUGH_MEMORY

RC_SE_REQUEST_IN_PROGRESS

RC_SE_WRITE_TO_DISK_ERROR

制限

この関数は、DesOpenDocument を呼び出してテキスト文書をオープンした後でしか呼び出すことができません。

DesEndBrowseSession

目的

DesStartBrowseSession によって開始されたブラウザ・セッションを終了し、そのブラウザ・セッションに割り当てられたストレージを解放します。

構文

```
DESRETURN
DesEndBrowseSession
(DESBROWSESESSION BrowseSession);
```

関数引き数

表 10. DesEndBrowseSession 引き数

データ・タイプ	引き数	用法	説明
DESBROWSESESSION	<i>BrowseSession</i>	入力	ブラウザ・セッション・ハンドルです。

使用

この関数では、DesGetBrowseInfo によってブラウザ・セッション用に割り当てられた記憶装置は解放されません。このストレージには、その後もその他のブラウザ・セッションで使用できるブラウザ情報が含まれています。このストレージを解放するには、DesFreeBrowseInfo を呼び出してください。

戻りコード

RC_SUCCESS

RC_INVALID_SESSION

RC_INVALID_PARAMETER

RC_SE_UNEXPECTED_ERROR

制限

この関数は、DesStartBrowseSession 呼び出しによってブラウザ・セッションを開始した後でしか呼び出すことができません。

DesFreeBrowseInfo

目的

DesGetBrowseInfo によってブラウザ情報に割り当てられたストレージを解放します。

構文

```
DESRETURN  
DesFreeBrowseInfo  
(DESBROWSEINFO BrowseInfo);
```

関数引き数

表 11. *DesFreeBrowseInfo* arguments

データ・タイプ	引き数	用法	説明
DESBROWSEINFO	<i>BrowseInfo</i>	入力	ブラウザ情報です。

戻りコード

RC_SUCCESS

RC_INVALID_PARAMETER

制限

この情報は、DesGetBrowseInfo 呼び出しによってブラウザ情報用にストレージを割り当てた後でしか呼び出すことができません。

DesGetBrowseInfo

目的

ハンドルによって識別されるテキストの内部を検索するための検索引き数を受け取ります。これは、DesStartBrowseSession が検出用語を強調表示するために必要となるブラウザ情報に対するポインタを戻します。

構文

```

DESRETURN
DesGetBrowseInfo
(SQLHDBC          hdbc,
SQLCHAR          *pHandle,
DESUSHORT        HandleLength,
char             *pSearchArgument,
DESSMALLINT      ArgumentLength,
DESBROWSEINFO    *pBrowseInfo,
DESMESSAGE       *pErrorMessage);

```

関数引き数

表 12. DesGetBrowseInfo 引き数

データ・タイプ	引き数	用法	説明
SQLHDBC	<i>hdbc</i>	入力	データベース接続ハンドルです。
SQLCHAR *	<i>pHandle</i>	入力	データベースから抽出されるハンドルへのポインタです。
DESUSHORT	<i>HandleLength</i>	入力	<i>pHandle</i> の長さです。DES_NTS はここでは使用できません。
char *	<i>pSearchArgument</i>	入力	検出したい情報を指定するテキスト検索引き数に対するポインタです。
DESSMALLINT	<i>ArgumentLength</i>	入力	<i>pSearchArgument</i> の長さ (ヌル・バイトの終止符を含まない)、または DES_NTS です。
DESBROWSEINFO *	<i>pBrowseInfo</i>	出力	テキスト文書のブラウザのために必要なデータを含むブラウザ情報に対するポインタです。このポインタは DesStartBrowseSession に渡されます。
DESMESSAGE *	<i>pErrorMessage</i>	出力	処理系定義のメッセージ・テキスト。エラーが発生した場合は、DB2 テキスト・エクステンダーによりエラー・コードとエラー・メッセージが戻されます。アプリケーション・プログラムによってサイズ DES_MAX_MESSAGE_LENGTH のバッファが割り当てられます。 <i>pErrorMessage</i> がヌル・ポインタである場合、エラー・メッセージは戻されません。

使用

DesGetBrowseInfo を呼び出す前に、アプリケーション・プログラムでデータベースへの接続を設定しておかなければなりません。

DesGetBrowseInfo API 関数

検索引き数に対するポインタの場合、*SQLCHAR** ではなく *char** が使用されます。これは、パラメータ値がデータベースから得られない可能性があるためです。

SQL データ・タイプと C データ・タイプ間のマッピングでは、ハンドルとして SQL 記号名 *SQL_VARBINARY* を使用する必要があります。Handle 値の C 表示を指すホスト変数のタイプは *SQLCHAR** です。

DB2 テキスト・エクステンダーにより、ブラウズ情報用にストレージが割り当てられます。アプリケーション・プログラムでは、*DesFreeBrowseInfo* を呼び出してストレージと関連するリソースを解放しなければなりません。

Handle 値はビット・データで、'¥0' 文字をいくつか含んでいるため、*pHandle* の長さを指定する必要があります。

pSearchArgument の検索引き数については、165 ページの『第 12 章 検索引き数の構文』で説明されています。

戻りコード

RC_SUCCESS
RC_NO_BROWSE_INFO

RC_ALLOCATION_ERROR
RC_FILE_IO_PROBLEM
RC_INTERNAL_ERROR
RC_INVALID_PARAMETER
RC_PARSER_INVALID_ESCAPE_CHARACTER
RC_PARSER_INVALID_USE_OF_ESCAPE_CHAR
RC_PARSER_SYNTAX_ERROR
RC_SE_COMMUNICATION_PROBLEM
RC_SE_EMPTY_INDEX
RC_SE_EMPTY_QUERY
RC_SE_FUNCTION_DISABLED
RC_SE_FUNCTION_IN_ERROR
RC_SE_INCORRECT_HANDLE
RC_SE_INDEX_DELETED
RC_SE_INDEX_NOT_ACCESSIBLE
RC_SE_INDEX_SUSPENDED
RC_SE_INSTALLATION_PROBLEM
RC_SE_IO_PROBLEM
RC_SE_MAX_NUMBER_OF_BUSY_INDEXES
RC_SE_MAX_OUTPUT_SIZE_EXCEEDED
RC_SE_NOT_ENOUGH_MEMORY
RC_SE_PROCESSING_LIMIT_EXCEEDED
RC_SE_QUERY_TOO_COMPLEX
RC_SE_SERVER_BUSY
RC_SE_SERVER_CONNECTION_LOST
RC_SE_SERVER_NOT_AVAILABLE
RC_SE_UNEXPECTED_ERROR

RC_SE_UNKNOWN_INDEX_NAME
RC_SE_UNKNOWN_SERVER_NAME
RC_SE_WRITE_TO_DISK_ERROR

重要: 以下の戻りコードは、関数は結果を返したが、予想とは異なることを示しています。

RC_SE_CONFLICT_WITH_INDEX_TYPE
RC_SE_DICTIONARY_NOT_FOUND
RC_SE_STOPWORD_IGNORED
RC_SE_UNKNOWN_SECTION_NAME
RC_SE_DOCMOD_READ_PROBLEM

制限

この関数は、データベースに接続して、DB2 テキスト・エクステンダー関数を使用してデータベースからハンドルを抽出した後でのみ、呼び出すことができます。

DesGetMatches

目的

文書ハンドルで記述されるテキスト文書の強調表示情報を含むデータ・ストリームを戻します。183 ページの『データ・ストリーム構文』を参照してください。強調表示情報には、テキスト文脈 (最低 1 つの段落) およびその文脈内のテキスト強調表示のための情報が含まれます。

DesGetMatches は、データ・ストリームの一部だけを戻します。これは、その出力構造体の一部の長さを示します。

DesGetMatches を連続して呼び出すと、テキスト文書全体の内容入手できます。テキスト文書の終わりに達すると、RC_SE_END_OF_INFORMATION が返されます。

構文

```

DESRETURN
DesGetMatches
(
  DESBROWSESESSION BrowseSession,
  DESHANDLE DocumentHandle,
  DESMATCHINFO *pMatchInfo,
  DESULONG *pMatchInfoLength,
  DESMESSAGE *pErrorMessage);

```

関数引き数

表 13. DesGetMatches 引き数

データ・タイプ	引き数	用法	説明
DESBROWSESESSION	<i>BrowseSession</i>	入力	ブラウザ・セッション・ハンドルです。
DESHANDLE	<i>DocumentHandle</i>	入力	DesOpenDocument により戻される文書ハンドルです。
DESMATCHINFO *	<i>pMatchInfo</i>	出力	受け取ったデータ・ストリーム部分を含むバッファに対するポインターです。DesGetMatches がそのバッファを割り当てます。
DESULONG *	<i>pMatchInfoLength</i>	出力	pMatchInfo で指されたデータ・ストリーム部分の長さです。
DESMESSAGE *	<i>pErrorMessage</i>	出力	処理系定義のメッセージ・テキスト。エラーが発生した場合は、DB2 テキスト・エクステンダーによりエラー・コードとエラー・メッセージが戻されます。アプリケーション・プログラムによってサイズ DES_MAX_MESSAGE_LENGTH のバッファが割り当てられます。pErrorMessage がヌル・ポインターである場合、エラー・メッセージは戻されません。

データ・ストリーム構文

▶ —05— —DB2TX_DOC— —DB2TX_START— —————▶

┌ —11— —DB2TX_DNAM— —DB2TX_ATOMIC— —*document_name*— ─┐

┌ Section ─┐ —05— —DB2TX_DOC— —DB2TX_END— —————▶

Section:

┌ —05— —DB2TX_DEL— —DB2TX_START— —————▶

┌ —11— —DB2TX_SNAM— —DB2TX_ATOMIC— —*section_name*— ─┐

┌ Text encoding ─┐ ┌ Paragraph ─┐ —05— —DB2TX_DEL— —DB2TX_END— —————▶

Text encoding:

┌ —07— —DB2TX_CCSID— —DB2TX_ATOMIC— —*coded_character_set_identifier*— —▶

▶ —07— —DB2TX_LANG— —DB2TX_ATOMIC— —*language_identifier*— —————┐

Paragraph:

┌ —05— —DB2TX_PAR— —DB2TX_START— —┐ Paragraph text ─┐ —————▶

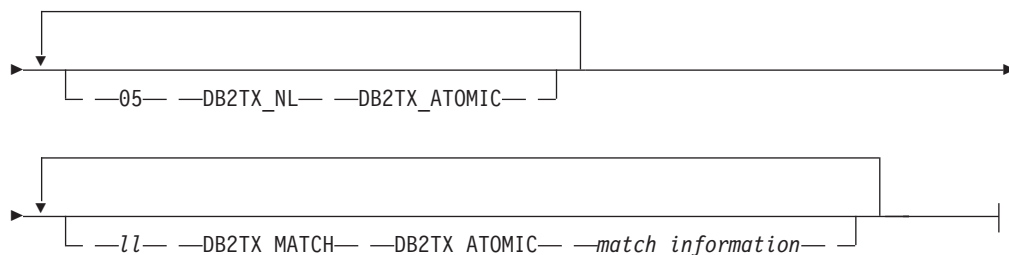
▶ —05— —DB2TX_PAR— —DB2TX_END— —————┐

Paragraph text:

┌ —11— —DB2TX_TEXT— —DB2TX_ATOMIC— —*text_unit*— ─┐

┌ —11— —DB2TX_LINK— —DB2TX_ATOMIC— —*media_ref*— ─┐

DesGetMatches API 関数



構文図の各セグメント、たとえば 05 DB2TX_DOC DB2TX_START は、整数タイプの長さフィールドで始まります。長さは、構文図中では、たとえば 05 のような明示的な数値、または変数 *ll* です。セグメントの長さには 2 バイトの長さフィールドが含まれます。

注: 長さはビッグ・エンディアン形式で表されます。

個々のセグメントには、以下の 1 バイト・タイプの ID が含まれます。

DB2TX_START

文書や段落などのセグメントの始まりを表します。

DB2TX_END

セグメントの終わりを表します。

DB2TX_ATOMIC

次に続く項目が、文書名や言語 ID のようにそれ以上分解できないものであることを表します。

データ・ストリーム項目は、それぞれが 2 バイトの長さを持ちます。以下はその例です。

DB2TX_DOC

文書の始まりと終わりを表します。

DB2TX_DNAM

文書名です。名前が指定されていない場合は、文書の ID が使用されます。

DB2TX_DEL

文書エレメントの始まりと終わりを表します。現在サポートされている唯一の文書エレメントのタイプは、*text section* です。

DB2TX_SNAM

テキスト・セクションの名前を指定します。現在 DB2 テキスト・エクステンダーではただ 1 つのテキスト・セクションしかサポートされておらず、自動的にデフォルトの名前が使用されます。セクション名を指定しても、これは無視されます。

DB2TX_PAR

現行セクション内でテキスト段落の始まりと終わりを表します。

DB2TX_TEXT

現行段落内で 1 つのテキスト部分を指定します。通常、*text_unit* にはテキストの 1 行が含まれ、TEXT 項目の後に DB2TX_NL 項目が続きます。しかし、

テキスト行をいくつかの部分に分割して、各部分をそれぞれ別個の DB2TX_TEXT 項目で指定することもできます。

テキストは、現行段落に関連した CCSID および言語を使用します。

DB2TX_LINK

DB2 テキスト・エクステンダー・ハイパーメディア参照を指定します。これは現行段落の CCSID を使用します。

DB2TX_NL

現行段落内の新しい行の始まりを表します。

DB2TX_MATCH

現行テキスト部分内の突き合わせに関するオカレンス情報を含みます。この情報は、一対になった 2 進数として示されます。個々の対の最初の数値は現行テキスト部分内での一致語のオフセットであり、2 番目の数値は、文字数単位のもの一致語の長さです。指定された長さが指定のテキスト部分を超えてしまう場合もあります。オフセットと長さは両方とも、ビッグ・エンディアン形式で指定された 2 バイトの値です。

DB2TX_CCSID

後続の段落にあるテキストの CCSID です。これは、段落の前に新しい DB2TX_CCSID 項目が来るまで有効です。以下の CCSID が戻されます。

DB2TX_CCSID_00500

(ラテン-1 EBCDIC コード・ページ 500 で記述されたテキストの場合)

DB2TX_CCSID_04946

(ラテン-1 ASCII コード・ページ 850 で記述されたテキストの場合)

DB2TX_CCSID_00819

(ASCII コード・ページ 819 で記述されたテキストの場合)

CCSID のこれらの記号名は、DB2 テキスト・エクステンダーが提供する DES_EXT.H で定義されます。2 バイト長の 2 進値は、ビッグ・エンディアン形式で指定されます。

DB2TX_LANG

後続の段落にあるテキストの言語 ID。これは、段落の前に新しい DB2TX_LANG 項目が来るまで有効です。DB2 テキスト・エクステンダーでサポートされるすべての言語 ID の記号名は、DB2 テキスト・エクステンダーに付属のファイル DES_EXT.H で定義されています。2 バイト長の 2 進値は、ビッグ・エンディアン形式で指定されます。

使用

テキスト文書の終わりに達すると、DesGetMatches は RC_SE_END_OF_INFORMATION を戻します。

戻りコード

RC_SUCCESS
 RC_SE_END_OF_INFORMATION

 RC_INVALID_PARAMETER
 RC_INVALID_SESSION

DesGetMatches API 関数

RC_SE_CAPACITY_LIMIT_EXCEEDED
RC_SE_INCORRECT_HANDLE
RC_SE_IO_PROBLEM
RC_SE_NOT_ENOUGH_MEMORY
RC_SE_REQUEST_IN_PROGRESS
RC_SE_LS_FUNCTION_FAILED
RC_SE_UNEXPECTED_ERROR

重要: 以下の戻りコードは、関数は結果を戻したが、予想とは異なることを示しています。

RC_SE_DICTIONARY_NOT_FOUND

制限

この関数は、DesOpenDocument を呼び出してテキスト文書をオープンした後でのみ、呼び出すことができます。

DesGetSearchResultTable

目的

テキスト列で識別されたテキスト文書を検索するための検索引き数を使用します。検出されたテキスト項目のハンドル・データは、結果表に書き込まれます。また、ランクや一致数に関するブラウズ情報も結果表に書き込むことができます。

構文

```

DESRETURN
DesGetSearchResultTable
(SQLHDBC          hdbc,
char              *pTableSchema,
DESSMALLINT      TableSchemaLength,
char              *pTableName,
DESSMALLINT      TableNameLength,
char              *pColumnName,
DESSMALLINT      ColumnNameLength,
char              *pSearchArgument,
DESSMALLINT      ArgumentLength,
char              *pResultSchema,
DESSMALLINT      ResultSchemaLength,
char              *pResultTableName,
DESSMALLINT      ResultTableNameLength,
DESSEARCHOPTION  SearchOption,
DESBROWSEOPTION  BrowseOption,
DESBROWSEINFO    *pBrowseInfo,
DESMESSAGE       *pErrorMessage);

```

関数引き数

表 14. DesGetSearchResultTable 引き数

データ・タイプ	引き数	用法	説明
SQLHDBC	hdbc	入力	データベース接続ハンドルです。
char *	pTableSchema	入力	検索対象の基礎表のスキーマです。
DESSMALLINT	TableSchemaLength	入力	pTableSchema の長さ (ヌル・バイトの終止符を含まない)、または DES_NTS です。
char *	pTableName	入力	検索される基礎表の名前のポインターです。
DESSMALLINT	TableNameLength	入力	pTableName の長さ (ヌル・バイトの終止符を含まない)、または DES_NTS です。
char *	pColumnName	入力	実行しようとしているテキスト検索によってアドレッシングされる列名に対するポインターです。列のタイプは DESTEXTH でなければなりません。
DESSMALLINT	ColumnNameLength	入力	pColumnName の長さ (ヌル・バイトの終止符を含まない)、または DES_NTS です。
char *	pSearchArgument	入力	テキスト検索引き数のポインターです。
DESSMALLINT	ArgumentLength	入力	pSearchArgument の長さ (ヌル・バイトの終止符を含まない)、または DES_NTS です。
char *	pResultSchema	入力	結果表を含むスキーマのポインターです。
DESSMALLINT	ResultSchemaLength	入力	pSchemaName の長さ (ヌル・バイト終止符を含まない)、または DES_NTS です。

DesGetSearchResultTable API 関数

表 14. DesGetSearchResultTable 引き数 (続き)

データ・タイプ	引き数	用法	説明
char *	pResultTableName	入力	直前に作成した結果表の名前のポインターです。この表には、検索の結果が保管されます。この表の構造体については、図 16 を参照してください。
DESSMALLINT	ResultTableNameLength	入力	pResultTableName の長さ (ヌル・バイトの終止符を含まない)、または DES_NTS です。
DESSEARCHOPTION	SearchOption	入力	ユーザーの要求が、ランキング情報、一致数、または一致するテキスト文書のハンドルのいずれであるかを指定するオプションです。 DES_RANK DES_MATCH DES_RANKANDMATCH DES_TEXTHANDLEONLY このオプションは、『使用』に説明されている結果表の内容を決めます。
DESBROWSEOPTION	BrowseOption	入力	予約済み。
DESBROWSEINFO *	pBrowseInfo	出力	BrowseOption の値に基づき、ブラウズ情報に対するポインターまたはヌルに対するポインターとなります。
DESMESSAGE *	pErrorMessage	出力	処理系定義のメッセージ・テキスト。エラーが発生した場合は、DB2 テキスト・エクステンダーによりエラー・コードとエラー・メッセージが戻されます。アプリケーション・プログラムによってサイズ DES_MAX_MESSAGE_LENGTH のバッファが割り当てられます。pErrorMessage がヌル・ポインターである場合、エラー・メッセージは戻されません。

使用

データベースへの接続は、DesGetSearchResultTable を呼び出すアプリケーション・プログラムで確立しなければなりません。

名前 pResultTableName は、事前に作成しておいた結果表です。サンプル・ディレクトリ内のユーティリティー DESRESTB は、テキスト・ハンドル用の結果表を作成します。この関数を呼び出すと、結果表には検索引き数に一致するテキスト値を識別する情報が入れます。次に結果表の構造体を示します。

RESULT TABLE

TEXTHANDLE	RANK	MATCHES

図 16. 結果表の構造体

TEXTHANDLE のデータ・タイプは DB2TEXTH または DB2TEXTFH です。
RANK のデータ・タイプは DOUBLE、MATCHES のデータ・タイプは INTEGER
です。

pSearchArgument の検索引き数については、165 ページの『第 12 章 検索引き数の
構文』で説明されています。

BrowseOption の値が *BROWSE* である場合、DB2 テキスト・エクステンダーは、サ
ーバー上の DB2 テキスト・エクステンダー検索プログラムからのブラウザ情報を
戻します。*pBrowseInfo* は、DesStartBrowseSession への入力であるブラウザ情報を指
します。*BrowseOption* の値が *NO_BROWSE* である場合、*pBrowseInfo* はヌルを指し
ます。

戻りコード

```
RC_SUCCESS
RC_NO_BROWSE_INFO
RC_SE_NO_DATA

RC_ALLOCATION_ERROR
RC_FILE_IO_PROBLEM
RC_INTERNAL_ERROR
RC_INVALID_BROWSE_OPTION
RC_INVALID_PARAMETER
RC_INVALID_SEARCH_OPTION
RC_INVALID_SESSION
RC_PARSER_INVALID_ESCAPE_CHARACTER
RC_PARSER_SYNTAX_ERROR
RC_RESULT_TABLE_NOT_EXIST
RC_SE_COMMUNICATION_PROBLEM
RC_SE_EMPTY_INDEX
RC_SE_EMPTY_QUERY
RC_SE_FUNCTION_DISABLED
RC_SE_FUNCTION_IN_ERROR
RC_SE_INCORRECT_HANDLE
RC_SE_INDEX_DELETED
RC_SE_INDEX_NOT_ACCESSIBLE
RC_SE_INDEX_SUSPENDED
RC_SE_INSTALLATION_PROBLEM
RC_SE_IO_PROBLEM
RC_SE_MAX_NUMBER_OF_BUSY_INDEXES
RC_SE_NOT_ENOUGH_MEMORY
RC_SE_PROCESSING_LIMIT_EXCEEDED
RC_SE_QUERY_TOO_COMPLEX
RC_SE_SERVER_BUSY
RC_SE_SERVER_CONNECTION_LOST
RC_SE_SERVER_NOT_AVAILABLE
RC_SE_UNEXPECTED_ERROR
RC_SE_UNKNOWN_INDEX_NAME
```

DesGetSearchResultTable API 関数

RC_SE_UNKNOWN_SERVER_NAME
RC_SE_WRITE_TO_DISK_ERROR
RC_SQL_ERROR_NO_INFO
RC_SQL_ERROR_WITH_INFO
RC_TEXT_COLUMN_NOT_ENABLED

重要: 以下の戻りコードは、関数は結果を戻したが、予想とは異なることを示しています。

RC_SE_CONFLICT_WITH_INDEX_TYPE
RC_SE_DICTIONARY_NOT_FOUND
RC_SE_STOPWORD_IGNORED
RC_SE_UNKNOWN_SECTION_NAME
RC_SE_DOCMOD_READ_PROBLEM

DesOpenDocument

目的

ブラウザ・セッション・ポインター、ハンドル、およびテキスト文書の解析に辞書を使用する必要があるかどうかを示すオプション (DES_EXTENDED または DES_FAST)。この関数は、文書テキストおよび強調表示情報を入手するためのハンドルに対応するテキスト文書を作成し、DesGetMatches の反復呼び出しに使用される文書ハンドルを戻します。

構文

```

DESRETURN
DesOpenDocument
(DES_BROWSE_SESSION BrowseSession,
SQLCHAR *pHandle,
DES_USHORT HandleLength,
DES_MATCH_MODE MatchMode,
DES_HANDLE *pDocumentHandle,
DES_MESSAGE *pErrorMessage);

```

関数引き数

表 15. DesOpenDocument 引き数

データ・タイプ	引き数	用法	説明
DES_BROWSE_SESSION	BrowseSession	入力	ブラウザ・セッション・ハンドルです。
SQLCHAR *	pHandle	入力	データベースから抽出されるハンドルへのポインターです。
DES_USHORT	HandleLength	入力	pHandle の長さです (DES_NTS は使用できません)。
DES_MATCH_MODE	MatchMode	入力	強調表示情報の検出に辞書を使用するかどうかを決定するモードです。 DES_FAST 辞書を使用しない。 DES_EXTENDED 辞書を使用する。
DES_HANDLE *	pDocumentHandle	出力	DesGetMatches の反復呼び出しのための文書ハンドルです。
DES_MESSAGE *	pErrorMessage	出力	処理系定義のメッセージ・テキスト。エラーが発生した場合は、DB2 テキスト・エクステンダーによりエラー・コードとエラー・メッセージが戻されます。アプリケーション・プログラムによってサイズ DES_MAX_MESSAGE_LENGTH のバッファが割り当てられます。pErrorMessage がヌル・ポインターである場合、エラー・メッセージは戻されません。

DesOpenDocument API 関数

使用

DES_FAST および DES_EXTENDED は、ブラウザ・テキストで強調表示する用語を検出するために言語処理を使用することを示します。詳細については、204 ページの『ブラウザのための言語的な処理』を参照してください。基本テキスト解析を使用するには DES_FAST を使用し、拡張マッチングを使用するには DES_EXTENDED を使用してください。

SQL データ・タイプと C データ・タイプ間のマッピングでは、ハンドルとして SQL 記号名 SQL_VARBINARY を使用する必要があります。TextHandle 値の C 表示を指すホスト変数のタイプは、SQLCHAR* です。

DB2 テキスト・エクステンダーにより、ブラウザ情報用にストレージが割り当てられます。アプリケーション・プログラムで、DesFreeBrowseInfo を呼び出してこのストレージと関連するリソースを解放しなければなりません。

TextHandle 値はビット・データで、'¥0' 文字をいくつか含んでいるため、pHandle の長さを指定する必要があります。

呼び出し元は、pHandle によって参照されたテキスト文書を含む表への読み取りアクセス権を持っている必要があります。

戻りコード

RC_SUCCESS

RC_ALLOCATION_ERROR

RC_INTERNAL_ERROR

RC_INVALID_MATCH_OPTION

RC_INVALID_PARAMETER

RC_INVALID_SESSION

RC_SE_DOCUMENT_NOT_ACCESSIBLE

RC_SE_DOCUMENT_NOT_FOUND

RC_SE_INCORRECT_HANDLE

RC_SE_IO_PROBLEM

RC_SE_LS_FUNCTION_FAILED

RC_SE_LS_NOT_EXECUTABLE

RC_SE_MAX_NUMBER_OF_BUSY_INDEXES

RC_SE_NOT_ENOUGH_MEMORY

RC_SE_REQUEST_IN_PROGRESS

RC_SE_UNKNOWN_INDEX_NAME

RC_SE_UNEXPECTED_ERROR

制限

この関数は、DesStartBrowseSession 呼び出しによってブラウザ・セッションを開始した後でのみ、呼び出すことができます。

DesStartBrowseSession

目的

ブラウザ・セッションを開始し、テキスト文書をブラウザしてその一致語を強調表示するために必要な環境を確立します。この関数は、DesGetBrowseInfo または DesGetSearchResultTable のいずれかよりブラウザ情報に対するポインタを受け取り、他のブラウザ関数が使用するためのブラウザ・セッション・ハンドルを戻します。

構文

```

DESRETURN
DesStartBrowseSession
(DESBROWSEINFO
char
DESSMALLINT
char
DESSMALLINT
DESBROWSESESSION
DESMESSAGE;
BrowseInfo,
*pUserId,
  UserIdLength,
*pPassword,
  PasswordLength,
*pBrowseSession,
*pErrorMessage);

```

関数引き数

表 16. DesStartBrowseSession 引き数

データ・タイプ	引き数	用法	説明
DESBROWSEINFO	<i>BrowseInfo</i>	入力	テキスト文書内の一致語をブラウザして強調表示するために必要な情報に対するポインタです。ポインタは DesGetSearchResultTable または DesGetBrowseInfo によって戻されます。
char *	<i>pUserId</i>	入力	データベースのユーザー ID です。
DESSMALLINT	<i>UserIdLength</i>	入力	データベースのユーザー ID の長さです。
char *	<i>pPassword</i>	入力	データベースのパスワードです。
DESSMALLINT	<i>PasswordLength</i>	入力	データベースのパスワードの長さです。
DESBROWSESESSION *	<i>pBrowseSession</i>	出力	他のブラウザ関数が使用するブラウザ・セッションのハンドルです。
DESMESSAGE *	<i>pErrorMessage</i>	出力	処理系定義のメッセージ・テキスト。エラーが発生した場合は、DB2 テキスト・エクステンダーによりエラー・コードとエラー・メッセージが戻されます。アプリケーション・プログラムによってサイズ DES_MAX_MESSAGE_LENGTH のバッファが割り当てられます。 <i>pErrorMessage</i> がヌル・ポインタである場合、エラー・メッセージは戻されません。

DesStartBrowseSession API 関数

使用

この関数により、テキスト文書をブラウズするためのブラウズ・セッションがオープンされます。ユーザーがデータベースへのアクセス権限を持っているかをチェックするために、ユーザー ID とパスワードの入力が要求されます。

ブラウズ・セッションのクローズは、DesEndBrowseSession を呼び出して行います。

BrowseInfo は、ブラウズ情報の作成に使用される検索引き数と基本テキスト列によって異なります。

戻りコード

RC_SUCCESS

RC_ALLOCATION_ERROR

RC_INVALID_BROWSE_INFO

RC_INVALID_PARAMETER

RC_INTERNAL_ERROR

RC_SE_NOT_ENOUGH_MEMORY

RC_SE_UNEXPECTED_ERROR

RC_SQL_ERROR_NO_INFO

RC_SQL_ERROR_WITH_INFO

制限

この関数を呼び出す前に、DesGetBrowseInfo または DesGetSearchResultTable を、適切なブラウズ・オプションを指定して呼び出しておかなければなりません。

第 14 章 サンプル API プログラム

DB2 テキスト・エクステンダーでは、サンプル・プログラム DESSAMP (タイプ C) を提供しています。このプログラムは、DB2 テキスト・エクステンダー製品ライブラリー QDB2TX の SAMPLES ファイルにあります。

DESSAMP は、DesGetSearchResultTable 関数とユーザー独自のブラウザー・プログラムを使用するプログラムの例です。これは、92 ページの図 15 に示されている API 関数呼び出しの順序に従います。

使用可能データベースおよび使用可能テキスト列にアクセスする必要があります。このサンプル・プログラムを実行するには、次の手順を実行します。

1. 推奨コンパイラーを使用してサンプル・コードをコンパイルします。
2. DB2 テキスト・エクステンダー製品ライブラリー QDB2TX から TXSAMPLE 制御言語プログラムを実行して、必要な表および索引をセットアップします。
3. 以下のサンプル・コードと一緒に使用する結果表を作成します。

```
CREATE TABLE DB2TX.RES_TBL (TEXTHANDLE DB2TX.DB2TEXTH,  
                             RANK DOUBLE, MATCHES INTEGER)
```

この表は、検索結果などの情報を保管するために使用されるもので、その構造は次のとおりです。

列	データ・タイプ
HANDLE	DB2TX.DB2TEXTH または DB2TX.DB2TEXTFH
RANK	DOUBLE
MATCHES	INTEGER

サンプル API プログラム

第 15 章 基本語形索引と一致語句索引の言語的な処理

DB2 テキスト・エクステンダーでは、以下のような検索の分野における言語的な処理を提供しています。

- **索引付け。** DB2 テキスト・エクステンダーが文書を分析してテキスト索引に保管される用語を抽出する場合、テキストが言語的に処理されてその索引に最適な用語を抽出します。これは、検索をできるだけ簡単かつ早くするために行われます。
- **検索。** DB2 テキスト・エクステンダーが、指定された検索項目のオカレンスを含む文書を検出するために文書索引内を検索する際に、検索項目も言語的に処理されて索引用語と突き合わされます。
- **ブラウズ。** 検索後に検出された文書をブラウズすると、言語的な処理が行われて文書内で検出された用語が強調表示されます。

索引作成時の言語的な処理

DB2 テキスト・エクステンダーは、文書の索引付けと検索を行う際に、テキストの言語的な分析を行います。以下の表から分かるとおり、言語的な処理の量は、索引のタイプによって異なります。Ngram 索引では、言語的な処理は行われません。

文書の索引付けに使用される言語処理は以下のとおりです。

- 基本テキスト分析
 - 用語の認識 (トークン化)
 - 用語の標準形への正規化
 - 文の認識
- 用語の基本形への変形
- 停止ワードのフィルター操作
- 分解 (複合語を分割)

表 17 は、索引タイプが**基本語形**で、追加の索引特性が要求されていない場合に、用語を索引付けする方法をまとめて示しています。

表 17. 基本語形索引の用語抽出

文書テキスト	索引中の用語	言語的な処理
Mouse Käfer	mouse kaefer	基本テキスト分析 (正規化)
mice swum	mouse swim	基本形への変形
a report on animals	report animal	停止ワードのフィルター操作。 停止ワードは a と on。

比較のため、198 ページの表 18 には索引タイプが**一致語句**の場合に用語を索引付けする方法をまとめています。

索引作成時の言語的な処理

表 18. 一致語句索引の用語抽出

文書テキスト	索引中の用語	言語的な処理
Mouse Käfer	Mouse Käfer	正規化なし
mice swum	mice swum	基本形への変形なし
a report on animals	report animals	停止ワードのフィルター操作。 停止ワードは a と on。
system-based Wetterbericht	system-based Wetterbericht	分解なし

基本テキスト分析

DB2 テキスト・エクステンダーは、電子辞書を使用せずに基本テキスト分析を処理します。

非英数字文字を含む用語の認識

文書を索引付けする場合には、非英数字文字を含んだ用語でも認識されます。たとえば、「\$14,225.23」、「mother-in-law」、および「10/22/90」などです。

以下のものは、用語の一部と見なされます。

アクセント

通貨記号

数値区切り記号（「/」や「.」など）

E メール・アドレスに含まれる「@」（英語の場合のみ）

「+」記号

以下のものを含む用語の認識には、言語に固有の規則も使用されます。

- ラテン系言語に含まれるアクセントつき接頭部（フランス語の l'aventure など）。
- 日付、時刻、数字の国別の形式。
- 英語では「/」を使用して示される（mission/responsibility など）代替語の表記。
- securita' などのイタリア語の後続アポストロフィ。これは通常、タイプ印刷のイタリア文に出てきます。アクセント記号付きの文字が文字セットになれば、イタリア語テキストを文字入力する場合は通常はその文字の後にアクセント記号を入力します。たとえば、「à」は「a」と入力されます。

用語の標準形への正規化

正規化は、大小文字が混合している用語や、アクセント記号付き文字や特殊文字を含む用語を、標準形に変形します。これは、索引タイプが基本語形である場合に、デフォルト処理として行われます。（一致語句索引の場合、大文字小文字は変わりません。つまり、検索では大文字小文字を区別します。）

たとえば、Computer という用語は、computer と索引付けされ、大文字は小文字に変形されます。computer という用語を検索すると、computer のオカレンスのほ

かに、Computer のオカレンスも検出されます。索引付け中に正規化を用いると、文書内の用語に大文字が含まれているかどうかに関係なく、同じように索引付けされるという効果があります。

正規化は、索引付け中のほかに、検索中にも適用されます。検索項目の中の大文字は、検索が行われる前に小文字に変形されます。たとえば、検索項目が Computer である場合、検索に使用される用語は computer です。

アクセント記号付き文字と特殊文字は、同じ方法で正規化されます。École といった、école のどのバリエーションを使用しても、école、Ecole などが検出されず。Bürger で buerger が、Maße で masse が検出されます。

検索項目にマスキング (ワイルドカード) 文字が含まれている場合、正規化が行われてからそのマスキング文字が処理されます。例: Bür_er は buer_er となります。

文の認識

同じ文中にある用語を検索することができます。このことを可能にするため、個々の文書は索引付け中に分析されて、それぞれの文の終了位置が検出されます。

DB2 テキスト・エクステンダーでは、2 種類の方法で文の最終位置が認識されます。

- Universal Unicode Tokenizer (アラビア語とヘブライ語以外の言語の場合)
この方式の方が簡単で高速です。トークン化機能は、ピリオド、感嘆符、疑問符などのうち、前にトークン文字 (文字など) があり、後ろにブランク、タブ、改行文字があるものを探します。これが本当に文の終了位置であり、単なる省略語の末尾のピリオドではないことを検査するために、言語固有の省略語のリストが検査されます。
- POE ベースのトークン化機能 (アラビア語とヘブライ語の場合)
このトークン化機能の方が、言語の面ではより拡張されていますが、高い処理能力が必要とされます。このトークン化機能では、基本的には句読点をマッチングして文の終了位置が検索されますが、特殊な入力タイプや語数も手掛かりとして用います。

トークン化機能による文の認識

POE ベースのトークン化機能は、句読点の規則と言語固有の処理 (省略語の処理を含む) を使用して、文 (または文の一部) の境界を判別します。省略語処理機能のレベルは言語によって大きく異なります。単一バイト・コード・ページを使用する言語のほとんどについては、関連した省略語補足辞書が POE に付属しています。通常、2 バイト言語ではピリオドを含む省略語を使用しないため、省略語補足辞書は使用できません。

基本的には、句読点をマッチングして文の終了位置が判別されます。以下の表には、終端の句読文字とその 図形文字グローバル ID (GCGID) がリストされています。

SBCS 文字の GCGID	DBCS 文字の GCGID	説明
SP110000	SP110080	ピリオド
SP020000	SP020080	感嘆符
SP150000	SP150080	疑問符

索引作成時の言語的な処理

SBCS 文字の GCGID	DBCS 文字の GCGID	説明
SP140000	N/A	セミコロン (ギリシャ語の疑問符)
N/A	JQ730080	2 バイトの丸の終止符

以下のいずれにも該当しない場合、終端の句読文字 (ピリオド、感嘆符、疑問符など) は文の終了位置を示していると見なされます。

- 以下の表にリストされている終結句読文字 (右小括弧など) が、終端の句読文字の後にある。

SBCS 文字の GCGID	DBCS 文字の GCGID	説明
SP070000	SP070080	右小括弧
SP040000	SP040080	二重引用符
SP050000	SP050080	単一引用符
SP180000	SP070083	角付き二重引用符
N/A	SM140080	右中括弧
N/A	SM080080	右大括弧
N/A	JQ720080	単一角引用符
N/A	JQ720081	二重角引用符
N/A	SP200080	右一重引用符
N/A	SP220080	右二重引用符
N/A	SP070081	右甲括弧
N/A	SP070082	右単一角付き引用符
N/A	SP070084	右角小括弧
N/A	SP370080	縦の右単一角引用符
N/A	SP370081	縦の右二重角引用符
N/A	SP250084	縦の右角小括弧
N/A	SP250080	縦の右小括弧
N/A	SP350080	縦の右中括弧
N/A	SP250081	縦の右甲括弧
N/A	SP250083	縦の右角度付き二重引用符
N/A	SP250082	縦の右単一角付き引用符

注: 「N/A」と示された項目は POE では終結句読文字と見なされません。縦の終結句読文字は中国語でのみサポートされます。

例:

...this sentence ends with two parentheses.))

上記の例で、2 つ目の小括弧が文の終了位置として検出されます。しかし、ドイツ語では、終結引用符の後にコンマがある場合、その引用符は文の終了位置とは見なされません。

- 終端の句読文字の後に別の終端文字がある。例:

This is a strong exclamation!!!

最後の感嘆符が文の終了位置として検出されます。

- 終端の句読文字の前に数値か句読文字があり、後に数字がある。「1.25」や「.314」などのストリングは文の終了位置とは見なされません。
- 終端の句読文字がピリオドで、文の終了位置として使用できない省略語の一部である。言語別に限定された省略語処理が実行されます。
- 終端の句読文字がピリオドで、後に空白文字 (ブランクや改行文字など) がない。「III.IV」などのヘッダーは文の終了位置として検出されません。

また POE ベースのトークン化機能は、省略語処理を行って、ピリオドが省略語の一部なのか、あるいは文の終了位置を示すのかを判別します。省略語補足辞書に省略語を追加できます。POE ベースのトークン化機能に辞書が渡されない場合は、単一文字の後にピリオドがあると、すべて省略語と見なされます。その他の省略語処理は行われません。

省略語を、通常の語の後にピリオドが付いたものと間違える可能性があるため、特定のテキストが省略語かどうか判別しにくい場合がよくあります。たとえば、以下の文の文字「no.」がこれに該当します。

Enter the no. of exemptions you are claiming.
Answer each question with yes or no.

特定のテキストが省略語であると分かっているにもかかわらず、依然として文の終了位置かどうかの判別は困難です。決して文の終了位置には使用されない省略語もあれば、使用される場合もある省略語もあります。たとえば、以下の文中の省略語「Hwy.」がこれに該当します。

The drive along Hwy. 1 to Santa Cruz was beautiful.
Many people speak highly of the Pacific Coast Hwy.

省略語は判別しにくいことがあり、文の終了位置でないこともあるので、POE では省略語を検出した後に分類します。文の終了位置に使用されることのある省略語の一部にピリオドがあると、それ以上の処理は実行されません。POE により、省略語が文の終了位置ではないと判別されると、ピリオドを表すトークンが省略語のテキストのトークンと結合されます。それ以外の場合、ピリオドを表すトークンは独立した状態のままです。

POE ベースの省略語処理では、以下の 3 種類の基準を使用して、ピリオドが省略語の一部かどうか判別されます。

- 単一文字の後にピリオドがある場合は、すべて省略語と見なされる。単一文字の省略語は、文の終了位置の可能性のあるものとして分類されます。
- 入力省略語補足辞書に含まれる語は、常に省略語と見なされる。補足中にある省略語が文の終了位置であるかどうかは、補足中のその語に関連した情報によって判別されます。たとえば、米国英語の省略語補足辞書で、「Mr.」は文の終了位置になることのできない省略語として示され、「etc.」は文の終了位置になることのある省略語として示されています。
- 長さ 2 ~ 6 文字の語の後にピリオドがある場合は、入力辞書や省略語補足辞書になくても、省略語と見なされる。「Jrnl.Comp. Ling」などの事例を処理する場合はこれに該当します。辞書検索によって判別される省略語は、常に文の終了位置の可能性のあるものとして扱われます。

索引作成時の言語的な処理

省略語が文の終了位置の可能性のあるものとして識別されると、その後のテキストが POE により検査され、その省略語が現行文の終了位置であるかどうかが決まります。その方法は、次の語の先頭が大文字かどうかを検査するというものです。

省略語の後に複数の改行、新しい文、または新しい段落のデータ・エレメントがある場合は、POE では文の終了位置に達したと見なします。また、後のテキストが逆疑問符か逆感嘆符の場合は、文の終了位置を示すマーカーが出力中に挿入されます。

POE によりピリオドが、文の終了位置にない省略語の一部であると判別された場合は、引き続き文の区切り文字が検索されます。それ以外の場合は、他の終端句読文字の例外条件（後に終端句読点や終結句読点があるか）が検査されてから、文の終了位置としてマーキングされます。

用語の基本形への変形 (lemmatization)

たとえば、基本語形索引で、`mouse` を検索し、`mice` を検出することができます。索引付けの際に用語は基本形に変形されるので、`mice` という用語は `mouse` と索引付けされます。その後、`mouse` という検索項目を使用するとその文書が検出されます。`mice` で検索を行った場合でも、その文書は検出されます。

`mouse` という用語のどのようなバリエーションが文書中に現れても、あるいはそのようなバリエーションを検索項目として使用しても、結果として、`mice` に関する情報を含む文書を検出する効果が得られます。

同様に、動詞の活用形は原形に変形されます。たとえば、`bought` は `buy` となります。

停止ワードのフィルター操作

停止ワードとは、文書内に頻繁に出てくる前置詞や代名詞などを指します。したがって、検索項目としては適切ではありません。このような語は、それぞれの辞書に関連した停止ワード・リストの中にあり、索引付けのプロセスからは除外されます。

停止ワード処理は、大文字小文字の区別をしません。したがって、`about` という停止ワードによって、文の最初の語である `About` も除外されます。この停止ワード・リストは、各国語で提供されており、修正することができます。

Ngram 索引には停止ワード・リストはありません。

検索のための言語的な処理

照会処理では、検索の再呼び出し率が向上するように、すなわち、より多くの関連文書が見つかるように、検索項目に融通性をもたせることを目指しています。この目標を達成するために、照会用語に対して、拡張と変形の 2 つの基本操作が適用されます。さらに、用語検索操作には、拡張と変形の両方の操作を含むものがあります。

- 拡張操作では、検索項目から 1 つの語あるいは複数語からなる用語を取り出して、それを代替検索用語セットと関連付けます。これは複数語からなる用語自体の場合もあります。ソース表記とターゲット表記セットは、DB2 テキスト・エ

クステンダーの照会言語でプール OR 表記を形成します。ソース用語は、拡張されても未変更のまま残されるため、ある程度索引タイプから独立しています。拡張操作は次のとおりです。

同義語の拡張

シソーラスの拡張

- 変形によって、検索項目がユーザーが指定したものより普遍的な形式に変更されます。これは検索項目を変更するため、変形は索引タイプに依存して、変更された用語が必ず一致するようにします。したがって、DB2 テキスト・エクステンダーは、照会の対象となる索引から変形情報を取り出します。変形は次のとおりです。

Lemmatization (202 ページの『用語の基本形への変形 (lemmatization)』を参照)

正規化 (198 ページの『用語の標準形への正規化』を参照)

停止ワード (202 ページの『停止ワードのフィルター操作』を参照)

- 操作によっては、検索項目の変更および代替用語のセットを使用する拡張の両方を行うものがあります。このような操作は固有の変形のため、やはり索引に含まれている情報に依存します。以下の操作はこのクラスに属しています。

文字と語のマスキング

発音の拡張

同義語

同義語は、セマンティック的に関係のある語です。通常、このような語はソース用語と同じ語クラス (名詞、動詞など) をもちます。同義語は、言語ごとの独立したファイルから得られます。これらは常に基本形で戻され、若干の例外を除き、複数語の用語ではありません。同義語を探す際には、検索項目語は常に、それらの基本形に変形されます。以下に 3 種の言語による同義語の例を示します。

- 英語

word:

comment remark statement utterance term expression
communication message assurance guarantee warrant bidding command
charge commandment dictate direction directive injunction instruction
mandate order news advice intelligence tidings gossip buzz cry
hearsay murmur report rumor scuttlebutt tattle tittle-tattle
whispering

- フランス語

mot:

expression parole terme vocable lettre billet missive épître
plaisanterie

- ドイツ語

Wort:

Vokabel Bezeichnung Benennung Ausdruck Begriff Terminus
Ehrenwort Brocken Bekräftigung Versprechen Zusicherung Gelöbnis
Beteuerung Manneswort Schwur Eid Ausspruch

シソーラスの拡張

検索項目は、特定の関係からアクセスできるシソーラスの用語を使用して拡張することができます。このような関係は、階層関係 (「用語の限定」関係など)、関連関

検索のための言語的な処理

係（「関連用語」関係など）、あるいは同義語の関係が考えられます。シソーラス用語は、複数語用語である場合がよくあります。

206 ページの『シソーラスの概念』で、シソーラス拡張について詳細に説明しています。

シソーラス検索を行う際には、検索項目（検索開始時の用語）は正規化されません。シソーラス検索で検出された語は、索引のタイプに従って基本形に変形されます。

発音の拡張

発音の拡張では、1 つの語を発音の似た語のセットにより拡張します。これは特に、検索する用語の正確なつづりがわからない場合に役に立ちます。

文字と語のマスキング

マスキングは、非言語的拡張手法で、これにより、通常の表記が、それを満たすすべての索引付き語の分離で置き換えられます。マスクされた表記もそのいずれの拡張も lemmatization、停止ワードの抽出、またはそれ以外のどの拡張手法にも従いません。その結果、たとえば、マスクされた用語 `swu*` を使用して検索する際に、`swum` のような不規則な動詞形式が一致語句索引で一致しますが、この形式が `swim` とい基本形に変形されている基本語句索引では一致しません。

語のマスキングを使用する場合、特に大きいサイズの索引を検索している際に、パフォーマンスが低下することがあります。

ブラウズのための言語的な処理

言語的な処理は、検索後に検出された文書をブラウズする際にも使用されます。これは以下の 2 つの段階で行われます。

1. 基本テキスト分析：正規化と用語拡張
2. 拡張マッチング

ステージ 1: 正規化と用語拡張

最初の段階は、電子辞書を使わずに行われます。

正規化

正規化は、198 ページの『基本テキスト分析』で説明されています。

用語拡張

用語拡張は、用語を基本形に変形することの逆の処理です。索引が基本語形索引の場合、検索項目は検索前に基本形に変形されます。

同様に、基本語形索引の場合、文書の利用語は索引に追加される前に基本形に変形されます。したがって、文書は用語の基本形に基づいて検出されます。

しかし、検出された文書をブラウズする場合には、基本形のあらゆる変化形を強調表示することができます。このような変化形を強調表示するために、検出された用語の拡張が行われます。

辞書で検出される各用語の変化形 (語形変化) はすべて作成可能です。ドイツ語 gehen (行く) に対して生成される変化語形は次のとおりです。

gegangen geh gehe gehen gehend gehest gehet gehst
ginge gingen gingest ginget gingst gingt geht

ステージ 2: 拡張マッチング

第 2 段階は拡張マッチングです。この段階は、基本テキストの分析および正規化により検出用語を強調表示できない場合に、まれに使用されます。拡張マッチングでは、一致度の低い用語を検出します。

DesOpenDocument API 関数にパラメーターとして DES_EXTENDED を指定することによって、拡張マッチングを選択します。

拡張マッチングでは、基本語形索引付けの際に行われる処理と同じ言語的な処理を使用します。

拡張マッチングにより追加の一致項目を検出できるのは、以下の場合です。

- 検索項目は、マスキング文字を含む変化形です。

マスキング文字が処理され、検索項目の語幹が変形されて、該当する文書が検出されます。拡張マッチングが行われないと、指定の検索基準に合ったテキストを強調表示することはできません。

例: 文書に変化形 swam が含まれている場合

- この用語は、索引付け中に swim に変形される。
- 検索項目が swi% の場合、語幹が swim に変形されるので上記の文書は検出される。
- 拡張マッチングを使用しないと、swi% という語と一致する語しか強調表示されない。拡張マッチングを使用すると、変化形である swam も強調表示される。

- 複合語が索引付けされている場合

ゲルマン系諸言語の文書で、中に複合語があり、かつ基本語形索引を使って索引付けされている場合、その文書索引には複合語とその各部分が保存されます。複合語の一部分を検索すると、その複合語を含んだ文書が検出されますが、拡張マッチングを使用しない限り、その語は強調表示されません。

例: 文書にドイツ語 Apfelbaum (りんごの木) が含まれている場合

- この語は、言語の索引付け中に apfel と baum に変形される。
- 索引で baum という用語を検索すると、Baum という用語とそれを含む文書が索引内で見つけれられる。
- 拡張マッチングを使用しないと、文書には Apfelbaum が含まれているが Baum は含まれていないため、どの用語も強調表示されない。拡張マッチングを使用すると、Apfelbaum という複合語は分割され、Baum 部分が見つけれられて強調表示される。

- 行末の語がハイフン付けされている場合

ワード・プロセッサでハイフンが自動的に挿入される場合は、ハイフン付けされた語を検出して強調表示することができます。しかし、ユーザーがハイフンを入力する場合は、その語を含んだ文書の検出は行われますが、拡張マッチングを使用しない限り、その語は強調表示されません。

ブラウザのための言語的な処理

例: 次のように、行の終わりで中断されたためにハイフン付けされている container という語が文書に含まれている場合

Another name for a folder is a container.

- この語は、索引付け時に container に正規化される。
- 索引を使用して container という語を検索すると、その語を含んだ用語と文書が検出される。
- 文書内で container と一致するあらゆる語を強調表示する試みが行われる。拡張マッチングを使用しないと、con-tainer 内のハイフンがユーザーによって入力されたのではなく、テキスト・プロセッサによって挿入された場合にのみ一致します。

シソーラスの概念

シソーラスとは、通常、特定の対象領域を網羅するセマンティックに関連のある用語の、制御された語いのことです。これは、各用語をノードによって表したセマンティック・ネットワークとして視覚化することができます。2つの用語が互いに関連している場合、それらのノードは、関係名ラベルの付いたリンクによって接続されます。特定の用語に直接関連しているすべての用語には、そのノードから出ている、すべての接続をたどることによって達することができます。間接的に関連している用語には、前のステップで到達したノードから出ている、すべての接続をさらにたどることによって達することができます。図17は、ごく小さいシソーラスの構造例を示しています。

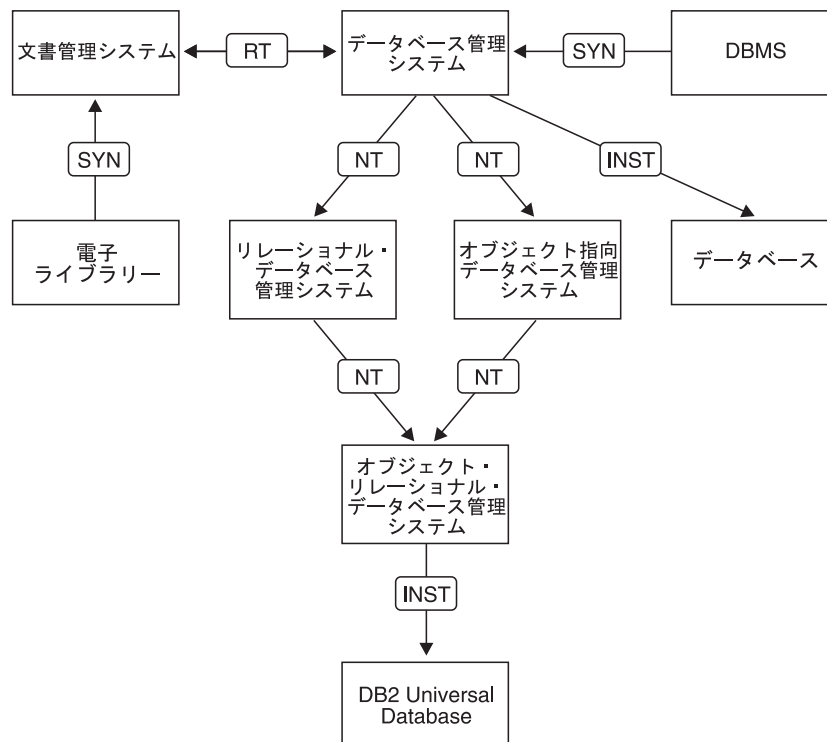


図17. ネットワークとして表されたシソーラス

DB2 テキスト・エクステンダーにより、事前に作成済みのシソーラスからさらに用語を追加することによって、検索項目を拡張することができます。照会でシソーラスの拡張を使用する方法については、165ページの『第12章 検索引き数の構文』を参照してください。

検索アプリケーションで使用するためにシソーラスを作成するには、シソーラス定義ファイルを内部形式 (シソーラス辞書) にコンパイルする必要があります。

基本語形索引および一致語句索引で使用される辞書形式は、Ngram 索引で使用されるものとは異なります。したがって、2つの異なるシソーラス・コンパイラーが製品に付属しています。それらは、基礎となる概念が若干異なるだけでなく、必要とするソース形式も異なっています。したがって、検索アプリケーション用のシソーラスを定義し始める前に、まずどの索引タイプを使用するかを決定する必要があります。

シソーラスの基本コンポーネントは「用語」と「関係」です。

用語

用語とは、シソーラスの対象領域内の概念を表す語または表記のことです。たとえば、以下の用語は、1つまたは複数のシソーラス内に含まれるものです。

```
data processing
helicopter
gross national product
```

DB2 テキスト・エクステンダー・シソーラスでは、用語は記述用語または非記述用語のいずれかに分類することができます。記述用語とは、同義語クラス内の、索引付けおよび検索に望ましい用語のことです。クラス内のその他の用語は非記述用語と呼ばれます。たとえば、outline と shape は同義語ですが、shape を記述用語にし、outline を非記述用語にすることができます。

Ngram シソーラスでは、記述用語と非記述用語は区別されません。

関係

関係は、2つの用語間の関連を表現するものです。関係には、以下の特性があります。

- 関係の深さ は、関係が波及するレベルの数です。これは、検索構文内で THESDEPTH キーワードを使用することによって指定されます。照会でシソーラスの拡張を使用する方法については、165ページの『第12章 検索引き数の構文』を参照してください。
- 関係の方向性 は、その関係が1つの用語と他の用語間で等しく真である (両方向) か、一方向にのみ真である (単一方向) かを示します。

シソーラスの拡張では、シソーラス内に定義されているすべての関係を使用することができます。また、その拡張の深さも指定できます。この値が、ソース用語からターゲット用語への変換の最大数です。ただし、深さが増加すると、用語セットが指数関数的に増える点に注意してください。

次の例は、深さが増加されたときに、新たに追加された用語を示しています。

シソーラスの概念

health

health service, paramedical, medicine, illness

allergology, virology, veterinary medicine, toxicology, surgery, stomatology, rheumatology, radiotherapy, psychiatry, preventive medicine, pathology, odontology, nutrition, nuclear medicine, neurology, nephrology, medical check up, industrial medicine, hematology, general medicine, epidemiology, clinical trial, cardiology, cancerology

DB2 テキスト・エクステンダー・シソーラスの関係

DB2 テキスト・エクステンダー・シソーラスに備わっている関係には、以下のものがあります。

- 結合
- 同義
- 階層
- その他

DB2 テキスト・エクステンダー・シソーラスでは、事前定義の関係はありません。各関係には、BROADER TERM などの名前を付けることができ、これは BT のような簡略形にすることができます。シソーラスの設計で一般に使用される関係は以下のとおりです。

- BT または BROADER TERM
- NT または NARROWER TERM
- RT または RELATED TERM
- SYN または SYNONYM
- USE
- UF または USE FOR

結合: 結合関係は記述用語間の両方向関係で、任意の深さに拡張されます。この関係は、同等でも階層的でもないが、セマンティック的に関連した 2 つの用語を結び付けて、その連結が、索引付けまたは検索で使用される追加の用語を想起できるようにします。

結合関係は一般に RT (関連用語) として示されます。例:

dog RT security
pet RT veterinarian

同義: DB2 テキスト・エクステンダー・シソーラスにおける場合と同様に、記述用語と非記述用語の区別があるときには、同義関係は、同じまたは類似した意味をもつ 2 つの用語間の単一方向関係です。同義語クラスでは、いずれか 1 つの用語が記述用語として指定されます。それ以外の用語は、非記述用語と呼ばれます。記述用語と非記述用語との区別がない場合の同義関係の定義については、209 ページの『Ngram シソーラスの関係』を参照してください。

一般に指定される USE は、特定の非記述用語からその記述用語を示します。一般に指定される USE FOR は、記述用語から各非記述用語を示します。たとえば、次のようになります。

feline USE cat
lawyer UF advocate

階層: 階層関係は、記述用語間の単一方向関係で、ある用語が他の用語より特定化されている、すなわち汎用性がなくなっていることを表します。この差異によって、ある用語が 1 つのクラスを表し、それに従属する用語はそのメンバー部分を示すように用語が階層的に表されます。たとえば、「mouse」という用語は、「rodent」クラスに属します。

BROADER TERM と NARROWER TERM は階層関係を示します。たとえば次のようになります。

car NT limousine
equine BT horse

その他: その他 のタイプの関係は最も汎用性があります。これは、他のカテゴリーに容易に含められない関連を表します。その他 のタイプの関係には、単一方向も両方向もあり、深さに制限はありません。また、記述用語と非記述用語との間に複数の関係があることもあります。

このような関係は、他の用語との適切な関係が決まるまでの間、シソーラスの新しい用語によく使用されます。

もちろん、記述子間の同義関係に結合 の関係タイプを使用するか、または任意の用語間の同義関係にその他 の関係タイプを使用して、独自の両方向同義関係を定義することもできます。

Ngram シソーラスの関係

Ngram シソーラスは、次の 2 つのタイプをサポートします。

- 結合
- 同義

2 つ存在する事前定義関係のそれぞれは、これら 2 つのタイプのうちの 1 つに基づいています。結合 タイプに基づく独自の関係を定義することもできます。詳しくは、213 ページの『Ngram シソーラスの作成』を参照してください。

結合

結合関係は、概念は異なっていながらも相互に関連している 2 つの用語の間の両方向関係です。事前定義関係 *RELATED_TO* およびすべてのユーザー定義の関係は、この関係タイプに基づいています。

例:

tennis RELATED_TO racket
German RELATED_TO sausage

同義

同義関係は、同じまたは類似した意味をもち、互いに代替語として使用できる、2 つの用語間の両方向関係です。この関係は、たとえば、用語とその省略語に使用することができます。事前定義関係 *SYNONYM_OF* のみが、このタイプに基づいています。

シソーラスの概念

例:

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

シソーラスの作成

213 ページの『Ngram シソーラスの作成』も参照してください。

ディレクトリー /QIBM/ProdData/imo/dict には、英語のシソーラス・コンパイラー入力ファイルのサンプル imothes.sgm が保管されています。

同じディレクトリーに、このシソーラスのコンパイル済みバージョンが保管されています。

このシソーラスに属するファイルは、imothes.th1、imothes.th2、... そして imothes.th6 という名前となっています。

シソーラスを作成するためには、まず、ファイルの内容を定義します。それぞれのシソーラスの定義には、分かりやすいディレクトリーを使用することをお勧めします。ファイルの拡張子には、th1 ~ th6 (これらはシソーラス辞書に使用される) 以外のものを付けることができます。Ngram シソーラスと同じディレクトリーを使用する場合には、さらに限定されるファイル拡張子について、213 ページの『Ngram シソーラスの作成』を参照してください。

次に、以下のコマンドを実行して、ファイルをコンパイルします。

```
CALL PGM(QIMO/IMOTHESC) PARM ('-f' '<filename>' '-ccsid' '<CSSID>')
```

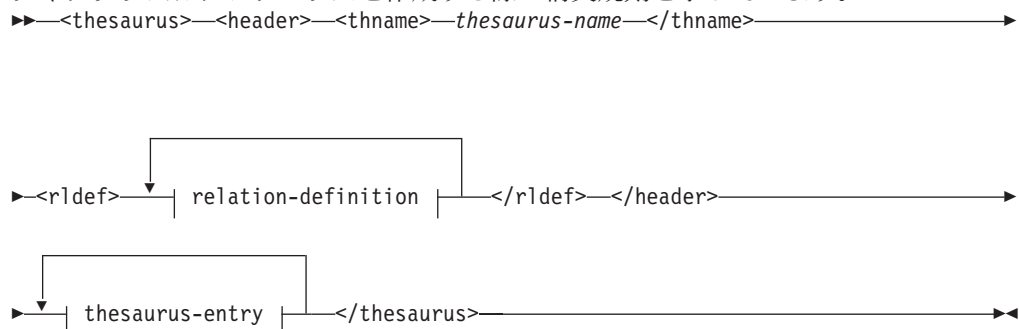
filename に使用できる文字は、a ~ z、A ~ Z、および 0 ~ 9 のみです。

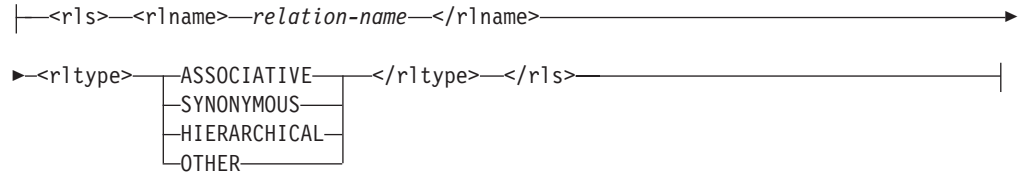
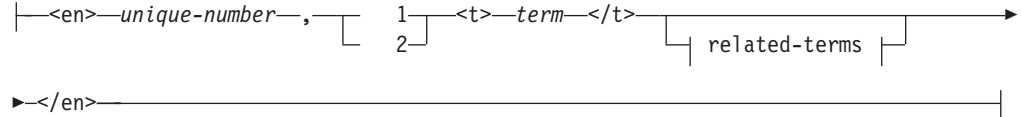
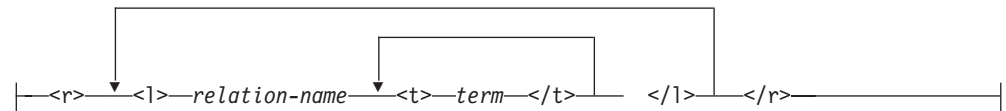
現時点では、CCSID 500 のみがサポートされています。

imothesc により、定義ファイルと同じディレクトリーに、*filename* という名前の、拡張子のないシソーラス・ファイル、および同じ名前で拡張子が th1 ~ th6 のシソーラス・ファイルが作成されます。すでに同じ名前のシソーラスがある場合には、警告なしでそれが上書きされます。

照会でシソーラスを使用する方法については、165 ページの『第 12 章 検索引き数の構文』を参照してください。

シソーラスの内容は、標準汎用印刷指定言語 (SGML) を使用して指定します。次のダイアグラムは、シソーラスを作成する際の構文規則を示しています。



relation-definition:**thesaurus-entry:****related-terms:**

relation-name に使用できる文字は、 a ~ z、A ~ Z、および 0 ~ 9 のみです。

212 ページの図 18 は、206 ページの図 17 に示されているシソーラスの SGML 定義を示しています。

シソーラスの概念

```
<thesaurus>
<header>
<thname>thesc example thesaurus</thname>
<rldef>

<rls>
<rlname>Related Term</rlname>
<rltype>associative</rltype>
</rls>

<rls>
<rlname>Narrower Term</rlname>
<rltype>hierarchical</rltype>
</rls>

<rls>
<rlname>Instance</rlname>
<rltype>hierarchical</rltype>
</rls>

<rls>
<rlname>Synonym</rlname>
<rltype>synonymous</rltype>
</rls>
</rldef>
</header>

<en> 2, 1
<t>database management system</t>
<r>
<l>Narrower Term
<t>oo database management system</t>
<t>relational database management system</t>
</l>

<l>Synonym
<t>DBMS</t>
</l>

<l>Related Term
<t>document management system</t>
</l>

<l>Instance
<t>database</t>
</l>
</r>
</en>
```

図 18. 簡単なシソーラスの定義 (1/2)


```

<en> 5, 1
<t> relational database management system </t>
<r>
  <l>Narrower Term
  <t>object relational database management system</t>
  </l>
</r>
</en>

<en> 3, 1
<t>object relational database management system</t>
<r>
  <l>Instance
  <t>DB2 Universal Database</t>
  </l>
</r>
</en>

<en> 6, 1
<t>object oriented database management system</t>
<r>
  <l>Narrower Term
  <t>object relational database management system</t>
  </l>
</r>
</en>

<en> 4, 1
<t>document management system</t>
<r>
  <l>Synonym
  <t>library</t>
  </l>
</r>
</en>

<en> 9, 1
<t>library</t>
</en>

<en> 10, 1
<t>DB2 Universal Database</t>
</en>

<en> 11, 1
<t>database</t>
</en>
</thesaurus>

```

図 18. 簡単なシソーラスの定義 (2/2)

Ngram シソーラスの作成

ディレクトリー /QIBM/ProdData/imo/dict には、英語の Ngram シソーラス・コンパイラー入力ファイルのサンプル imothes.def が保管されています。

このディレクトリーには、このサンプル・シソーラスのコンパイル済みバージョンも保管されています。このシソーラスに属するファイルは、imonthes.<拡張子> という名前になっています。拡張子は次のとおりです (n は数字)。

シソーラスの概念

- 辞書ファイルの場合: wdf、 wdv、 grf、 grv、 MEY、 ROS、 NEY、 SOS、 lkn
- 一時ファイルの場合: wnf、 wnv、 gnf、 gnv、 M!1、 M!2、 N!1、 N!2、 R!1、 R!2、 S!1、 S!2、 Mnn、 Nnn、 Rnn、 Snn、 \$00、 \$01、 \$10、 \$11、 \$20、 および \$21

Ngram シソーラスを作成するには、まず定義ファイルにその内容を定義します。同じディレクトリーに複数のシソーラスを入れることはできますが、それぞれのシソーラスに別個のディレクトリーを使用することをお勧めします。拡張子を除いたファイル名の長さは、8 文字以下でなければなりません。拡張子は任意指定ですが、3 文字までで、上記以外のものでなければなりません。

他の DB2 テキスト・エクステンダー・シソーラス用に同じディレクトリーを使用する場合、210 ページの『シソーラスの作成』に記載されている拡張子を使用してはなりません。

次に、以下のコマンドを実行して、ファイルをコンパイルします。

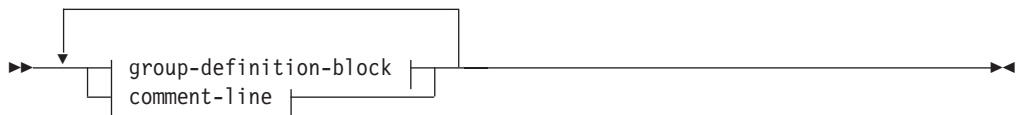
```
CALL PGM(QIMO/IMOTHESN) PARM('-f' <definition-file-name> ' -ccsid' <CSSID>')
```

以下に、Ngram シソーラスでサポートされているコード・ページをリストします。

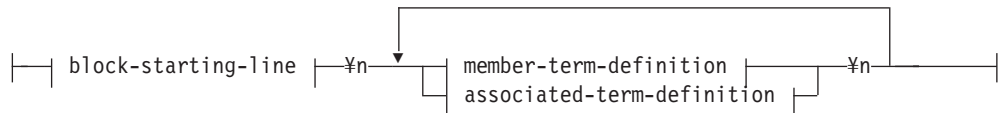
500、5035、1388、933、および 937。

imothesn により、上記の拡張子を持つ、*definition-file-name* と同じ名前のシソーラス・ファイルが作成されます。このファイルは、定義ファイルと同じディレクトリーに作成されます。すでに同じ名前のシソーラスがある場合には、警告なしでそれが上書きされます。

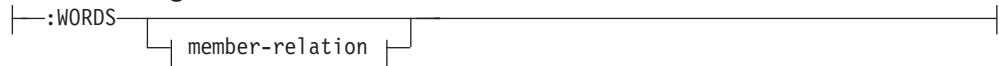
シソーラスの内容の指定には、次の構文図を使用します。



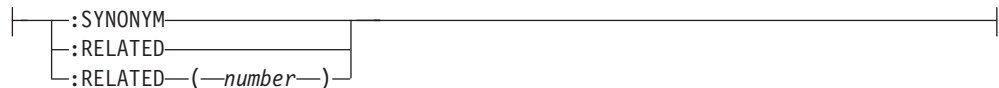
group-definition-block:



block-starting-line:



member-relation:



member-term-definition:

```
|—member-term—|
```

associated-term-definition:

```
|—associated-term—|
|  .RELATED_TO
|  .SYNONYM_OF
|  .RELATED_TO—(number)—|
```

comment-line:

```
|—#any-comment—|
```

それぞれのメンバー用語は単一行に書き込む必要があります。それぞれの関連用語の前には、関係名を付ける必要があります。メンバー用語が互いに関連している場合には、メンバー関係を指定します。

メンバー用語および関連用語の長さは、164 文字までに制限されています。同一文字の単一バイト文字と 2 バイト文字は、同じと見なされます。大文字小文字は区別されません。用語には空白文字を含めることができますが、単一バイト文字のピリオド (.) およびコロン (:) については、いずれか一方のみが使用できます。

ユーザー定義の関係はすべて、結合タイプに基づいています。これらの関係は 1 ~ 128 の固有の数字で識別されます。

アプリケーションでそのシソーラス関係を表すのに関係名や数値ではなく記号名を使用する場合には、アプリケーション側でマッピングの管理を行う必要があります。たとえば、OPPOSITE_OF 関係を RELATED_TO(1) と定義した場合、アプリケーションはこの名前を内部の関係名 RELATED_TO(1) にマップする必要があります。照会でシソーラスの拡張を使用する方法については、165 ページの『第 12 章 検索引き数の構文』を参照してください。

第 16 章 構成ファイル

この章では、構成ファイルについて説明します。これらのファイルは、`txicrt` コマンドを使用して DB2 テキスト・エクステンダー・インスタンスを作成する際、または `descrc1` コマンドを使用してクライアント・プロファイルを作成する際に、自動的に生成されます。これらのファイルはコード・ページ 500 で生成されます。

構成ファイルを編集してご使用のシステムを調整できますが、ファイルを編集する際には正しいコード・ページを使用しているか確認してください。セクション名とオプション名は、大文字と小文字を区別しません。コメント区切り文字としてセミコロンが使用されます。

オプションがブール値の場合は、値 `TRUE`、`YES`、`ON`、および `1` は大文字小文字にかかわらず真と見なされます。他の値はすべて偽と見なされます。

クライアント構成ファイル

ファイル名 `imocl.ini`

場所 `/QIBM/UserData/DB2Extenders/Text/instance`

オプションを更新すると、次回 `StartSession` 関数を実行する際にそれらのオプションがアクティブになります。

セクション	オプション	デフォルト値	説明
[INSTANCE]	DESWORKCL		一時ファイル用に使用される作業ディレクトリーを指します。
	DESNLPCL		リソース・ディレクトリーを指します。
	DESDTDPATHSRV		XML 文書の DTD が保管されるディレクトリーを指します。
	DESDTDPATHCL		DTD ファイルを保管しなければならない場所のパスを示します。XML ファイルの場合に使用されます。
[BUFFER]	BUFFERSEGMENTSZIE	32 000	バッファリングに使用されるブロック・セグメントのサイズ (バイト単位)。
	BUFFERSEGMENTCOUNT	3	バッファが一時ファイルにスワップされる前に使用されるセグメントの最大数。バッファ・セグメントは <code>BUFFERSEGMENTSZIE</code> によって定義されます。

クライアント構成ファイル

セクション	オプション	デフォルト値	説明
[DOCUMENTFORMAT]	USEREXIT		サポートされない文書形式の処理に使用されるユーザー出口の名前。ユーザー出口が PATH ステートメントの一部であるディレクトリに保管されている場合はファイル名を指定し、それ以外の場合は完全修飾ファイル名を指定してください。 形式変換用のユーザー出口に関する詳細は、24 ページの『サポートされていない文書形式の使用』を参照してください。
	FORMATRECOGNITION	TRUE	文書形式の形式認識を起動します。 TRUE: 形式認識がオン FALSE: 形式認識がオフ
	UseExitForAllFormats	FALSE	deslsdef.h にリストされていない文書形式を処理するユーザー出口が、いつ呼び出されるかを判別します。USEREXIT の値を設定しなければなりません。 TRUE: ユーザー出口は常時呼び出されます。この値を設定すると、FORMATRECOGNITION は無視されます。 FALSE: EHW_USER_FORMATS の値を上回るすべての文書形式の場合に、ユーザー出口を呼び出します。

サーバー構成ファイル

ファイル名 imosrv.ini

場所 /QIBM/UserData/DB2Extenders/Text/instance/txins000

オプションを更新すると、次回サーバー・インスタンスを開始する際にそれらのオプションがアクティブになります。

表 19. サーバー・構成ファイルのオプション

セクション	オプション	デフォルト値	説明
[INSTANCE]	DESWORKSRV		一時ファイル用に使用される作業ディレクトリを指します。
	DESNLPSSRV		リソース・ディレクトリを指します。

表 19. サーバー・構成ファイルのオプション (続き)

セクション	オプション	デフォルト値	説明
[DAEMON]	MaxMtEntries	30	一度に並列使用される索引の最大数。セマフォや共用メモリーなどのリソースが足りない場合は、この数値を小さくしてください。使用可能なリソースはプラットフォームに応じて異なるため、デフォルト値もプラットフォームに応じて異なります。
	MaxIndexEntries	1000	使用される索引の最大数。共用メモリーなどのリソースが足りない場合は、この数値を小さくしてください。
[BUFFER]	BUFFERSEGMENTSIZ	32 000	バッファリングに使用されるブロック・セグメントのサイズ (バイト単位)。これは EhwUpdate で使用されます。
	BUFFERSEGMENTCOUNT	3	索引の更新処理中の、バッファが一時的にファイルにスワップされる前に使用されるセグメントの最大数。文書のコレクションに大きな文書が含まれている場合は、この数値を大きくしてください。
	BUFFERSORTSIZE	4 000 000	一時作業ファイルのソートに使用されるバッファのサイズ (バイト単位)。この値の上限は 16 000 000 です。
[DOCUMENTFORMAT]	USEREXIT		deslsdef.h にリストされていない文書形式の処理に使用されるユーザー出口の名前。ユーザー出口が PATH ステートメントの一部であるディレクトリーに保管されている場合はファイル名を指定し、それ以外の場合は完全修飾ファイル名を指定してください。 形式変換用のユーザー出口に関する詳細は、24 ページの『サポートされていない文書形式の使用』を参照してください。
	FORMATRECOGNITION	TRUE	deslsdef.h にリストされていない文書形式の形式認識を起動します。 TRUE: 形式認識がオン FALSE: 形式認識がオフ

サーバー構成ファイル

表 19. サーバー・構成ファイルのオプション (続き)

セクション	オプション	デフォルト値	説明
	UseExitForAllFormats	FALSE	deslsdef.h にリストされていない文書形式を処理するユーザー出口が、いつ呼び出されるかを決定します。USEREXIT の値を設定しなければなりません。 TRUE: ユーザー出口は常時呼び出されます。この値を設定すると、FORMATRECOGNITION は無視されます。 FALSE: EHW_USER_FORMATS の値を上回るすべての文書形式の場合に、ユーザー出口を呼び出します。
[LINGPREC] 基本タイプが基本言語または一致語句であるすべての索引	UPDATETHRESHOLD	4 000 000	索引更新処理は、複数の更新処理と再編成処理に内部的に分割されます。この値は、1 つの更新ステップで収集される語の数を指定します。
	UPDATESLICE	1	内部再編成処理が開始される前に行われる更新処理の数。更新処理は UPDATETHRESHOLD によって定義されます。
[NGRAM] DBCS がサポートされているすべての索引	UPDATETHRESHOLD	10 000 000	1 つの更新処理中に索引に追加される文書の合計サイズ (バイト単位)。しきい値を超えると、再編成処理が自動的に開始されます。
	UPDATESLICE	10 000	2 次索引内の文書の最大数。この数値は、更新処理が終わるたびにチェックされます。文書の数がこの値より大きいと、再編成処理が自動的に開始されます。

第 17 章 戻りコード

この章では、DB2 テキスト・エクステンダー API が関数呼び出しに対して戻すコードを記載します。コードはアルファベット順に記載されています。

DB2 テキスト・エクステンダー API 呼び出しはすべて、数値の戻りコードを C 関数値として戻します。戻りコードは、DB2 テキスト・エクステンダーが提供する組み込みファイル DES_EXT.H で定義されています。

DB2 テキスト・エクステンダー API はエラー状態を代行受信し、そのエラー状態を戻りコードとともに報告します。

DB2 テキスト・エクステンダー API 関数を呼び出すアプリケーションでは、必ず初めに戻りコードをチェックしてから、他の出力パラメーターを処理するようにしてください。それぞれの呼び出しで戻される可能性のある戻りコードが、175 ページの『第 13 章 検索およびブラウズのための API 関数』にパラメーターとともに記載されています。

一部のケースでは、DB2 テキスト・エクステンダーが代行受信できない API サービスにおいて、使用されなくなったセッション・ポインターなどを誤って入力したために異常終了することがあります。

RC_ALLOCATION_ERROR

説明: 内部使用のためのストレージ割り当てができません。

ユーザーの処置: 使用可能なメモリーが十分にあるか確認してください。

RC_FILE_IO_PROBLEM

説明: DB2 テキスト・エクステンダーがファイルの読み取りまたは書き込みを行えませんでした。

ユーザーの処置: サーバーに使用可能なディスク・スペースとメモリーが十分にあるかチェックしてください。環境変数とテキスト構成設定値が正しく設定されているかチェックしてください。

RC_INVALID_BROWSE_INFO

説明: DesGetSearchResultTable または DesGetBrowseInfo によって戻され、かつ DesStartBrowseSession の入力に使用されるブラウズ情報が無効です。

ユーザーの処置: プログラミング・エラーによってブラウズ情報が一時変更されていないかチェックしてください。

RC_INVALID_BROWSE_OPTION

説明: DesGetSearchResultTable 内のブラウズ・オプションが無効です。

ユーザーの処置: オプションが BROWSE または NO_BROWSE であることを確認してください。

RC_INVALID_MATCH_OPTION

説明: DesOpenDocument で使用されている突き合わせオプションが無効です。

ユーザーの処置: オプションが FAST または EXTENDED であるかチェックしてください。

RC_INVALID_PARAMETER

説明: 入力パラメーターの 1 つに誤りがあります。

ユーザーの処置: DB2 テキスト・エクステンダーによって戻されるエラー・メッセージを読み、原因を判別してください。

RC_INVALID_SEARCH_OPTION

説明: DesGetSearchResultTable 内の検索オプションが無効です。

ユーザーの処置: オプションが

戻りコード

DES_TEXTHANDLEONLY、DES_RANK、DES_MATCH、または DES_RANKANDMATCH であることを確認してください。

RC_INVALID_SESSION

説明: 現行のサービス呼び出しに指定されているセッション・ポインターに誤りがあるか、または使用されなくなっています。

ユーザーの処置: エラーの原因究明に役立つ情報をすべて保管した後、アプリケーションを終了してください。

RC_NO_BROWSE_INFO

説明: DB2 テキスト・エクステンダーはブラウズ情報を何も戻りませんでした。これは、検索引き数が何も検索結果を得られなかったためです。これはエラーではありません。

ユーザーの処置: 何もする必要はありません。

RC_PARSER_INVALID_ESCAPE_CHARACTER

説明: 検索基準に誤ったエスケープ文字が含まれています。このエラーは、ブランクをエスケープ文字として使用した場合、または検索基準で 1 つの語あるいは句について 1 つ以上のエスケープ文字を指定した場合などに報告されます。例: ESCAPE " " または ESCAPE "#\$".

ユーザーの処置: 検索引き数の構文をチェックし、再試行してください。

RC_PARSER_INVALID_USE_OF_ESCAPE_CHAR

説明: 検索基準内のエスケープ文字の構文が解釈できません。

ユーザーの処置: エスケープ文字の構文をチェックしてください。たとえば、\$ が指定されたエスケープ文字である場合、語句に含めることができるのは、\$\$、\$_ または \$% のみです。ここで、_ および % は 2 つのマスキング記号を表しています。

RC_PARSER_SYNTAX_ERROR [: position]

説明: 検索基準構文は解釈することができません。位置が示されている場合は、その位置が、問題が検出された検索ストリング内の文字位置になります。これは、正確な位置ではない場合もあるため、指定された文字位置の前のいくつかの文字もチェックしてください。

ユーザーの処置: 165 ページの『第 12 章 検索引き数の構文』を参照し、検索引き数の構文をチェックしてください。

RC_RESULT_TABLE_NOT_EXIST

説明: 存在しない表内に検索結果を保管しようとしています。

ユーザーの処置: 188 ページの図 16 に示すように結果表を作成してください。

RC_SE_BROWSER_TIME_OUT

説明: ブラウズ・プロセスが開始されましたが、許容時間内に応答がありませんでした。そこで、DB2 テキスト・エクステンダーは保留状態のプロセスを取り消しました。

このエラーは、システムに十分なストレージがないか、負荷がかかり過ぎる場合に起こる可能性があります。

ユーザーの処置: DesEndBrowseSession を呼び出してブラウズ・セッションを終了させ、DesFreeBrowseInfo で割り振り済みストレージを解放し、それから再試行してください。

RC_SE_CAPACITY_LIMIT_EXCEEDED

説明: 要求された機能は処理できません。メモリーまたはディスク・スペースが不足しています。

ユーザーの処置: プログラムを終了し、システムのリソースを調べてください。

RC_SE_COMMUNICATION_PROBLEM

説明: DB2 テキスト・エクステンダー・サーバーとの通信に失敗しました。ストレージが足りなかったか DB2 テキスト・エクステンダーのインストールに誤りがあったためにエラーが発生したと考えられます。

ユーザーの処置: エラー検出に役立つ情報をすべて保管し、アプリケーションを終了させてください。

RC_SE_CONFLICT_WITH_INDEX_TYPE

説明: 照会の検索項目の基本語形指定が、索引のタイプに対応していません。たとえば、PRECISE FORM OF を基本語形索引で使用することはできません。デフォルトの基本語形指定は、171 ページの表 7 に示すように使用されます。

ユーザーの処置: アプリケーションを変更して、その索引のタイプと対立する照会オプションを指定しないようにしてください。

RC_SE_DICTIONARY_NOT_FOUND

説明: DB2 テキスト・エクステンダー基本語形サービスが辞書ファイルを検出できません。照会は言語的なサポートなしに処理されます。指定の言語コード (複数の場合もある) に対応する辞書ファイルが、あるべきパスにありません。

ユーザーの処置: API 呼び出しを継続してください。UNIX の場合、必要な辞書がパス {DB2TX_INSTOWNERHOMEDIR}/db2tx/dicts に入っているかどうかをチェックしてください。必要であれば、必要な辞書をインストールしてください。

RC_SE_DOCMOD_READ_PROBLEM

説明: DB2 テキスト・エクステンダー・インスタンスの作成時に、文書モデル・ファイル desmodel.ini がインスタンス・ディレクトリーに保管されます。索引の作成時に、desmodel.ini ファイルは索引ディレクトリー IXnnnnnn にも保管されます。この文書モデル・ファイルを読み取れませんでした。

ユーザーの処置: 文書モデル・ファイルが存在し、正しいディレクトリーに入っていることをチェックしてください。

RC_SE_DOCUMENT_NOT_ACCESSIBLE

説明: 要求されたテキスト文書が検出されましたが、現在はアクセスできません。

ユーザーの処置: その文書が、他のタスクやユーザーによって排他的に使用されていないかチェックしてください。

RC_SE_DOCUMENT_NOT_FOUND

説明: 要求されたテキスト文書が見つかりませんでした。その原因としては、第一に、テキスト文書がストレージから削除されてしまっているが、DB2 テキスト・エクステンダーの索引からはまだ削除されていないことが考えられます。また、損傷を受けたハンドルによって識別された文書をブラウズしようとする場合にも起こることがあります。

ユーザーの処置: ほとんどの場合、この戻りコードは無視しても問題ありません。この戻りコードは、次の索引更新時以降は表示されなくなります。

もし表示される場合は、アプリケーション・プログラムが検出ハンドルをブラウズ用に正常に渡しているかチェックしてください。

RC_SE_EMPTY_INDEX

説明: 検索要求によってアドレッシングされたハンドル列に対応する DB2 テキスト・エクステンダー索引が空です。すなわち、索引にテキスト文書が何も追加されなかったか、またはすべてのテキスト文書が索引から除去されたかのいずれかです。

これは、テキスト列を使用可能にしたものの、列内の文書にまだ索引が付けられていない場合に起こる可能性があります。つまり、後で (周期的な索引付け設定により決められたときに) 索引を作成することを ENABLE TEXT COLUMN コマンド内で指定していたということです。

これは、テキスト表を使用可能にしてすべてのテキスト列に関する空の共通索引を作成してあるが、テキスト列はどれも使用可能になっていなかった場合にも起こることがあります。

ユーザーの処置: ENABLE TEXT TABLE ですべてのテキスト列に関する空の共通索引を作成した場合は、検索するテキストを含むテキスト列の少なくとも 1 つに ENABLE TEXT COLUMN を実行してください。このコマンドでは、索引を即時に作成するか、または定期的な索引付け設定によって指定された時点で作成するかを決めることができます。

GET INDEX STATUS を実行して、索引が正常に作成されたことをチェックしてください。

RC_SE_EMPTY_QUERY

説明: 指定の検索基準は、DB2 テキスト・エクステンダーによって言語学的に分析されて処理されました。プログラミング・エラーのために照会に検索項目が含まれなくなったか、またはすべての検索項目が、照会から除去された停止ワード (DB2 テキスト・エクステンダーによる索引付けをされていない語) であるかのいずれかです。結果は検索項目ではありませんでした。

ユーザーの処置: 照会を繰り返してください。それでも問題が解決されない場合には、プログラミング・エラーをチェックしてください。

RC_SE_END_OF_INFORMATION

説明: これはエラーではありません。文書の終わりに到達しました。DesGetMatches に関する情報をこれ以上得ることはできません。

ユーザーの処置: この戻りコードを使用して、DesGetMatches による文書の反復処理を終了させてください。

戻りコード

RC_SE_FUNCTION_DISABLED

説明: 要求された関数が、管理者により禁止されている DB2 テキスト・エクステンダー関数を呼び出しました。

ユーザーの処置: 管理担当者に支援を依頼してください。DB2 テキスト・エクステンダーをいったん停止して再始動する必要がある場合があります (txstop/txstart)。

RC_SE_FUNCTION_IN_ERROR

説明: DB2 テキスト・エクステンダー・サーバー上のエラーのために、要求された関数がロックされています。API 呼び出しは処理できません。

ユーザーの処置: 索引状況をチェックしてください。索引ディレクトリーで使用できるスペースをチェックしてください。索引状況をリセットし、コマンドを再実行してください。

RC_SE_INCORRECT_HANDLE

説明: 入力パラメーターに指定されたハンドル (たとえばブラウズ・セッション・ハンドル) が無効です。このハンドルは、直前の呼び出しで戻されて現在は使用されていないものと推定されます。

ユーザーの処置: エラーの原因究明に役立つ情報をすべて保管した後、DesEndBrowseSession を呼び出してセッションを終了してください。

プログラミング・エラーによって誤ったハンドルが作成されていないかチェックしてください。

RC_SE_INDEX_DELETED

説明: アクセスした DB2 テキスト・エクステンダーの索引が削除されています。

ユーザーの処置: DB2 テキスト・エクステンダー管理者に連絡し、索引を作成し直してください。

RC_SE_INDEX_NOT_ACCESSIBLE

説明: DB2 テキスト・エクステンダーの索引にアクセスできず、現行の呼び出しは処理できません。

ユーザーの処置: DB2 テキスト・エクステンダー管理者に連絡を取り、索引へのアクセスが可能かどうかをチェックしてください。

RC_SE_INDEX_SUSPENDED

説明: DB2 テキスト・エクステンダーが、他のセッションまたは現行のセッションを延期している索引に関する要求を受信しました。

ユーザーの処置: DB2 テキスト・エクステンダー管理者に連絡を取り、索引の状況をチェックしてください。

RC_SE_INSTALLATION_PROBLEM

説明: DB2 テキスト・エクステンダーで、インストール上の問題が検出されました。

ユーザーの処置: 環境変数 DB2INSTANCE、DB2TX_INSTOWNER、DB2TXINSTOWNERHOMEDIR の現行の設定をチェックしてください。descfgcl -d および descfgsv -d -i txinsnnn を使用して、検索サービス構成をチェックしてください。

RC_SE_IO_PROBLEM

説明: サーバーがそのサーバーの索引ファイルの 1 つをオープンして読み取ろうとした際に、エラーが発生しました。この原因としては、以下のいずれかが考えられます。

管理者が意図せずに、DB2 テキスト・エクステンダー索引ファイルの削除などを行ってしまった場合
テキスト構成内の設定に誤りがある場合

ユーザーの処置: アプリケーションを終了してください。管理者と一緒に以下の点をチェックしてください。

- 現行の DB2 テキスト・エクステンダー索引の全ファイルが存在している。
- テキスト構成設定が正しい。

RC_SE_LS_FUNCTION_FAILED

説明: ブラウズの目的でテキスト文書を検索するためにデータベースにアクセスした関数が、失敗しました。ユーザーがデータベースにアクセスできなくなったか、またはユーザーにテキスト表に関する権限が与えられていません。

ユーザーの処置: ユーザー ID などの関数への入力が正しいかチェックしてください。データベースがアクセス可能か、またこのタスクに関する権限がユーザーに与えられているかチェックしてください。

RC_SE_LS_NOT_EXECUTABLE

説明: ブラウズの目的でデータベースにアクセスしてテキスト文書を検索しようとしている関数を実行できません。

ユーザーの処置: DB2 テキスト・エクステンダーが正常にインストールされているかチェックしてください。問題が解決できない場合は、最寄りの IBM 担当員に連絡してください。

RC_SE_MAX_OUTPUT_SIZE_EXCEEDED

説明: 異常に多くの一致が検出されました。ブラウズ情報のサイズが、処理可能な最大サイズを超えています。要求は処理できません。

ユーザーの処置: 照会をより特定のものに行うか、またはシステム・メモリーが拡張可能かどうか確認してください。

RC_SE_MAX_NUMBER_OF_BUSY_INDEXES

説明: 要求された関数の実行が検索サービスによって阻止されました。現在活動状態にある索引が最大数に達しているためです。

ユーザーの処置: 少し時間が経過してから関数呼び出しを再発行してください。通常、問題は一時的なものです。

RC_SE_NO_DATA

説明: これはエラーではありません。検索基準に一致するテキスト文書がありません。ブラウズ情報を要求しても、戻される情報はありません。ブラウズ情報用にストレージが割り当てられていません。

ユーザーの処置: 何もする必要はありません。

RC_SE_NOT_ENOUGH_MEMORY

説明: クライアントまたはサーバー・システム上に十分なストレージがありません。現行の要求は処理できません。

ユーザーの処置: ストレージを解放し、アプリケーションを終了してください。

RC_SE_PROCESSING_LIMIT_EXCEEDED

説明: 現行の検索要求が、クライアント / サーバー環境用に指定された最大結果サイズまたは最大処理時間を超えています。要求は取り消されました。

ユーザーの処置: 検索要求をより特定の範囲に制限してください。最大処理時間の延長を検討してください。

RC_SE_QUERY_TOO_COMPLEX

説明: 指定された照会が複雑過ぎます。

ユーザーの処置: アプリケーションを変更して、マスキング文字および同義語を過度に使用しないようにしてください。

マスキング記号または SYNONYM オプションを使い過ぎると、照会の範囲が、DB2 テキスト・エクステンダーで管理できないサイズにまで広がってしまうことがあります。

RC_SE_REQUEST_IN_PROGRESS

説明: DB2 テキスト・エクステンダー・ブラウズ API サービスが呼び出されましたが、同一セッションで別のブラウズ API 要求が活動状態でした。

ユーザーの処置: DesEndBrowseSession を呼び出してセッションを終了し、DesFreeBrowseInfo でストレージを解放してください。

DB2 テキスト・エクステンダー・ブラウズ API は、同一ブラウズ・セッションに対する並行アクセスをサポートしていません。

同一プロセスで並行実行中のアプリケーションはすべて、そのアプリケーション自体のブラウズ・セッションを処理するようにしてください。

RC_SE_SERVER_BUSY

説明: DB2 テキスト・エクステンダー・クライアントは、要求された DB2 テキスト・エクステンダー・サーバーとのセッションを現在確立できません。または DB2 テキスト・エクステンダー・サーバー通信リンクに割り込みが行われて確立し直すことができません。

DB2 テキスト・エクステンダー・サーバーは正常に開始されましたが、並列サーバー処理の数が最大値に達しました。

ユーザーの処置: これが一時的な問題ではない場合は、サーバー上で通信の構成を変更してください。

RC_SE_SERVER_CONNECTION_LOST

説明: クライアントとサーバーとの間の通信に割り込みが行われ、再度確立することができません。

管理者が DB2 テキスト・エクステンダー・サーバー・タスクを停止したか、またはサーバー・ワークステーションが遮断した可能性があります。

ユーザーの処置: これらの条件のどちらかが発生していないかチェックし、訂正を行ってください。

戻りコード

RC_SE_SERVER_NOT_AVAILABLE

説明: DB2 テキスト・エクステンダー API サービスは、要求された DB2 テキスト・エクステンダー・サーバーとのセッションを確立できませんでした。

DB2 テキスト・エクステンダー・サーバーが開始されていない可能性があります。

ユーザーの処置: DB2 テキスト・エクステンダー・サーバーが正常に開始されているかチェックしてください。エラーが解決しない場合は、インストールに問題があります。

RC_SE_STOPWORD_IGNORED

説明: この情報コードは、指定された照会に停止ワードのみから成る検索項目が少なくとも 1 つ含まれていた場合に戻されます。検索項目は照会の処理の間無視されました。

ユーザーの処置: API 呼び出しを継続しても問題ありません。DB2 テキスト・エクステンダー照会では停止ワードを使用しないでください。

RC_SE_UNEXPECTED_ERROR

説明: DB2 テキスト・エクステンダーのインストール方法に誤りがあったことが原因と思われるエラーが発生しました。

ユーザーの処置: エラーの原因究明に役立つと思われる情報を保管し、アプリケーションを終了させてください。

RC_SE_UNKNOWN_INDEX_NAME

説明: テキスト列に関連したテキスト索引の名前がハンドルの一部になっています。

ユーザーの処置: DesGetBrowseInfo への入力として使用するハンドルが正しいかどうかを確認してください。

RC_SE_UNKNOWN_SECTION_NAME

説明: 指定したセクション名は、文書モデル・ファイル中に指定されているモデルや、使用されているデフォルト・モデルの一部ではありません。

ユーザーの処置: 指定されているモデルまたはデフォルト・モデルの一部であるセクション名を指定してください。

RC_SE_UNKNOWN_SERVER_NAME

説明: DB2 テキスト・エクステンダー・サーバーの名前がハンドルの一部になっています。

ユーザーの処置: DesGetBrowseInfo への入力として使用するハンドルが正しいかどうかを確認してください。

RC_SE_WRITE_TO_DISK_ERROR

説明: DB2 テキスト・エクステンダー・サーバー・ワークステーション上でディスクがいっぱいになったため、または DB2 テキスト・エクステンダーのインストール方法に誤りがあったために発生した書き込みエラーです。

ユーザーの処置: エラーの原因究明に役立つと思われる情報を保管し、アプリケーションを終了させてください。サーバーのディスク・スペースに十分空きがあるかチェックしてください。

RC_SQL_ERROR_WITH_INFO

説明: SQL エラーが発生しました。エラー・メッセージが戻されます。

ユーザーの処置: 詳しくは、SQL エラー・メッセージや SQLState およびネイティブの SQL エラー・コードなどの、DB2 テキスト・エクステンダーから戻されるエラー・メッセージをチェックしてください。

RC_SQL_ERROR_NO_INFO

説明: SQL エラーが発生しました。エラー・メッセージは戻されません。

RC_TEXT_COLUMN_NOT_ENABLED

説明: 指定されたハンドル列は、指定の表内の列ではありません。

ユーザーの処置: 指定したハンドル列名が正しいかどうかチェックしてください。その表内のテキスト列が使用可能になっているか確認してください。

第 18 章 メッセージ

この章では、以下の内容について説明します。

- **DB2 テキスト・エクステンダーの関数により戻される SQL の状態:** これらのメッセージは、DB2 テキスト・エクステンダーの関数を使用している際に、表示されることがあります。
- **DB2TX コマンド行プロセッサからのメッセージ:** これらのメッセージは、コマンド行プロセッサ DB2TX を使用して管理コマンドを入力した際に、表示されることがあります。個々のメッセージ番号の接頭部は DES です。

DB2 テキスト・エクステンダーの関数により戻される SQL の状態

DB2 テキスト・エクステンダーが備えている SQL 関数は、エラー状態に戻すことができます。例:

```
SQL0443N User-defined function
"DB2TX.CONTAINS" (specific name "DES5A")
has returned an error SQLSTATE with
diagnostic text "Cannot open message file".
SQLSTATE=38702
```

このセクションのメッセージは、SQLSTATE 番号順に記載されています。

01H10 ファイル *file-name* を開けません。

ユーザーの処置: ファイルが存在するか、DB2 のインスタンス名にそのファイルをオープンする許可が与えられているかどうかを確認してください。

01H11 テキスト・ハンドルが完了していません。

説明: 初期化されているハンドルを使用しようとしたが、完了しませんでした。文書の言語と形式についての事前設定値が入っている INIT_TEXT_HANDLE を使って、部分ハンドルが作成されました。しかし、そのハンドルはトリガーにより完了しませんでした。

ユーザーの処置: 完了したハンドルだけを使用してください。そのハンドルをハンドル列に保管する場合は、これに対応するテキスト列を使用可能にするか再使用可能にしてください。

01H12 検索引き数が長過ぎます。2 番目の検索引き数は無視されました。

説明: REFINE 関数を使用して 2 つの検索引き数が結合されましたが、結合された検索引き数の長さが LONG VARCHAR の最大値を超えています。REFINE 関数は、結合された検索引き数の代わりに最初の検索引き数を戻します。

ユーザーの処置: 検索引き数のどちらか一方または両方の長さを短縮してから、照会を繰り返してください。

01H13 検索引き数は停止ワードを含みます。

説明: 指定された照会には、停止ワードだけからなる少なくとも 1 つの検索用語が含まれています。検索項目は照会の処理の間無視されました。

ユーザーの処置: DB2 テキスト・エクステンダー照会では停止ワードを使用しないでください。

01H14 言語処理のための言語辞書がありません。

説明: DB2 テキスト・エクステンダー基本語形サービスが辞書ファイルを検出できません。照会は言語的なサポートなしに処理されます。指定の言語コード (複数の場合もある) に対応する辞書ファイルが、あるべきパスにありません。

ユーザーの処置: UNIX の場合は、必要な辞書がパス {DB2TX_INSTOWNERHOMEDIR}/db2tx/dicts に入っているかどうかをチェックしてください。必要であれば、必要な辞書をインストールしてください。

DB2 テキスト・エクステンダーの関数により戻される SQL の状態

01H15 言語検索項目の指定が索引タイプと一致しません。

説明: 照会の検索項目の基本語形指定が、索引のタイプに対応していません。たとえば、PRECISE FORM OF は基本語形索引と一緒に使用できません。デフォルトの基本語形指定は、171 ページの表 7 に示すように使用されます。

ユーザーの処置: アプリケーションを変更して、その索引のタイプと対立する照会オプションを指定しないようにしてください。

38700 テキスト・エクステンダーのライブラリーが現行のものではありません。

説明: 後続バージョンの DB2 テキスト・エクステンダーでしか解釈できないハンドルを使用しようとした。

ユーザーの処置: 現行ライブラリーのバージョンに対するパスが正しく設定されているか、また、このライブラリーへのアクセス許可が与えられているか確認してください。

この問題が起こる関数については、IMPLEMENTATION 列にある DB2 カタログ・ビュー SYSCAT.FUNCTIONS を参照してください。

38701 このトレース・ファイル *tracefile* をオープンすることができません。

説明: ディレクトリー DB2TX_TRACEDIR 内のファイル DB2TX_TRACEFILE に書き込みを行うトレース機能を使用しようとした。ファイルが存在しないか見つからない、またはファイルに必要な許可が与えられていないかのいずれかです。

38702 メッセージ・ファイル *message-file* をオープンすることができません。

説明: DB2 テキスト・エクステンダーがメッセージを戻そうとするような状態が発生しました。メッセージを格納しているファイルが存在しないか見つからない、または使用が許可されていないかのいずれかです。

ユーザーの処置: ファイルが存在するか、パスの設定が正しいか、さらにファイルをオープンする許可が与えられているかどうかを確認してください。

38704 テキスト・ハンドルの形式が誤りです。

説明: DB2 テキスト・エクステンダーの関数の引き数として、正しくない形式のハンドルが使用されました。

ユーザーの処置: ハンドルが INIT_TEXT_HANDLE に

よって生成されたものではないことを確認してください。

38705 UDF 名 誤った UDF 宣言です。

説明: DB2 テキスト・エクステンダーの関数の特定の名前が、関数を宣言しているスクリプト内で変更されています。DB2 テキスト・エクステンダーの関数の名前は変更可能ですが、関数の特定の名前は変更できません。

ユーザーの処置: DB2 テキスト・エクステンダーの関数の宣言が入っているスクリプト DESCVDF.DDL をチェックして、まだ正しい名前が使用されているかどうかを確認します。名前が元の配布メディア内に保管されているものと同じかチェックしてください。

38706 属性 この属性値を認識できません。

説明: CCSID、形式、または言語に無効な値を設定しようとした。

ユーザーの処置: 正しい値については、21 ページの『第 4 章 必要に応じた検索の計画』を参照してください。

38707 要求された機能はまだサポートされていません。

説明: 指定された機能はまだサポートされていません。

ユーザーの処置: 指定された機能をチェックしてください。

38708 戻りコード

説明: 検索要求の処理中にエラーが発生しました。

ユーザーの処置: 221 ページの『第 17 章 戻りコード』の戻りコードに関する説明を参照してください。

38709 使用可能なメモリーが不十分です。

説明: メモリー不足のため、DB2 テキスト・エクステンダーの関数を実行できません。

ユーザーの処置: 不要なアプリケーションをクローズしてメモリーを解放してから、再度実行してください。

38710 エラー番号 検索結果にアクセスできません。

説明: 検索サービスにより戻された検出文書のリスト(結果リスト)を読み取ろうとした場合に発生するエラーです。

ユーザーの処置: 検索を繰り返してください。それでも

DB2 テキスト・エクステンダーの関数により戻される SQL の状態

問題が解決されない場合は、検索サービスを再始動してください。それでもエラーが起こる場合は、最寄りの IBM 担当員にエラー番号を添えてご連絡ください。

38711 重大な内部エラーです。

説明: 重大エラーが発生しました。

ユーザーの処置: 最寄りの IBM 担当員に連絡し、重大エラーが発生したときの状況をお知らせください。

38712 索引名 このテキスト索引の中のテキスト・ハンドルは誤りです。

説明: ハンドルが損傷しています。

ユーザーの処置: UPDATE INDEX を使用して索引を作成し直してください。

38714 DB2TX_INSTOWNERHOMEDIR 環境変数を短くしてください。

説明: インスタンス所有者のホーム・ディレクトリー名は、256 文字以下でなければなりません。

ユーザーの処置: リンクによりディレクトリー名の長さを短縮してください。

38717 指定されたシソーラス (類語辞典) が見つかりませんでした。

説明: 指定されたシソーラスが見つかりません。

ユーザーの処置: 指定されたシソーラス名をチェックしてください。

38718 指定された関係名をシソーラスの中で見つけることができませんでした。

説明: 指定された関係が指定されたシソーラスの中に存在しません。

ユーザーの処置: 指定した関係が存在することを確認してください。

38719 検索処理エラーが発生しました。理由コード: rc

説明: 示されている理由のために検索ができませんでした。

ユーザーの処置: 理由コードで報告される問題の解決を試みてください。示されている理由が役に立たず、それ以上の情報が desdiag.log ファイルに見つからない場合、トレースを作成してその情報を最寄りの IBM 担当員に報告してください。

38720 共用メモリーの接続エラーが発生しました。

説明: システムは共用メモリーにアクセスできません。

ユーザーの処置: システム構成をチェックして共用リソースを増やすか、または現行の共用リソースの使用 (ipcs) をチェックして必要のないリソースをクリーンアップしてください。

38721 セマフォの作成またはアクセス・エラーが発生しました。

説明: システムはセマフォの作成またはアクセスができません。

ユーザーの処置: システム構成をチェックして共用リソースを増やすか、または現行の共用リソースの使用 (ipcs) をチェックして必要のないリソースをクリーンアップしてください。

38722 検索処理が戻りませんでした。

説明: 検索要求の処理中にエラーが発生しました。

ユーザーの処置: システム構成 descfgcl を検査してすべてのノードが起動して、稼働中であるかチェックしてください。

38723 索引および照会 CCSID が一致しません。

説明: 照会ストリングで使用されるデータベース CCSID がテキスト索引と同じではありません。

ユーザーの処置: テキスト索引を使用不可にした後データベースの CCSID を使用して再作成してください。

38724 セクションまたはモデル名が正しくありません。

説明: 照会で指定されたセクションまたはモデル名が正しくありません。

ユーザーの処置: セクションまたはモデル名をチェックしてください。

38726 モデル・ファイルの読み取りエラーが発生しました。

説明: モデル定義ファイルが存在しないか、またはオープンできません。

ユーザーの処置: モデル定義ファイルが索引ディレクトリーに存在するかチェックしてください。

DB2 テキスト・エクステンダーからのメッセージ

各メッセージには、接頭部 (DES)、メッセージ番号、および接尾部文字から成るメッセージ ID が付いています。接尾部の文字は、そのメッセージを出したエラーの重大度を表します。

- I** 通知メッセージ
- W** 警告メッセージ
- N** エラー (または「否定」) メッセージ
- C** 重大エラー・メッセージ

DES0001N db2txinstance 引き数の数値が無効です。

説明: db2txinstance コマンドには 2 つの引き数が必要です。

ユーザーの処置: これらの引き数と共にコマンドを再入力してください。

```
db2txinstance instanceName db2InstanceName
```

ここで、*instanceName* はこのインスタンスの所有者として割り当てられている既存の UNIX ユーザーのログイン名で、*db2InstanceName* は対応する DB2 インスタンスの所有者のログイン名です。

DES0002N 無効なインスタンス名です。

説明: 指定するインスタンス名は既存の UNIX ユーザーのログイン名でなければなりません。

ユーザーの処置: インスタンス名を訂正するか、既存の UNIX ユーザーを選択するか、またはインスタンスの所有者となる UNIX ユーザーを作成してください。

db2txinstance コマンドを次のように再入力してください。

```
db2txinstance instanceName
```

ここで、*instanceName* は、選択された UNIX ユーザーのログイン名です。

DES0004N 指定されたインスタンスは、すでに存在します。コマンドを処理できません。

説明: *instanceName* は、インスタンスの所有者である UNIX ユーザーのログイン名を示します。このインスタンス所有者は、ホーム・ディレクトリーに db2tx ディレクトリーをすでに持っています。

ユーザーの処置: インスタンスを作成するには、既存のインスタンスを除去してからコマンドを再実行してください。

DES0005N インストール・メッセージのカタログを検出できません。

説明: インストール・スクリプトに必要なメッセージ・カタログがシステムにありません。削除されたか、またはデータベース・プロダクトが正しくロードされていない可能性があります。

ユーザーの処置: db2tx_01_01_0000.client プロダクト・オプションが正しくインストールされていることを確認してください。検査エラーが見つかった場合は、そのオプションをインストールし直してください。

DES0015W 言語検索項目の指定が索引タイプと一致しません。

説明: 照会の検索項目の基本語形指定が、索引のタイプに対応していません。たとえば、PRECISE FORM OF は基本語形索引と一緒に使用できません。デフォルトの基本語形指定は、171 ページの表 7 に示すように使用されます。

ユーザーの処置: アプリケーションを変更して、その索引のタイプと対立する照会オプションを指定しないようにしてください。

DES0016W 現行索引タイプに対してサポートされていない言語が指定されています。

説明: ユーザーが指定した言語は、指定された索引タイプに対してサポートされていません。

ユーザーの処置: 索引タイプに対してサポートされている言語のリストについては資料を参照してください。

DES0017W 特徴抽出が使用可能ではありません。

説明: 照会でフィーチャー検索引き数が使用されましたが、索引が、索引オプション FEATURE_EXTRACTION を指定して作成されていません。

ユーザーの処置: 索引オプション FEATURE_EXTRACTION はサポートされていません。

DES0018W 現行索引タイプに対してサポートされていないオプションが指定されています。

説明: ユーザーが要求した検索オプションは、現行索引タイプおよび索引オプションに対してサポートされていません。

ユーザーの処置: 要求した検索オプションをサポートする索引タイプまたは索引オプションをチェックしてください。171 ページの表 7 を参照してください。

DES0121N メモリーを割り振りができませんでした。

説明: アプリケーションをインストールするためのストレージを予約できませんでした。

ユーザーの処置: ページ・スペースを増やしてください。

DES0333N クライアントが、要求されたサーバーとのセッションを確立できません。

説明: DB2 テキスト・エクステンダー・クライアントは、要求されたサーバーとのセッションを確立できませんでした。

ユーザーの処置: DB2 テキスト・エクステンダー・サーバーが開始されているかを確認してください。開始していない場合は、TXSTART を実行します。

DES0377N テキスト索引ファイルの入出力に問題が発生しました。

説明: DB2 テキスト・エクステンダーがテキスト索引にアクセスできません。これは、テキスト構成内の DIRECTORY 設定が無効な辞書を指し示す場合に起きる可能性があります。

ユーザーの処置: テキスト構成設定を確認してください。

DES0500N IBM Text Search Engine (5722DE1、オプション 3) が正しくインストールされていません。

説明: IBM Text Search Engine は、DB2 テキスト・エクステンダーに対する前提条件のライセンス・プログラム・オプションです。これは、システムにインストールされていません。

ユーザーの処置: ライセンス・プログラム 5722DE1 のオプション 3 をインストールしてから再試行してください。

DES0700N ノード番号の値 'node' はノード・グループの定義に含まれていません。

説明: 指定されたノード番号は無効です。

ユーザーの処置: DB2 ノード番号をチェックしてください。

DES0701N ノード番号の値 'node' は範囲外です。

説明: 指定されたノード番号は無効です。

ユーザーの処置: DB2 ノード番号をチェックしてください。

DES0704N フォーマット 'format' は索引プロパティの指定が必要です。

説明: 文書の形式は索引タイプの情報と互換性がありません。

ユーザーの処置: 文書の形式と互換性のある索引プロパティを指定してください。

DES0705N 指定された文書のモデル名 'model' がモデル定義ファイルに見つかりませんでした。

説明: モデル定義ファイル中に文書モデル名がありませんでした。モデル名には大文字小文字の区別がある点に注意してください。

ユーザーの処置: モデル定義ファイルで指定されているモデル名を使用してください。

DES0706N DB2 テキスト・エクステンダーのサーバーでモデル定義ファイルにアクセスできません。

説明: モデル定義ファイルが存在しないか、またはオープンできません。

ユーザーの処置: モデル定義ファイルが存在することをチェックしてください。

DES0707N フォーマット 'format' は指定された索引プロパティをサポートしません。

説明: 文書の形式ではこの索引プロパティがサポートされていません。

ユーザーの処置: 文書の形式と互換性のある索引プロパティを指定してください。

DB2 テキスト・エクステンダーからのメッセージ

DES0709W 言語の辞書が未インストールです。

説明: DB2 テキスト・エクステンダーが、辞書ファイルを見つけることができません。

ユーザーの処置: 指定した言語の辞書をインストールするか再インストールしてください。

DES0710N ヌル・ポインターはパラメーター '*parameter*' には許可されていません。

説明: *parameter* に値が指定されていません。

ユーザーの処置: パラメーターに値を指定してください。

DES0711N テキスト・エクステンダーの内部エラーが発生しました。診断情報: *message*。

説明: 内部処理エラーが発生しました。

ユーザーの処置: 診断メッセージをチェックして、問題を解決してください。内部エラーの原因がインストールに関する問題でない場合は、*desdiag.log* ファイルまたは作成されたトレース・ファイルに追加情報がある可能性があります。それでも解決できない場合は、利用可能な情報を収集して IBM 技術員に連絡してください。

DES0712N パラメーター '*parameter*' は長すぎます。

説明: 指定されたパラメーターは範囲外です。

ユーザーの処置: 有効な長さを使用してパラメーターを指定してください。

DES0713N パラメーター '*parameter*' の長さは誤りです: %d1。

説明: 指定されたパラメーターは範囲外です。

ユーザーの処置: 有効な長さを使用してパラメーターを指定してください。

DES0714N パラメーター *parameter* を直接指定するか、または構成表内に指定してください。

説明: CCSID、形式、または言語が指定されておらず、この値に対する構成設定がありません。

ユーザーの処置: 欠落しているパラメーターを直接 ENABLE TEXT COLUMN コマンドに指定するか、またはテキスト構成設定に値を設定してください。

DES0715N データ・タイプ *schema.type* はテキスト・データにはサポートされていません。

説明: *schema.type* は、テキスト列のスキーマ名およびタイプ名、またはアクセス関数の結果です。テキスト列のデータ・タイプは DB2 テキスト・エクステンダーではサポートされていません。これは、CHAR、GRAPHIC、VARGRAPHIC、LONG VARGRAPHIC、DBCLOB、VARCHAR、LONG VARCHAR または CLOB のいずれかでなければなりません。上記のデータ・タイプ以外の場合は、入力がテキスト列のデータ・タイプで出力が VARCHAR、LONG VARCHAR、または LOB であるアクセス関数を使用しなければなりません。

ユーザーの処置: *schema.type* がテキスト列タイプの場合は、結果がタイプ VARCHAR、LONG VARCHAR、または LOB のアクセス関数を登録しなければなりません。*schema.type* がアクセス関数の結果である場合、これは使用できません。結果が必須タイプのアクセス関数を使用してください。

DES0716N フォーマット *format* はサポートされていません。

説明: *format* は、テキスト・エクステンダーがサポートしていない形式です。

ユーザーの処置: 23 ページの『サポートされる文書形式』のサポートされている形式のリストをチェックしてください。

DES0717N 言語 *language* はサポートされていません。

説明: *language* は DB2 テキスト・エクステンダーがサポートしていない言語であるか、または選択された言語が指定した索引タイプによってサポートされていないかのいずれかです。

ユーザーの処置: 40 ページの表 4 のサポートされている言語のリストをチェックしてください。

DES0718N CCSID *ccsid_value* はサポートされていません。

説明: 無効な CCSID 値が指定されました。

ユーザーの処置: サポートされている CCSID のリストについては、資料を参照してください。

DB2 テキスト・エクステンダーからのメッセージ

DES0719N DB2 テキスト・エクステンダーのプログラム *program* へのコールはリターン・コード *rc* で失敗しました。

説明: インストールの最中にエラーが発生したと考えられます。DES_EXT.H ファイル内に戻りコードがリストされています。

ユーザーの処置: インストールが正常に終了したかどうかをチェックしてください。DB2TX_INSTOWNER や DB2TX_INSTOWNERHOMEDIR などの環境変数が正しく設定されているかどうかを調べてください。

DES0720N アクセス関数 *schema.function* はデータベースに登録されていません。

説明: 関数名が間違っているか、またはデータベースに登録されていません。

ユーザーの処置: アクセス関数の名前をチェックしてください。正しい場合は、その関数がデータベース・システムに認識されているかどうかチェックします。アクセス関数をデータベースに登録するには、CREATE FUNCTION を使用します。

DES0721N データベースが不整合です。DB2 テキスト・エクステンダーカタログ・ビューがありません。

説明: DB2 テキスト・エクステンダー・カタログ・ビューの 1 つがデータベース内に存在しません。

ユーザーの処置: DISABLE DATABASE コマンドで残りのカタログ・ビューを除去してから、再び ENABLE DATABASE コマンドを入力してください。索引データが失われています。テキスト文書に索引を付け直してください。

DES0722N 表 *schema.table* はデータベースの基本表ではありません。

説明: 表がデータベース内に存在しないか、または表が結果表かビューとなっています。テキスト列は、基礎表内になければ DB2 テキスト・エクステンダーで使用できるようになりません。

ユーザーの処置: 表名が正しいかどうか、またそれが基礎表であるか確認してください。

DES0723N 表 *schema.table* のハンドル列 *handlecolumnname* のテキスト索引の (再)作成に失敗しました。

説明: ハンドル列のテキスト索引を作成できませんでした。

ユーザーの処置: txstatus を使ってサーバーの状況を調べてください。サーバー上で保守が正常に行われている場合は、DISABLE TEXT COLUMN または DISABLE TEXT TABLE を使用して、一貫性のある状態に戻してください。その後、ENABLE TEXT COLUMN または ENABLE TEXT TABLE を使用して、テキスト列を再び使用可能にします。

DES0724N 表 *schema.table* のハンドル列 *handlecolumn* のテキスト索引のカタログ・ビューへの入力に誤りです。

説明: TextIndices カタログ・ビューが損傷していません。

ユーザーの処置: DISABLE TEXT COLUMN または DISABLE TEXT TABLE を使用して、一貫性のある状態に戻してください。その後、ENABLE TEXT COLUMN または ENABLE TEXT TABLE を使用してテキスト列を使用可能にします。

DES0725N 要求はサーバーによってリジェクトされました。ライセンス検査は、メッセージ ID '%s1' により失敗しました。

説明: DB2 テキスト・エクステンダーに有効なライセンス・キーが見付かりませんでした。

ユーザーの処置: ライセンス・キーを要求して、インストールしてから再試行してください。

DES0727N 列 *column* は表 *schema.table* で既に使用可能です。

説明: このメッセージは、最初に列を使用不能にしないで、同じテキスト列を使って表を消去してから作成し直した場合に発生することがあります。

ユーザーの処置: 列を使用不能にしてから再試行してください。

DES0728N 列 *column* は表 *schema.table* に存在しません。

説明: 存在しないテキスト列を使用可能にしようとしています。

ユーザーの処置: 表名または列名を変更してから再試行してください。

DES0729N ハンドル列 *handlecolumn* は表 *schema.table* に存在しません。

説明: 存在しないハンドル列を使用しようとしています。

DB2 テキスト・エクステンダーからのメッセージ

ユーザーの処置: GET STATUS コマンドを使用して、ハンドル列が存在するかどうか、さらにその名前が正しく指定されているかチェックしてください。

DES0730N 表 *schema.table* は既に共通索引表として使用可能です。

説明: すでに共通索引表として使用可能になっている表を使用可能にしようとしています。

ユーザーの処置: 表を使用可能にせずに継続するか、または DISABLE TEXT TABLE コマンドを実行して使用不能にしてから再び使用可能にしてください。

DES0731N 表 *schema.table* はテキスト・エクステンダーには使用できず、使用不能にすることもできません。

説明: 使用可能になっていない表を使用不能にしようとしています。

ユーザーの処置: 表名をチェックしてください。

DES0732N ロケーション *location* 付近の更新頻度は誤りです。期待されるのは *parameter* です。

説明: 更新頻度の *parameter* 指定が正しくありませんでした。

ユーザーの処置: 更新頻度パラメーターをチェックし、コマンドを再入力してください。

DES0733N 表 *schema.table* には使用可能な列が含まれています。それを共通索引表として使用可能にすることはできません。

説明: この表には、すでに独自の索引を持っているテキスト列が含まれています。この固有の索引が存在する限り、すべてのテキスト列に共通の索引を作成することはできません。

ユーザーの処置: DISABLE TEXT COLUMN を使用して使用可能な列を使用不能にし、その後再び ENABLE TEXT TABLE コマンドを入力してください。

DES0734N ハンドル列 *handlecolumn* は部分テキスト表 *schema.table* に属しているため、別個に使用不能にすることはできません。

説明: 部分テキスト表として使用可能とされた表内で、単一のテキスト列を使用不能にすることはできません。

ユーザーの処置: 部分テキスト表全体を使用不能にしてください。

DES0736N *handlecolumn* は既に表 *schema.table* のハンドル列です。

説明: 既存のハンドル列名を使用しようとしています。

ユーザーの処置: 別のハンドル列名を使用してコマンドを再入力してください。

DES0737N 表 *schema.tablename* は STORAGE オプション *storage_option* を使用して共通索引表として使用可能です。

説明: 外部ファイルに対して共通索引表を使用可能にすることはできません。

ユーザーの処置: 外部ファイルに対して表を使用可能にしたい場合は、複数索引表を使用してください。

DES0738N アクセス関数 *schema.function* は誤ったパラメーターを持っています。

説明: *schema.function* の入力パラメーターまたは出力パラメーターに誤りがあります。

- 入力パラメーターは 1 つしか使用できず、そのデータ・タイプは使用可能にするテキスト列のデータ・タイプでなければなりません。
- 出力パラメーターのタイプは CHAR、GRAPHIC、VARGRAPHIC、LONG VARGRAPHIC、DBCLOB、VARCHAR、LONG VARCHAR または CLOB のいずれかでなければなりません。

DES0739W 表 *schema.table* ハンドル列 *handlecolumn* の索引の更新プログラムが開始できませんでした。

説明: 索引を更新するプログラムが開始できませんでした。インストールの最中にエラーが発生したと考えられます。

ユーザーの処置: インストールが正常に終了したかどうかをチェックしてください。DB2TX_INSTOWNER や DB2TX_INSTOWNERHOMEDIR などの環境変数が正しく設定されているかどうかを調べてください。

DES0740N ハンドル列 '*column*' はタイプ *columnstype* のテキスト列にのみ、再使用できます。

説明: このハンドル列は、すでに別のハンドル列タイプで使用されています。

ユーザーの処置: 新しいハンドル列名を指定してください。

DES0741N プログラムまたはファイル *parameter* が見つからないか、開始できませんでした。

説明: ENABLE DATABASE または DISABLE DATABASE コマンドでファイル *parameter* をオープンすることができませんでした。インストールの最中にエラーが発生したと考えられます。

ユーザーの処置: インストールが正常に終了したかどうかをチェックしてください。

DES0742N この表は列索引を使用します。コマンドにハンドル列を指定してください。

説明: 指定された表は、複数索引表として使用できません。特定の列を処理するには、関連するハンドル列を指定してください。

ユーザーの処置: ハンドル列を指定してください。

DES0745N DB2TX インスタンスの所有者 *instance-owner* は有効なユーザー ID ではありません。

説明: 環境変数 DB2TX_INSTOWNER に、正しいユーザー ID が含まれていません。

ユーザーの処置: 環境変数を訂正してください。

DES0747N 現行の CCSID は索引タイプ *index_type* にはサポートされていません。

説明: 要求される索引タイプに対してサポートされない CCSID が指定されました。

ユーザーの処置: サポートされている CCSID のリストについては、資料を参照してください。

DES0748N コミット・カウント '*commitcount*' は DB2 テキスト・エクステンダーではサポートされていません。

説明: 指定されたコミット・カウント値はサポートされていません。

ユーザーの処置: 有効なコミット・カウント値を指定してください。

DES0749N 更新索引値 '*indexvalue*' は DB2 テキスト・エクステンダーでは認識されません。

説明: 指定された '更新索引' 値が無効です。

ユーザーの処置: 有効な更新索引値を指定してください。

DES0750N 索引タイプ値 '*indextype*' は DB2 テキスト・エクステンダーでは認識されません。

説明: 指定された索引タイプ値が無効です。

ユーザーの処置: 有効な索引タイプ値を指定してください。

DES0751N 指定された操作を実行する権限がありません。

説明: この操作を実行するために必要なデータベース管理者権限が与えられていません。

ユーザーの処置: この操作はデータベース管理者が行ってください。

DES0756N そのデータベースはテキスト・エクステンダーには使用できません。

説明: このコマンドを実行する前にデータベースを使用可能にしなければなりません。

ユーザーの処置: ENABLE DATABASE を実行してからコマンドを再実行してください。

DES0763N 更新頻度が誤っています。

説明: 指定された '更新頻度' 値が無効です。

ユーザーの処置: 有効な '更新頻度' を指定してください。

DES0765N そのデータベースは、既にテキスト・エクステンダーには使用可能です。

説明: すでに使用可能となっているデータベースを使用可能にしようとしています。

ユーザーの処置: データベースを使用可能とせず継続するか、または DISABLE DATABASE で使用不能にしてからこれを再び使用可能にしてください。

DES0766N 処理によって表または一時表の最大行サイズを超えました。

説明: ENABLE TEXT COLUMN コマンドは、ハンドル列を表に追加します。表がもともと大きい場合は、表の行サイズが最大値 4005 を超える場合があります。

ENABLE TEXT COLUMN コマンドは、すでに使用可能となっているテキスト列の数に釣り合ったサイズの一時表も作成します。多くのテキスト列がすでに使用可能になっている場合は、一時表のサイズが最大値を超えることがあります。

ユーザーの処置: ENABLE TEXT COLUMN は、この制限を超えない表に限り使用してください。

DB2 テキスト・エクステンダーからのメッセージ

DES0769W 警告: 索引の特性が指定されましたが、無視されます。表 *'tablename'* は共通索引表です。

説明: 指定された表は共通索引表なので、索引特性を指定できません。

ユーザーの処置: 何もする必要はありません。

DES0770N 環境変数 *env-variable* が定義されていません。

説明: コマンドにパラメーターが指定されていないため、システムは環境変数 *env-variable* からデフォルト値の値を読み取ろうとしましたが、この環境変数は定義されていませんでした。

ユーザーの処置: 必要な環境変数を定義してください。

DES0774N 変数 *'variable'* の値の長さ *length* は範囲外です。

説明: パラメーターの値の長さが範囲外です。

ユーザーの処置: 有効な長さを使用してパラメーターを指定してください。

DES0775N 索引ディレクトリーの値 *'directory'* は誤りです。

説明: 索引ディレクトリーの値が誤っています。ディレクトリーの長さが誤っている可能性があります。

ユーザーの処置: 有効な索引ディレクトリーの値を指定してください。

DES0776N 表スペース名 *'tablespace'* は誤りです。

説明: 指定された表スペース名が誤っています。表スペースの値の長さが誤っている可能性があります。

ユーザーの処置: 有効な表スペースの値を指定してください。

DES0777N 表スペース *tablespace* はデータベース管理システムには認識されません。

説明: 指定された表スペースはデータベース・システムに認識されていません。

ユーザーの処置: 指定した表スペースがデータベース中に存在することをチェックしてください。

DES0778N *'tablespace'* は正規の表スペースではありません。キーワード *'keyword'* で作成されました。

説明: 指定された表スペースのデータ・タイプはサポートされていません。

ユーザーの処置: 正規の表スペースを指定してください。

DES0779I 索引の作成は正常に開始されました。索引の作成状況をチェックするには **'GET INDEX STATUS'** を使用してください。

説明: 索引付けプログラムが開始しています。'GET INDEX STATUS' コマンドを使用して、索引付けプロセスの状況をチェックできます。

ユーザーの処置: GET INDEX STATUS コマンドの出力をチェックしてください。

DES0780I 索引の再編成は正常に開始されました。索引の状況をチェックするには **'GET INDEX STATUS'** を使用してください。

説明: 再編成プログラムが開始しています。'GET INDEX STATUS' コマンドを使用して、再編成プロセスの状況をチェックできます。

ユーザーの処置: GET INDEX STATUS コマンドの出力をチェックしてください。

DES0789W 警告: 現在のノード・グループの区分化マップが更新されました。TXNCHECK ユーティリティを呼び出してください。

説明: 現行ノード・グループの区分化マップが更新されています。

ユーザーの処置: TXNCHECK コマンドを使用してください。

DES0800I *command* コマンドが正常に完了しました。

説明: 指定されたコマンドは正常に完了しました。

ユーザーの処置: 何もする必要はありません。

DES0810N 閉じの引用符がありません。

説明: 引用符が検出されましたが、後ろの引用符が欠落しています。

ユーザーの処置: コマンド構文をチェックして再試行してください。

DES0811N *"token"* は予期されません。索引特性またはテキスト情報をチェックしてください。

説明: 索引の特性またはテキスト情報に誤りがあります。

ユーザーの処置: 構文をチェックして再試行してください。

DES0812N 表 *schema.table* が存在しないか、または DB2 テキスト・エクステンダーには使用できません。

説明: GET コマンドの実行中に、データベース表の名前が間違っているか存在しない、またはまだ使用可能になっていないかのいずれかです。

ユーザーの処置: 表名が正しい場合は、GET STATUS コマンドを使用してこれが使用可能になっているかチェックしてください。表を使用可能にし、再試行してください。

DES0813N 表 *schema.table* が存在しないか、DB2 テキスト・エクステンダーには使用できないか、またはハンドル列 *column* を含んでいません。

説明: GET コマンドの実行中に、ハンドル列の項目が表内に見つかりません。表が存在する場合は、これが使用可能になっていないか、またはハンドル列が含まれていません。

ユーザーの処置: 表名が正しい場合は、GET STATUS コマンドを使用してこれが使用可能になっているかチェックしてください。表を使用可能にし、再試行してください。

DES0814N 表 *tablename* が存在しないか、DB2 テキスト・エクステンダーでは使用可能でないか、またはこの表内で使用可能なテキスト列がありません。

説明: 指定された表は DB2 テキスト・エクステンダーには使用できません。

ユーザーの処置: 表が DB2 テキスト・エクステンダーで使用できるようにしてください。

DES0815N 空の引用符 *"* が検出されました。名前を引用符の中に入れてください。

説明: 2 つの引用符の間にテキストが入力されていないものが検出されました。

ユーザーの処置: 構文をチェックして再試行してください。

DES0816N ワード *"token"* は予期されません。キーワード *keyword* または *keyword* のいずれかを使用してください。

説明: 予想外のトークンが検出されました。

ユーザーの処置: メッセージに示されているキーワードのいずれかを、コマンド内に指定してください。

DES0817N *"token"* は予期されません。キーワード *keyword* を使用してください。

説明: 予想外のトークンが検出されました。

ユーザーの処置: メッセージに示されているキーワードをコマンド内に指定してください。

DES0818N 予期しないコマンドの終了です。キーワード *keyword* が見込まれます。

説明: キーワードが欠落しています。

ユーザーの処置: メッセージに示されているキーワードをコマンド内に指定してください。

DES0819N 予期しないコマンドの終了です。次のキーワードのいずれかが見込まれます。
keyword または *keyword*。

説明: キーワードが欠落しています。

ユーザーの処置: メッセージに示されているキーワードのいずれかを、コマンド内に指定してください。

DES0820N 索引オプション *index_option* は索引タイプ *index_type* にはサポートされていません。

説明: 要求された索引タイプに対してサポートされない索引オプションが指定されました。

ユーザーの処置: 要求する索引タイプに対してサポートされている索引オプションについて資料を参照してください。

DES0821N 名前 *"token"* が長すぎます。 *variable* の名前に許可されるのは、*nm* 文字だけです。

説明: 名前の長さが指定範囲を超えています。

ユーザーの処置: 有効な長さの名前を指定してください。

DB2 テキスト・エクステンダーからのメッセージ

DES0822N コマンドに認識されないトークン "*token*" が含まれています。コマンドの終了が見込まれます。

説明: コマンドの終わりを検出しましたが、キーワードが必要です。

ユーザーの処置: コマンド構文をチェックして再試行してください。

DES0823N "*schema*" の後に、表名が来なければなりません。

説明: “.” の後に、表名または関数名が欠落しています。

ユーザーの処置: コマンド構文をチェックして再試行してください。

DES0824N 予期しないコマンドの終了です。 *keyword* の値は必須です。

説明: メッセージ内のキーワードが構文から欠落しています。

ユーザーの処置: コマンド構文をチェックして再試行してください。

DES0825N '*alias*' は認識されているデータベースの別名ではありません。許可されるのは *nm* 文字だけです。

説明: 指定されたデータベース別名は、データベース・システムに認識されていません。

ユーザーの処置: 指定した別名が有効であることをチェックしてください。

DES0826N データベース別名 *alias* を引用符で囲んではいけません。

説明: メッセージ内の名前がデータベース別名として解釈されています。これは引用符で囲まないでください。

ユーザーの処置: コマンド構文をチェックして再試行してください。

DES0827N CCSID "*ccsid*" はサポートされていません。

説明: その CCSID は DB2 テキスト・エクステンダーではサポートされていません。

ユーザーの処置: サポートされている CCSID のリストについての資料を参照してください。

DES0829N ユーザー名 *userid* は引用符で囲む必要はありません。

説明: ユーザー名が引用符で囲んで入力されました。

ユーザーの処置: 引用符を取り除いてください。

DES0830N パラメーター "*parameter*" は *enable/disable DATABASE* コマンドで認識されません。コマンドの終了が見込まれます。

説明: コマンド ENABLE DATABASE および DISABLE DATABASE には、パラメーターを指定できません。

ユーザーの処置: パラメーターを指定せずにコマンドを入力し直してください。

DES0831N 予期しないコマンドの終了です。表名がありません。

説明: コマンドには表名が必要です。

ユーザーの処置: 適切な表名を入力してください。

DES0832N 予期しないコマンドの終了です。データベース名がありません。

説明: コマンドにはデータベース名が必要です。

ユーザーの処置: 適切なデータベース名を入力してください。

DES0833N 予期しないコマンドの終了です。列名がありません。

説明: コマンドには列名が必要です。

ユーザーの処置: 適切な列名を入力してください。

DES0899N 未知の DB2TX コマンド: *command*

説明: 指定されたコマンドは、DB2 テキスト・エクステンダーではサポートされていません。

ユーザーの処置: **db2tx ?** と入力して、コマンドのリストを入手してください。

DES0971N 索引ディレクトリーはノード指定なしで一度、ノード指定ありで複数回指定することができます。

説明: 索引ディレクトリーの指定が正しくありません。

ユーザーの処置: 索引ディレクトリーの指定をチェックしてください。ノードを指定せずに 1 つの索引ディレクトリーを指定するか、またはノードを指定して複数の

DB2 テキスト・エクステンダーからのメッセージ

索引ディレクトリーを指定することができます。

DES0972N ノード指定が誤りです。符号なしの数値が要求されています。

説明: ノード番号として数字でない値が指定されています。

ユーザーの処置: ノード番号には符号なしの数値を指定してください。

DES0973N ノード指定が誤りです。右括弧の前に 1 文字以上の予期しない文字が検出されました。

説明: ノードを指定した構文が誤っています。

ユーザーの処置: ノードを指定した構文をチェックして再試行してください。

DES0974N TO 文節のノード番号の順序が誤りです (2 番目のノードが最初のノードより小さい)。

説明: ノードを指定した構文が誤っています。

ユーザーの処置: ノードを指定した構文をチェックして再試行してください。

DES0975N ノード情報の指定構文が誤りです。

説明: ノードを指定した構文が誤っています。

ユーザーの処置: ノードを指定した構文をチェックして再試行してください。

DES0976N ノード情報が不完全です。

説明: ノードの指定が不完全です。一部の情報が欠落しています。

ユーザーの処置: ノードを指定した構文をチェックして再試行してください。

DES0977N ノード情報が不完全です。左括弧がありません。

説明: ノードの指定が不完全です。左括弧が欠落しています。

ユーザーの処置: ノードの指定を訂正して再試行してください。

DES0998N 文書モデル名の長さが正しくありません。

説明: '文書モデル名'値の長さが誤っています。

ユーザーの処置: モデル名の値をチェックして再試行してください。

DES0999N 文書モデルの指定構文が正しくありません。

説明: モデル名を指定した構文が誤っています。

ユーザーの処置: モデルを指定した構文をチェックして再試行してください。

DES5250E テキスト・エクステンダー・インスタンス・ディレクトリーを作成できません。

説明: DB2 テキスト・エクステンダー辞書 /QIBM/userData/DB2Extenders/Text/instance を作成できません。

ユーザーの処置: ディレクトリーを作成するのに必要な権限があることを確認してから、再試行してください。

DES5251I このテキスト・エクステンダー・インスタンスは既にあります。

説明: このインスタンスは、既に存在しているため、作成できません。

ユーザーの処置: 既存のインスタンスを使用するか、インスタンスを除去して再作成してください。

DES5252E テキスト・エクステンダー・インスタンスを除去できません。

説明: インスタンス・ディレクトリー /QIBM/UserData/DB2Extenders/Text/instance が存在しません。インスタンスを除去できません。

ユーザーの処置: TXSTATUS を実行して、インスタンスが既に除去されているかどうかをチェックしてください。インスタンスが除去されている場合は、インスタンスを再作成することができます。

DES5253I テキスト・エクステンダー・インスタンスが存在しません。

説明: 存在しないインスタンスを除去しようとした。

DB2 テキスト・エクステンダーからのメッセージ

DES9994N テキスト検索エンジン・エラーが発生しました。理由コード: **reason-code**

説明: DB2 テキスト・エクステンダーが使用するテキスト検索エンジンにエラーが発生しました。

ユーザーの処置: 241 ページの『第 19 章 検索エンジンの理由コード』を参照して検索エンジンの理由コードをチェックしてください。報告された理由が問題の解決に役立たなかった場合は、`desdiag.log` ファイルまたは作成されたトレース・ファイルに追加情報がある可能性があります。それでも解決できない場合は、利用可能な情報を収集して IBM 技術員に連絡してください。

DES9995N テキスト・エクステンダー・エラーが発生しました。メッセージ・テキスト:
message-text

説明: DB2 テキスト・エクステンダーのエラーが発生しました。

ユーザーの処置: DB2 テキスト・エクステンダーからのメッセージを使用して、問題を解決してください。報告されたメッセージが問題の解決に役立たなかった場合は、`desdiag.log` ファイルまたは作成されたトレース・ファイルに追加情報がある可能性があります。それでも解決できない場合は、利用可能な情報を収集して IBM 技術員に連絡してください。

DES9996N DB2 テキスト・エクステンダーの内部エラーが発生しました。理由コード:
reason_code

説明: 内部処理エラーが発生しました。

ユーザーの処置: DB2 テキスト・エクステンダーのインストールが正常に完了しているかを確認してください。正常に完了している場合は、理由コードを書き留めて、IBM 技術員に連絡してください。

DES9997N SQL エラーが発生しました。 **SqlState:**
state **QL** エラー・コード: *rc*;
SqlErrorMessage: *message*

説明: SQL エラーが発生しました。

ユーザーの処置: メッセージと共に表示される SQL エラー・メッセージに基づいて処置してください。

DES9998N SQL エラーが発生しました。それ以上の情報はありません。

DES9999N 対応するエラー・メッセージがありません。

説明: 内部処理エラーが発生しました。

ユーザーの処置: 診断メッセージをチェックして、問題を解決してください。インストールに関する問題が内部エラーの原因でない場合は、`desdiag.log` ファイルまたは作成されたトレース・ファイルに追加情報がある可能性があります。それでも解決できない場合は、利用可能な情報を収集して IBM 技術員に連絡してください。

第 19 章 検索エンジンの理由コード

この章では、DB2 テキスト・エクステンダーの検索エンジンによって戻される理由コードを記載します。

表 20. 検索エンジンの理由コード

理由コード	値
RC_DONE	0
RC_CONTINUATION_MODE_ENTERED	1
RC_END_OF_INFORMATION	2
RC_EMPTY_LIST	3
RC_MORE_INFORMATION	4
RC_INDEX_GROUP_SEARCH_ERROR	7
RC_INDEX_SPECIFIC_ERROR	8
RC_DICTIONARY_NOT_FOUND	9
RC_PROCESSING_LIMIT_EXCEEDED	12
RC_UNKNOWN_SERVER_NAME	16
RC_INCORRECT_AUTHENTICATION	17
RC_DATASTREAM_SYNTAX_ERROR	18
RC_QUERY_SCOPE_TOO_COMPLEX	20
RC_QUERY_TOO_COMPLEX	22
RC_MEMBER_OF_INDEX_GROUP	23
RC_UNKNOWN_INDEX_NAME	24
RC_INCORRECT_HANDLE	25
RC_INDEX_NOT_MEMBER_OF_GROUP	26
RC_UNKNOWN_SESSION_POINTER	27
RC_UNKNOWN_COMMUNICATION_TYPE	29
RC_UNKNOWN_SERVER_INFORMATION	30
RC_INVALID_MASKING_SYMBOL	31
RC_UNEXPECTED_ERROR	32
RC_SERVER_NOT_AVAILABLE	33
RC_INDEX_ALREADY_OPENED	35
RC_MAX_NUMBER_OF_OPEN_INDEXES	36
RC_MAX_NUMBER_OF_RESULTS	37
RC_CCS_NOT_SUPPORTED	41
RC_LANGUAGE_NOT_SUPPORTED	42
RC_CONFLICT_WITH_INDEX_TYPE	43
RC_MAX_INPUT_SIZE_EXCEEDED	46
RC_SERVER_BUSY	47
RC_SERVER_CONNECTION_LOST	48
RC_SERVER_IN_ERROR	49
RC_REQUEST_IN_PROGRESS	50
RC_UNKNOWN_INDEX_TYPE	51
RC_INCORRECT_INDEX_NAME	52
RC_INCORRECT_LS_EXECUTABLES	53
RC_INCORRECT_LIBRARY_ID	54
RC_INDEX_ALREADY_EXISTS	55
RC_MAX_NUMBER_OF_INDEXES	56
RC_INCORRECT_LOCATION	57
RC_LOCATION_IN_USE	58
RC_UNKNOWN_CONDITION	59

検索エンジンの理由コード

表 20. 検索エンジンの理由コード (続き)

理由コード	値
RC_INDEX_DELETED	60
RC_INDEX_SUSPENDED	61
RC_INDEX_NOT_ACCESSIBLE	62
RC_MAX_NUMBER_OF_BUSY_INDEXES	63
RC_CONFLICTING_TASK_RUNNING	64
RC_NOT_ENOUGH_MEMORY	65
RC_MAX_OUTPUT_SIZE_EXCEEDED	68
RC_COMMUNICATION_PROBLEM	70
RC_NO_ACTION_TAKEN	71
RC_EMPTY_INDEX	72
RC_EMPTY_QUERY	73
RC_INSTALLATION_PROBLEM	74
RC_FUNCTION_DISABLED	75
RC_FUNCTION_IN_ERROR	76
RC_IO_PROBLEM	77
RC_WRITE_TO_DISK_ERROR	78
RC_SERVER_VERSION_NOT_CURRENT	79
RC_FUNCTION_NOT_SUPPORTED	80
RC_RESULT_ALREADY_RANKED	81
RC_RESULT_VIEW_EXISTS	82
RC_INDEX_NOT_OPEN	83
RC_NO_RANKING_DATA_AVAILABLE	84
RC_LINGUISTIC_SERVICE_FAILED	85
RC_THESAURUS_PROBLEM	86
RC_INVALID_IDENTIFIER	88
RC_DOCUMENT_MODEL_ALREADY_EXISTS	89
RC_UNKNOWN_DOCUMENT_SECTION_NAME	90
RC_DOCMOD_READ_PROBLEM	91
RC_UNKNOWN_DOCUMENT_MODEL_NAME	92
RC_SECTION_NAME_ALREADY_EXISTS	94
RC_SECTION_TAG_ALREADY_EXISTS	95
RC_MAX_NUMBER_OF_TASKS	96
RC_LS_NOT_EXECUTABLE	97
RC_LS_FUNCTION_FAILED	98
RC_CAPACITY_LIMIT_EXCEEDED	99
RC_DOCUMENT_NOT_ACCESSIBLE	100
RC_DOCUMENT_CURR_NOT_ACCESSIBLE	101
RC_DOCUMENT_NOT_TO_INDEX	102
RC_DOCUMENT_NOT_FOUND	103
RC_DOCUMENT_IN_ERROR	104
RC_DOCUMENT_NOT_SUPPORTED	105
RC_CROSSIDX_SEARCH_NOT_ALLOWED	110
RC_DOCUMENT_GROUP_NOT_FOUND	111
RC_INVALID_ATTRIBUTE_VALUE	112
RC_INVALID_SECTION_TYPE	113
RC_INCORRECT_RELEVANCE_VALUE	120
RC_NO_RAT_EXPANSION	130
RC_DOCUMENT_NOT_IN_VIEW	131

第 20 章 エラー・イベント理由コード

この章では、DB2 テキスト・エクステンダーが文書を索引付けする際に発生するおそれのあるエラー・イベントを記載します。これは、たとえば、次のような場合に起きることがあります。

- 文書を索引付けすることができない。
- 文書は索引付けされたが、問題が発生した。
- 言語辞書が見つからない。

ヒント

理由コードが文書化されない場合

1. ディスク・スペースに十分空きがあるかチェックしてください。
2. 可能な限りエラー情報を収集してください。
 - desdiag.log ファイル
 - イベント・メッセージ
3. IBM サービス担当員に連絡してください。

1 ストレージが足りません。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。

64 索引プロセスがまだ実行中であるか、索引がまだ再編成処理中です。この理由コードは通知のみを目的としています。

116

データ・ストリーム構文エラー

280

文書は索引付けされていません。索引ファイルの 1 つをオープンできませんでした。

281

文書は索引付けされていません。索引ファイルの 1 つを読み取れませんでした。

441

文書は索引付けされていません。このメッセージが生じるのは、Ngram 索引のみです。文書のコード・ページと索引が作成されたコード・ページとが異なります。これは、索引が UTF8 で作成されていない場合に HTML および XML 文書で生じることがあります。

500

文書は索引付けされていません。ライブラリー・サービスをロードできませんでした。DLL が使用可能であること、およびリソース・パスが有効であることを確認してください。

エラー・イベント理由コード

501

文書は索引付けされていません。ライブラリー・サービス内の Lib_Init はフラット・ファイル・システムで失敗しました。DIT ファイルが見つからないか、有効なディレクトリーにないか、または DIT の内容が正しくありません。

502

文書は索引付けされていません。ライブラリー・サービス LIB_read_doc_content 内の文書の内容の読み取り中に、エラーが発生しました。

503

文書は索引付けされていません。ライブラリー・サービス LIB_access_doc 内でエラーが発生しました。

504

文書は索引付けされていません。ライブラリー・サービス LIB_doc_index_status は、エラーを返しました。

505

文書のクローズに失敗しました。ライブラリー・サービス LIB_close_doc は、エラーを返しました。

506

ライブラリー・サービスの終了に失敗しました。ライブラリー・サービス LIB_end はエラーを返しました。

507

ライブラリー・サービス呼び出し LIB_read_doc_content が、予期しないリターン・コードを戻して失敗しました。

508

ライブラリー・サービス呼び出し LIB_close_doc が、RC_TERMINATION エラーを戻しました。

545

文書は索引付けされていません。一時索引ファイルの 1 つをオープンできませんでした。

546

文書は索引付けされていません。一時索引ファイルの 1 つをクローズできませんでした。

548

内部エラー。診断ログ内の情報を IBM 担当員に送ってください。

549、550

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。

551-564

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であることをチェックしてください。

565

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。

566-587

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。

588-590

内部エラー。診断ログ内の情報を IBM 担当員に送ってください。

591-604

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。

605

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。

606-623

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であるかをチェックしてください。

624

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。

625-631

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であるかをチェックしてください。

632

索引付けで作成された一時ファイルの 1 つを、書き込みアクセスによってオープンできませんでした。アクセス権限を調べてください。

633

索引付けで作成された一時ファイルの 1 つをクローズできませんでした。

634

索引付けで作成された一時ファイルの 1 つに書き込めませんでした。索引作業ディレクトリーに十分なディスク・スペースがあるかを調べてください。

635

索引付けで作成された一時ファイルの 1 つを読み取ることができませんでした。

636

索引付けで作成された一時ファイルの 1 つを、読み取りアクセスによってオープンできませんでした。アクセス権限を調べてください。

659

索引付けで作成された一時ファイルの 1 つをオープンできませんでした。

エラー・イベント理由コード

660

索引付けで作成された一時ファイルの 1 つに書き込めませんでした。

661

索引付けで作成された一時ファイルの 1 つをクローズできませんでした。

662

索引付けで作成された一時ファイルの 1 つをオープンできませんでした。

663

索引付けで作成された一時ファイルの 1 つに書き込めませんでした。

664

索引付けで作成された一時ファイルの 1 つをクローズできませんでした。

665

索引付けで作成された一時ファイルの 1 つをオープンできませんでした。

667

索引付けで作成された一時ファイルの 1 つに書き込めませんでした。

668-669

文書が索引付けされていませんでした。文書内に存在するセクション・タグと文書モデル・ファイル内に定義されているセクション・タグが一致しないという問題が生じています。

670-672

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であるかをチェックしてください。

673

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。

674

内部エラー。診断ログ内の情報を IBM 担当員に送ってください。

675-687

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であるかをチェックしてください。

688、690

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。構成ファイル内の値を小さくしてみてください。

689

内部エラー。診断ログ内の情報を IBM 担当員に送ってください。

691-695

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、

書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であることをチェックしてください。

696-707

内部エラー。診断ログ内の情報を IBM 担当員に送ってください。

708

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。構成ファイル内の値を小さくしてみてください。

709-718

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であることをチェックしてください。

719-721

内部エラー。診断ログ内の情報を IBM 担当員に送ってください。

722、729

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。構成ファイル内の値を小さくしてみてください。

730、732、733、735-738

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であることをチェックしてください。

731、739-742、744-746、749、755-758、760-761、767

内部エラー。診断ログ内の情報を IBM 担当員に送ってください。

743、748、750-754、759、765-766、768-770

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であることをチェックしてください。

747、763、764

ストレージがありません (割り当てに失敗しました)。サーバーのメモリーが十分ではありません。作業負荷を減らしてください。構成ファイル内の値を小さくしてみてください。

815

次の 2 つの原因が考えられます。

- 障害の元となった文書に使用される言語のサポートに必要なリソース・ファイルの 1 つが見つからない
- その文書が要求する言語を DB2 テキスト・エクステンダーがサポートしていない

エラー・イベント理由コード

831

文書は索引付けされていません。テキストが見つかりません。文書の長さが 0 バイトです。

860

ファイル・オープン・エラー。一部の辞書またはシソーラス・ファイルが見つかりません。リソース・パス内に辞書ファイルがあるかを調べてください。検索時にシソーラス・ファイルのパス情報を指定した場合、そのロケーションおよびファイル名を調べてください。

861

文書は索引付けされていません。テキストのトークン化で有効なトークンが検出されませんでした。サポートされている言語および含まれているワードに関して、文書内容の妥当性検査を行ってください。このエラーは、停止ワードのみを含む文書により引き起こされる可能性があります。

954-956

内部エラー。診断ログ内の情報を IBM 担当員に送ってください。

957-967

文書は索引付けされていません。索引ファイルの 1 つのオープン、読み取り、書き込み、またはクローズができませんでした。索引に使用されているディスクのスペースに十分空きがあるか、およびアクセス権限が適正であることをチェックしてください。

1000

ファイルをオープンするときにエラーが発生しました。アクセス権限をチェックしてください。

1001

ファイルを追加するときにエラーが発生しました。アクセス権限をチェックしてください。

1002

ファイルを読み取るときにエラーが発生しました。ファイルが壊れている可能性があります。

1003

ファイルに書き込むときにエラーが発生しました。ディスク・スペースおよびアクセス権限をチェックしてください。

1005

ファイルを読み取るときにエラーが発生しました (ファイル内に位置決めされています)。ファイルが壊れている可能性があります。

1006

一時ファイルの名前を変更するときにエラーが発生しました。アクセス権限を調べてください。

1007

ファイルを作成するときにエラーが発生しました。アクセス権限をチェックしてください。

1008

ファイルを圧縮するときにエラーが発生しました。アクセス権限を調べてください。

1009

ファイルをクローズするときにエラーが発生しました。ファイルが壊れている可能性があります。

1010

指定された索引名は、すでに使用されています。他の索引名を使用してください。

1011

指定されたパスは、すでに使用されています。他の場所を使用してください。

1012

データおよび作業ディレクトリーに同一のパスが使用されています。他の場所を使用してください。

1013

指定された索引名は、無効です。索引名は大文字または数字でなければならず、8文字以下でなければなりません。

1014

ファイルをコピーするときにエラーが発生しました。アクセス権限およびディスク・スペースを調べてください。

1017

索引名が不明です。正しいスペルをチェックしてください。

1019

ファイルを削除するときにエラーが発生しました。アクセス権限をチェックしてください。このエラー・メッセージは「2次エラー」として発生する場合があります。診断ファイルを表示して、直前のエラー項目から追加の情報が得られるかどうかを調べてください。

1020

一般ファイル・エラー。アクセス権限をチェックしてください。

1070-1074

文書は索引付けされていません。指定されたこのコード・ページは、全般に無効であるか、またはアクセス中の索引に関して無効です。

1085

文書は索引付けされていません。索引ページの読み取り中にエラーが発生しました。

1086

文書は索引付けされていません。索引待ち行列が空です。

1116-1117

文書は索引付けされていません。サーバー・インスタンスの初期設定ファイルから得られる情報は処理できません。初期設定ファイル内の項目が有効であること、およびファイルがアプリケーションからアクセス可能であることを確認してください。

1129

どの文書も索引付けされていません。バックグラウンド処理の開始に失敗しました。

エラー・イベント理由コード

1158

ファイルの名前を変更するときにエラーが発生しました。アクセス権およびディスク・スペースを調べてください。

1162

Ngram 索引の索引ファイルが破壊された可能性があります。

1163、1164

文書は予期しないエラーのために、索引付けされていません。

1165

文書は予期しないファイル終わり (EOF) 状態のために、索引付けされていません。

1176

Ngram 索引にこれ以上索引付けをすることはできません。文書数に関してオーバーフロー状態になっています (項目の多いことによるオーバーフロー)。多数の削除を行っていたり、同一の文書を繰り返し更新している場合、EhwReorg を呼び出して問題の解決を試みてください。それ以外の場合は、新規の文書に 2 番目の索引を使用することを検討してください。

1177

文書は索引付けされていません。Ngram インデクサーはその文書が大き過ぎると判断しました。

1189

文書は索引付けされていません。境界シーケンスに問題が発生しました (韓国語に特定)。

1198-1200

文書は索引付けされていません。索引のアクセスに問題が発生しました。索引が破壊された可能性があります。

1201

文書は索引付けされていません。文書のコード・ページを索引に特定のコード・ページに変換することができません。このエラーが発生するのは、UTF8 コード・ページの Ngram 索引のみです。

1202

文書は索引付けされていません。文書内に無効なデータがあるため、文書のコード・ページを索引に特定のコード・ページに変換することができません。このエラーが発生するのは、UTF8 コード・ページの Ngram 索引のみです。

1500-1505

文書分析コンポーネントに問題があります。初期化できないか (LIBPATH および構成ファイルの内容を調べてください)、または内部問題に起因する障害が生じています。詳しくは診断ファイルを参照してください。

1904

文書は索引付けされていません。セクションが使用可能な索引のために、文書モデルにアクセスするときに問題が発生しました。アクセス権およびファイルの存在を調べてください。

2000

文書は索引付けされていません。文書タイプがサポートされていません。ライブラリー・サービス Lib_access_doc は、無効な文書タイプを戻しました。

2001

文書は索引付けされていません。文書のデータ・ストリーム中で、一連の誤りフィールドが検出されました。

2002

文書は索引付けされていません。文書のデータ・ストリーム中で、誤りの構造化フィールドが検出されました。

2003

文書は索引付けされていません。 DB2 テキスト・エクステンダー・テキスト形式の文書に許されるテキスト・セクションは 1 つのみです。

2005

文書は索引付けされていません。文書のデータ・ストリーム中に指定されている言語がサポートされていません。

2006

文書は索引付けされていません。文書のデータ・ストリーム中に指定されている CCSID がサポートされていません。

2007

ライブラリーまたはデフォルト規則が指定した予期された文書形式が正しくありません。文書のヘッダーの形式が正しくありません。デフォルト規則が特殊文書ヘッダーのついた文書であるかどうかをチェックし、その規則が正しくない場合には変更してください。

2008

文書にアクセスできなかったため、索引付けされませんでした。

2009

文書は使用中でアクセスできなかったため、索引付けされませんでした。

2010

文書は索引付けされていません。指定された CCSID が誤りです。

2011

文書が正しい IBM DCA RFT または FFT 文書でないため、索引付けされていません。ページの終わりは、文書の本体テキストの最後の制御ワードでなければなりません。

2012

文書が正しい IBM DCA RFT または FFT 文書でないため、索引付けされていません。構造化フィールドに含まれている長さ指定が誤りです。

2013

文書が正しい IBM DCA RFT または FFT 文書でないため、索引付けされていません。文書中に、誤りの制御ワードが検出されました。

2014

文書が正しい IBM DCA RFT または FFT 文書でないため、索引付けされていません。文書中に、誤りのマルチバイト制御ワードまたは構造化フィールドが検出されました。

2015

文書が正しい IBM DCA RFT または FFT 文書でないため、索引付けされていません。文書のパラメーターが重複して検出されました。

エラー・イベント理由コード

2016

文書が正しい IBM DCA RFT または FFT 文書でないため、索引付けされていません。空のテキスト単位が検出されました。

2018

文書が、サポートされていない形式になっているか、または文書の拡張子について DIT に “exclude” 項目があります。文書に索引付けを許可する拡張子があることをチェックしてください。

2020

文書は索引付けされていません。これは WordPerfect 文書でも WordPerfect ファイルでもありません。

2021

文書は索引付けされていません。これは WordPerfect ファイルですが、WordPerfect 文書ではありません。

2022

文書は索引付けされていません。このバージョンの WordPerfect はサポートされていません。

2023

文書は索引付けされていません。これは暗号化された WordPerfect ファイルです。文書は、暗号化しないで保管してください。

2026

フットノートまたはエンド・ノートで END_TXT が発生しました。WordPerfect ファイルに損傷がある可能性がありますので、チェックしてください。

2028

パーサーは非文書のテキストを戻しました。ファイルの内容、特に形式指定ワードに関する部分を調べてください。文書形式がサポートされているかを調べてください。自動形式認識が失敗した場合、正しいパーサーが呼び出されていることを確認してください。

2030

文書は索引付けされていません。これは、Microsoft Word ファイルでないか、サポートされていないバージョンの Word です。

2031

文書は索引付けされていません。Microsoft Word 文書で、予期しないファイル終わりが検出されました。

2032

文書は索引付けされていません。Microsoft Word 文書で、誤りの制御ワードが検出されました。

2033

文書は索引付けされていません。これは、*fastsave* オプションを指定して、複合形式で保管されています。*fastsave* オプションを除去して保管してください。

2034

文書は索引付けされていません。Microsoft Word 文書で、必要なフィールド終わりマークが欠落しています。

2035

文書は暗号化されています。Microsoft Word で、暗号化なしで保管してください。

2036

これは Word Macintosh 版の文書です。処理できません。Word Windows 版形式で文書を保管してください。

2037

この Word 文書には、組み込み OLE オブジェクトが含まれています。

2040

文書が有効な ECTF ファイルではないため、索引付けされませんでした。

2041

文書は索引付けされていません。文書に含まれている .SO LEN 制御ワードには、数値の指定が続いていません。

2042

文書は索引付けされていません。文書に含まれている .SO LEN 制御ワードには正しくない数値の指定が続いています。この数値は 1 ~ 79 です。

2043

文書は索引付けされていません。 .SO DOC 制御ワードは 1 つしか指定できません。 ECTF 文書はそれぞれ別々のファイルに保管してください。

2044

文書は索引付けされていません。 .SO HDE 制御ワードには、開始タグと終了タグを続けなければなりません。

2046

文書は索引付けされていません。この文書では、.SO DOC 制御ワードの前にテキストが含まれています。

2047

文書は索引付けされていません。この文書では、.SO PID 制御ワードの前にテキストが含まれています。

2048

文書は索引付けされていません。開始タグの後の終了タグが欠落しています。

2050

文書は索引付けされていません。 .SO HDE 制御ワードの後で、誤りのタグが検出されました。

2051

文書は索引付けされていません。 .SO 制御ワードの後で、行の終わりが検出されました。

2052

文書は索引付けされていません。予期しないテキスト終結が検出されました。

2060

文書は索引付けされていません。この文書は AmiPro 文書ではないか、またはサポートされていないバージョンの AmiPro 文書です。

2061

文書は索引付けされていません。AmiPro 文書中の制御ワードが長過ぎます。

エラー・イベント理由コード

2062

文書は索引付けされていません。このバージョンの AmiPro はサポートされていません。サポートされているのは、AmiPro Architecture 第 4 版のみです。

2063

AmiPro 様式用紙が索引付けされていません。

2064

文書は索引付けされていません。誤りの文字セットが検出されました。サポートされるのは Lotus 文字セット 82 (Windows ANSI) のみです。

2065

文書は索引付けされていません。 AmiPro 文書で、予期しないファイル終わりが検出されました。

2072

文書は暗号化されているため、走査できません。

2073

文書形式が不整合です。

2074

文書に “bad file” フラグ・ビット・セットがあります。

2080

文書は索引付けされていません。この文書は RTF 文書でないか、サポートされていないバージョンの RTF 文書です。

2081

文書は索引付けされていません。長過ぎる RTF 制御ワードが検出されました。

2083

文書は索引付けされていません。 Macintosh コード・ページはサポートされていません。

2084

文書は索引付けされていません。この文書は RTF 文書ですが、このバージョンの RTF 文書はサポートされていません。サポートされているのは RTF 第 1 版のみです。

2090

文書は索引付けされていません。この HTML 文書に含まれるタグは、パーサーによって長過ぎると見なされます。

2093

文書は索引付けされていません。この XML 文書は、XML パーサーによって拒否されました。

2100

文書に損傷があるか、または他の理由で読み取れません。新しい共通解析プログラムが、問題を解決できませんでした。

2101

文書は空であるか、テキストが入っていないため、索引付けできません。文書に図形しか含まれていないかどうかをチェックしてください。

2102

文書は、パスワード保護されているか暗号化されているため、索引付けできません。

2105

文書タイプは認識されていますが、フィルターが使用できません。

2106

文書が空のため、索引付けできません。

2107

文書がオープンできないため、索引付けできません。文書アクセスをチェックしてください。

2112

文書は実行可能ファイルであるため、索引付けできません。

2113

文書は圧縮されているため、索引付けできません。

2114

文書は図形であるため、索引付けできません。図形文書形式が受け入れ可能なテキストに戻す場合には、この文書形式を索引付け処理に組み込むように要求を出してください。

2120

ユーザー出口の出力ファイルが存在しないか、またはアクセスできません。新しい共通解析プログラムのバージョンが、問題を解決できませんでした。

2121

出力ファイルはオープンして読み取ることはできません、または空です。新しい共通解析プログラムのバージョンが、問題を解決できませんでした。

2122

ユーザー出口出力ファイルを使用しようとしたが、オブジェクトにファイル名が指定または設定されていません。

2130

ユーザー出口プログラムを実行することはできませんでした。PATH 環境変数により設定されたパスに、実行可能プログラムが見つかるかどうかを確認してください。トレースおよびダンプを作成して、環境 (errno) 戻りコードについての追加情報を入手してください。

2131

戻りコードが正しくなかったために、ユーザー出口プログラムは失敗しました。トレースおよびダンプを作成して、環境 (errno) 戻りコードについての追加情報を入手してください。

第 3 部 付録

特記事項

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品、プログラムまたはサービスの操作性の評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権の許諾については、下記の宛先に書面にてご照会ください。

〒106-0032 東京都港区六本木 3 丁目 2-31
IBM World Trade Asia Corporation
Intellectual Property Law & Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

本書は定期的に見直され、必要な変更 (たとえば、技術的に不適切な表現や誤植など) は、本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Canada Limited

Office of the Lab Director
1150 Eglinton Ave. East
North York, Ontario
M3C 1H7
CANADA

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができませんが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、さまざまなオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのすべての部分、またはすべての派生した創作物には、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年).このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. _年を入れる_. All rights reserved.

商標

以下は、IBM Corporation の商標です。

AIX	IBM
AS/400	iSeries
DB2	
DB2 Extenders	

以下は、それぞれ各社の商標または登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

UNIX は、The Open Group がライセンスしている米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名などはそれぞれ各社の商標または登録商標です。

用語集

この用語集では、本書で使用するさまざまな用語と略語を定義します。探している用語が見つからない場合は、索引か、あるいは「*Dictionary of Computing*」、New York: McGraw-Hill, 1994 を参照してください。

[ア行]

アクセス関数 (access function). 列に保管されるテキストのデータ・タイプを、DB2 テキスト・エクステンダーで処理できるタイプに変換する、ユーザー定義の関数。

アプリケーション・プログラミング・インターフェース (application programming interface (API)). アプリケーション・プログラムと DB2 テキスト・エクステンダー情報検索サービス間の汎用インターフェース。

一致語句索引 (precise index). 抽出されたテキスト文書内にあった用語とまったく同じ用語が入っているテキスト索引。基本語形索引 (*linguistic index*)、および *Ngram* 索引 (*Ngram index*) も参照。

インスタンス (instance). DB2 テキスト・エクステンダーの論理的な環境。同一ワークステーション上に DB2 テキスト・エクステンダーのインスタンスを複数持つことができる。ただし、個々の DB2 インスタンスには 1 つのインスタンスしか持てない。これらのインスタンスを使用すると、次のことが可能である:

開発環境と実稼働環境を分ける

機密情報へのアクセスを特定の人々のみに制限する

インスタンス変数 (instance variable). インスタンス所有者の名前、またはインスタンス所有者のホーム・ディレクトリーの名前のデフォルト値を提供する変数。

エクステンダー (extender). DB2 エクステンダー (*DB2 Extender*) を参照。

エスケープ文字 (escape character). それ以降の文字が、解釈されない マスク文字 (*masking character*) であることを示す文字。

オカレンス (occurrence). 突き合わせ、一致 (*match*) の同義語。

[カ行]

外部ファイル (external file). DB2 に制御される表中のセルの形式ではなく、オペレーティング・システムのファイル・システムに保管されるファイル形式のテキスト文書。

カウント (count). シソーラスの中の用語のレベル数 (深さ) を指定するのに使用されるキーワード。このレベルは、指定された関係に基づく検索項目を拡張するために使用される。

拡張する (expand). シソーラスから導出された用語を検索項目に追加すること。

拡張マッチング (extended matching). 検索項目と完全には一致していない用語を強調表示する辞書を使用するプロセス。

カタログ・ビュー (catalog view). 管理目的のために DB2 テキスト・エクステンダーが作成するシステム表のビュー。カタログ・ビューには、DB2 テキスト・エクステンダーが使用できる表と列に関する情報が含まれる。

環境プロファイル (environment profile). DB2 テキスト・エクステンダーの提供するスクリプト。環境変数の設定を含む。

環境変数 (environment variable). DB2 テキスト・エクステンダー環境の値に対するデフォルトを提供するために使用する変数。

関数 (function). アクセス関数 (*access function*) を参照。

基本語形索引 (linguistic index). 言語的な処理により基本語形になった用語を含むテキスト索引。たとえば “Mice” (ねずみの複数形) は、“mouse” (単数形) として索引付けされる。一致語句索引 (*precise index*)、および *Ngram* 索引 (*Ngram index*) も参照。

境界検索 (bound search). 韓国語文書における、ワード境界を区別した検索。

共通索引表 (common-index table). テキスト列が共通のテキスト索引を共有する DB2 表。複数索引表 (*multi-index table*) を参照。

形式 (format). 文書のタイプ、たとえば ASCII や WordPerfect。

言語 (language). 索引付け、検索、およびブラウズの際に使用されるディクショナリー (*dictionary*) の名前。

検索する (retrieve). DB2 テキスト・エクステンダーの検索関数の 1 つに検索引き数を使用してテキスト文書を検出すること。

検索引き数 (search argument). 検索の際に指定される条件で、1 つまたは複数の検索項目と検索パラメーターから成る。

コード・ページ (code page). すべてのコード・ポイントに図形文字および制御機能の意味を割り当てたもの。たとえば、8 ビット・コードの場合、文字および意味を 256 のコード・ポイントへ割り当てたもの。

更新頻度 (update frequency). テキスト索引が更新される頻度。日付、時間、分、および索引付けを行う前にログ表 にリストする必要がある文書名の最小数で表される。

コマンド行プロセッサ (command line processor). DB2TX というプログラムで、次のような機能がある:

DB2 テキスト・エクステンダー・コマンドを入力できるようにする。

コマンドを処理する

結果を表示する

混成検索 (hybrid search). ブール検索 とフリー・テキスト検索 との組み合わせ。

[サ行]

索引タイプ (index type). テキスト索引 の特性で、その索引が文書内の用語を実際の形で含むか、それとも言語処理した形で含むかを決定する。一致語句索引 (*precise index*)、基本語形索引 (*linguistic index*)、および Ngram 索引 (*Ngram index*) も参照。

索引付け (index). テキストから有効な用語を抽出し、テキスト索引 に保管すること。

索引特性 (index characteristics). 以下のことを判別するテキスト索引 の特性。

索引を保管するディレクトリー

索引のタイプ

索引を更新する頻度

索引を最初に更新する時点

シェル・プロファイル (shell profile). 環境プロファイル (*environment profile*) を参照。

辞書 (dictionary). DB2 テキスト・エクステンダーがテキストの分析、索引付け、検索を行ったり、特定言語の文書を強調表示したりするのに使用する、言語関連の情報のコレクション。

周期的索引付け (periodic indexing). 日付、時間、分で表される既定の時間間隔、および索引付けを行う前にそのログ表 にリストする必要がある文書名の最小数を指定した索引付け。

使用可能にする (enable). サブシステム、テキスト表、またはテキスト列を DB2 テキスト・エクステンダーで使用できるように準備すること。

詳細化 (refine). 前の検索の検索基準を後の検索基準に追加して、一致 数を減らすこと。

使用不可にする (disable). 使用可能プロセス中に作成された項目を除去することにより、サブシステム、テキスト表、またはテキスト列を DB2 テキスト・エクステンダーで使用可能になる前の条件に復元すること。

情報の強調表示 (highlighting information). データ・ストリーム (*data stream*) を参照。

初期化ハンドル (initialized handle). 前もって準備され、テキスト形式かテキスト言語 (またはその両方) だけを含むハンドル。

[タ行]

突き合わせ、一致 (match). テキスト文書内の検索項目のオカレンス。

データ・ストリーム (data stream). API 機能から戻される情報で、(少なくとも 1 つの段落からなる) テキストで構成される。テキストには検索対象の用語、および見付かった用語をそのテキスト内で強調表示するための情報が入っている。

停止ワード (stop word). テキスト文書 (*text document*) において、テキスト索引 (*text index*) から除外され、検索引き数 (*search argument*) に含まれていても無視される一般的な語 (たとえば “before”)。

テキスト構成 (text configuration). 索引、テキスト、および処理の値のデフォルト設定。

テキスト索引 (text index). テキスト文書から抽出された、重要な用語の集まり。それぞれの用語は、抽出元の文書と関連付けられる。文書そのものではなくこの索引を検索することによって、検索時間を大幅に改善する

ことができる。一致語句索引 (*precise index*)、および 基本語形索引 (*linguistic index*) も参照。

テキスト情報 (text information). 以下のことを記述するテキスト文書 の特性。

CCSID

形式

言語

テキスト表 (text table). テキスト列 を含む DB2 表。

テキスト文書 (text document). DB2 表の中に保管される、タイプ CHAR、GRAPHIC、VARGRAPHIC、LONG VARGRAPHIC、DBCLOB、VARCHAR、LONG VARCHAR、または CLOB のテキスト。

テキスト列 (text column). テキスト文書 を含んでいる列。

特殊タイプ (distinct type). ユーザー定義特殊タイプ (*user-defined distinct type*) を参照。

トリガー (trigger). テキスト列から文書が追加、変更、削除されるたびに、ログ表 への索引付けに必要な文書に関する情報を自動的に追加する機構。

トレース処理 (tracing). エラーの原因を追及するために後で使用される情報をファイルに保管する処置。

[ナ行]

ノード (node). 区分データベース 環境のサーバー。論理ノード (*logical node*)、物理ノード (*physical node*)、およびノード・グループ (*nodegroup*) も参照。

ノード・グループ (nodegroup). 1 つまたは複数のデータベース区画サーバーのサブセット。物理的に別個のマシンに割り当てられるノード。論理ノード (*logical node*) も参照。

[ハ行]

ハンドル (handle). テキスト文書を識別する 2 進値。次のものを含む:

文書 ID

関連する索引の名前と場所

その文書のテキスト情報

DB2 が制御しない外部ファイルに保管されている文書の場合、そのファイルのパスと名前。

ハンドルは、テキスト列が DB2 テキスト・エクステンダーで使用可能な場合に、そのテキスト列内のテキスト文書ごとに作成される。

ブール検索 (Boolean search). 1 つまたは複数の言葉をブール演算子を使って結合する検索。

ファイル・ハンドル (file handle). ハンドル (*handle*) を参照。

ファジー検索 (fuzzy search). 検索項目と似たつづりの語を検出する検索。

複数索引表 (multi-index table). テキスト列が個別のテキスト索引を持つ DB2 表。共通索引表 (*common-index table*) を参照。

物理ノード (physical node). 物理的に別個のマシンに割り当てられるノード。論理ノード (*logical node*) も参照。

ブラウザー (browser). コンピューターのモニター上にテキストを表示できるようにする DB2 テキスト・エクステンダー機能。

ブラウズする (browse). コンピューター・モニター上にテキストを表示すること。

フリー・テキスト検索 (free-text search). 検索項目がフリー・フォームのテキストとして表される検索 (フリー・フォーム・テキストとは、検索対象の事柄を自然言語で記述した句または文)。

プロファイル (profile). 環境プロファイル (*environment profile*) を参照。

文書 (document). テキスト文書 (*text document*) を参照。

文書ハンドル (document handle). ハンドル (*handle*) を参照。

文書モデル (document model). 文書が含むセクションに関する、文書の構造の定義。DB2 テキスト・エクステンダーは索引付けの際に、文書モデルにより文書中のセクションを認識する。文書モデルでは、各セクションを識別するマークアップ・タグをリストする。セクション照会に使用するためのセクション記述名を、それぞれのタグごとに指定できる。1 つの文書モデル・ファイルには、1 つまたは複数の文書モデルを指定できる。

[マ行]

マスキング文字 (masking character). 検索項目の前面、中央部、および後面で任意指定文字を表すのに使用

する文字。マスキング文字は通常、一致語句索引でさまざまな用語を検索するのに使用される。

[ヤ行]

ユーザー定義関数 (user-defined function、UDF).

DB2 のユーザーが作成する SQL 関数。DB2 に備えられている SQL 関数と対比される。DB2 テキスト・エクステンダーに備えられている、CONTAINS などの管理および検索関数は、UDF の書式である。

ユーザー定義特殊タイプ (user-defined distinct type、UDT). DB2 のユーザーが作成するデータ・タイプ。LONG VARCHAR など、DB2 に備えられているデータ・タイプと対比される。

[ラ行]

ランク (rank). タイプ DOUBLE の 0 と 1 の間の絶対値で、ある文書が、見つかった他の文書と比べてどの程度検索基準に合致しているかを示す。この値は文書内で検出された一致数と、その文書サイズを示す。

ログ表 (log table). テキスト文書の索引付けに関する情報を含む、DB2 テキスト・エクステンダーが作成する表。使用可能なテキスト列内の文書が追加されるたびに、この情報をログ表に保管するには、トリガーを使用する。

論理ノード (logical node). 同じ物理マシンに他のノードと共に割り当てられるノード。物理ノード (*physical node*) も参照。

[ワ行]

ワイルドカード文字 (wildcard character). マスキング文字 (*masking character*) を参照。

A

API. アプリケーション・プログラミング・インターフェース (Application programming interface)。

C

CCSID. コード化文字セット ID (Coded Character Set Identifier)。

D

DB2 エクステンダー (DB2 Extender). 従来の数値データや文字データ以外のデータ・タイプ (たとえばイメ

ージ、オーディオ、ビデオ・データ) および複雑な構造の文書の保管や検索を可能にする、複数のプログラムからなるグループ。

DBCS. 2 バイト文字サポート (Double-byte character support)。

N

Ngram 索引 (Ngram index). テキスト索引 (*text index*) の 1 つで、DBCS 文書をサポートし、SBCS 文書のあいまい検索をサポートする。基本語形索引 (*linguistic index*)、および一致語句索引 (*precise index*) も参照。

S

SBCS. 1 バイト文字サポート (Single-byte character support)。

U

UDF. ユーザー定義関数 (User-defined function)。

UDT. ユーザー定義特殊タイプ (user-defined distinct type)。

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

アクセス関数
説明 52
ENABLE TEXT COLUMN 中の 115
アプリケーション・プログラミング・インターフェース (API)
一致情報の入手 95
解説 175
概説 91
概要 175
強調表示情報 182
検索関数 91
検索結果表の入手 93
情報強調表示のためにポインターを手する 182
テキストの検索 93
ブラウズ関数 91
ブラウズ情報ストレージの解放 96
ブラウズ情報の入手 94
ブラウズする文書のオープン 94
ブラウズ・セッションの開始 94
ブラウズ・セッションの終了 96
プログラムの例 195
文書のクローズ 95
メッセージ 227
戻りコード 221
DesCloseDocument 関数 176
DesEndBrowseSession 関数 177
DesFreeBrowseInfo 関数 178
DesGetBrowseInfo 関数 179
DesGetMatches 関数 182
DesGetSearchResultTable 関数 187
DesOpenDocument 関数 191
DesStartBrowseSession 関数 193
一致語句検索 33
一致語句索引
検索オプションのデフォルト 170
説明 33
イベント理由コード 243
インスタンス
作成 11
除去 11
インストール検査 9

エクステンダー
使用例 3
エスケープ文字
構文 173
使用法 65
エラー・イベント
記録 50
削除 80
表示 84
理由コード 243
DELETE INDEX EVENTS 107
GET INDEX STATUS コマンド 129
オープン、文書の、API 関数
使用法 94
説明 191
オカレンス、検索項目の 161

[カ行]

開始、ブラウズ・セッションの、API 関数
使用法 94
説明 193
開始、DB2 テキスト・エクステンダー・サーバーの 141
外部ファイル
索引更新の考慮事項 78
使用可能化 53
使用不可にする 89
ハンドル 61
ハンドルからのパス / 名前の抽出 73
ハンドル内でのパス / 名前の変更 74
ハンドル内のファイル名を入手または変更する 157
DISABLE TEXT FILES コマンド 110
ENABLE TEXT FILES コマンド 120
FILE 関数 157
解放、ブラウズ情報用のストレージの、API 関数
使用法 96
説明 178
拡張、強調表示のための用語 204
拡張マッチング 205
カタログ・ビュー
削除 90
作成 45
内容 86
各国語サポート 28
環境、クライアント/サーバー 4
関数
解説 153

関数 (続き)
概説 154
関数パス 61
検索関数 57
検索パフォーマンスの向上 75
検索引き数の指定 63
説明 57
直前の検索の詳細化 70
データ・タイプの変換用 52
テキストの検索 61
により戻される SQL の状態 227
ハンドル内の情報の設定と抽出 72
API 関数
参照: アプリケーション・プログラミング・インターフェース (API)
CCSID 155
CONTAINS 156
DB2 テキスト・エクステンダー関数のパスの設定 61
FILE 157
FORMAT 158
LANGUAGE 159
NO_OF_DOCUMENTS 160
NUMBER_OF_MATCHES 161
RANK 162
REFINE 163
SEARCH_RESULT 164
SET CURRENT FUNCTION PATH ステートメント 61
管理
外部ファイルの索引の更新 78
クライアント・コマンドの要約 99
コマンド行プロセッサ 99
サーバー状況の表示 12
サーバーを使用不可にする 90
サーバー・コマンドの要約 137
索引イベントの削除 80
索引状況のリセット 79
索引状況の表示 82
索引設定の表示 85
索引設定の変更 79
索引の再編成 80
索引の即時更新 78
作成、DB2 テキスト・エクステンダー・インスタンスの 11
サンプル・データベース、作成用ユーティリティ 9
サンプル・データベースの作成 152
シソーラス定義ファイルのコンパイル 144

管理 (続き)

状況情報の入手 81
省略語ファイル、編集 41
データベース、表、および列の状況の表示 81
停止ワードと省略語ファイルの変更 41
停止ワード・ファイル、変更 41
テキスト構成の変更 11
テキスト索引の保守 77
テキスト情報設定の表示 85
テキスト表を使用不可にする 89
テキスト列を使用不可にする 88
テキスト・ファイルを使用不可にする 89
トレース障害 13
バックアップおよび復元 12
CHANGE INDEX SETTINGS コマンド 101
CHANGE TEXT CONFIGURATION コマンド 103
CONNECT コマンド 106
DB2 テキスト・エクステンダー・インスタンスの除去 11
DB2 テキスト・エクステンダー・サーバーの開始 141
DB2 テキスト・エクステンダー・サーバーの停止 143
DB2TX コマンド 99
DELETE INDEX EVENTS 107
DELETE INDEX EVENTS コマンド 107
DISABLE SERVER FOR DB2TEXT コマンド 108
DISABLE TEXT COLUMN コマンド 109
DISABLE TEXT FILES コマンド 110
DISABLE TEXT TABLE コマンド 111
GET INDEX SETTINGS コマンド 127
GET INDEX STATUS コマンド 129
GET STATUS コマンド 130
GET TEXT CONFIGURATION コマンド 131
GET TEXT INFO コマンド 132
IMOTHESC コマンド 144
IMOTHESN コマンド 146
IMOTRACE コマンド 148
Ngram シソーラス定義ファイルのコンパイル 146
QUIT コマンド 133
REORGANIZE INDEX コマンド 134
RESET INDEX STATUS コマンド 135
TXICRT コマンド 138

管理 (続き)

TXIDROP コマンド 139
TXSAMPLE コマンド 140
TXSTART コマンド 141
TXSTATUS コマンド 142
TXSTOP コマンド 143
TXVERIFY コマンド 152
UPDATE INDEX コマンド 136
規則、検索引き数の 173
基本形への変形、用語の 202
基本テキスト分析
強調表示 204
正規化 198
非英数字文字を含む用語 198
文の認識 199
用語の索引付け 198
境界検索、例 69
強調表示情報
データ・ストリーム 95
データ・ストリーム構文 182
共通索引表
作成 46
説明 34, 35
ENABLE TEXT TABLE コマンド 123
組み込みファイル des_ext.h 91
クライアント/サーバー環境 4
クローズ、文書の、API 関数
使用法 95
説明 176
形式、テキスト文書の 23
サポートされていないものを変換 24
サポートされるもののリスト 23
説明 23
テキスト構成内のデフォルト 10
ハンドルからの抽出 73
ハンドル内での変更 74
CHANGE TEXT CONFIGURATION 中の 105
ENABLE TEXT COLUMN 中の 115
FORMAT 関数 158
FORMAT キーワード 105, 115
GET TEXT INFO コマンド 132
結果表 188
権限
管理者権限 14
ユーザー権限 14
言語、テキスト文書の
検索引き数 67
説明 26
テキスト構成内のデフォルト 10
ハンドルからの抽出 73
ハンドル内での変更 74
リスト 26
GET TEXT INFO コマンド 132
LANGUAGE 関数 159

言語索引

検索オプションのデフォルト 170
説明 31
言語的な処理
拡張マッチング 205
基本テキスト分析 198
検索のための 202
語のマスキング 204
索引作成時 197
説明 197
停止ワードのフィルター操作 202
同義語 203
発音の拡張 204
ブラウズ 204
マスキング 204
文字のマスキング 204
用語拡張 204
用語の基本形への変形 202
言語パラメーター、リスト 40
検索、言語的な処理 202
検索、テキストの
概説 61
検出された一致数の入手 63
検出文書のランクの入手 63
構文 166
照会の作成 62
直前の検索の詳細化 70
パフォーマンスの向上 75
プログラムの例 195
API の使用 93
REFINE 関数 163
SEARCH_RESULT 関数 164
検索エンジンからの理由コード 241
検索エンジンの理由コード 241
検索関数 91
検索状況の表示
構文 129
例および出力 82
検索状況のリセット
構文 135
例 79
検索引き数
同じ段落中の用語の検索 66
同じ文中の用語の検索 66
規則と制約事項の要約 173
境界検索 69
構文 166
混成検索 70
さまざまな言語での用語の検索 67
シソーラスの検索 69
指定 63
説明 165
属性値 168
同義語の検索 67
任意の順序での用語の検索 66
発音の似た語の検索 69

検索引き数 (続き)

ファジー検索 68, 170
複数の用語の検索 63
フリー・テキスト検索 70
文書セクションでの用語の検索 66
マスキング文字の使用 65
用語の一部の検索 65
用語の変化形の検索 64
AND キーワード 170
BOUND キーワード 170
COUNT キーワード 168
EXPAND キーワード 169
FUZZY FORM OF キーワード 170
IN SAME PARAGRAPH AS 169
IN SAME SENTENCE AS 170
MODEL キーワード 167
NOT キーワード 169
NOT を使用した検索 68
PRECISE FORM OF キーワード 170
RESULT LIMIT キーワード 169
SECTION キーワード 167
STEMMED FORM OF キーワード 170
SYNONYM FORM OF キーワード 170
TERM OF キーワード 169
THESAURUS キーワード 168
& (AND) 演算子 169
& および! を使用した検索 64
| (OR) 演算子 169
更新、テキスト索引の
定期的に行う 35
頻度の変更 79
CHANGE INDEX SETTINGS 中の
UPDATEFREQ 101
CHANGE TEXT CONFIGURATION 中
の UPDATEFREQ 104
更新状況の表示
構文 129
例および出力 82
更新状況のリセット
構文 135
例 79
更新頻度
構文 36
説明 35
変更 79
CHANGE INDEX SETTINGS 中の
UPDATEFREQ 101
CHANGE TEXT CONFIGURATION 中
の UPDATEFREQ 104
GET INDEX SETTINGS コマンド
127
構成 10
構成表
作成 45

構成表 (続き)

表示 82
CHANGE TEXT CONFIGURATION コ
マンド 103
GET TEXT CONFIGURATION コマン
ド 131
構成ファイル 217
構造、文書の
検索構文中の MODEL キーワード
167
検索構文中の SECTION キーワード
167
検索構文中の属性値 168
検索例 66
セクション・サポートの使用可能化
37
語の分割 33
コマンド
クライアント・コマンドの要約 99
サーバー・コマンドの要約 137
CHANGE INDEX SETTINGS 101
CHANGE TEXT
CONFIGURATION 103
CONNECT 106
DB2TX 99
DELETE INDEX EVENTS 107
DISABLE SERVER FOR
DB2TEXT 108
DISABLE TEXT COLUMN 109
DISABLE TEXT FILES 110
DISABLE TEXT TABLE 111
ENABLE SERVER FOR
DB2TEXT 112
ENABLE TEXT COLUMN 113
ENABLE TEXT FILES 120
ENABLE TEXT TABLE 123
GET INDEX SETTINGS 127
GET INDEX STATUS 129
GET STATUS 130
GET TEXT CONFIGURATION 131
GET TEXT INFO 132
IMOTHESC 144
IMOTHESN コマンド 146
IMOTRACE 148
QUIT 133
REORGANIZE INDEX 134
RESET INDEX STATUS 135
TXICRT 138
TXIDROP 139
TXSAMPLE 140
TXSTART 141
TXSTATUS 142
TXSTOP 143
TXVERIFY 152
UPDATE INDEX 136

コマンド行プロセッサ

開始 44
ヘルプ 45
DB2TX コマンド 99
QUIT コマンド 133
混成検索、例 70
コンパイル、シソーラス定義ファイルの
144
コンパイル、Ngram シソーラス定義ファ
イルの 146

[サ行]

サーバー

開始 141
使用可能化 45
状況情報の表示 81
状況の表示 142
使用不可にする 90
設定および保守 11
停止 143
トレース障害 13
バックアップおよび復元 12
CONNECT コマンド 106
DISABLE SERVER FOR DB2TEXT コ
マンド 108
ENABLE SERVER FOR DB2TEXT コ
マンド 112
GET STATUS コマンド 130
IMOTRACE コマンド 148
TXICRT コマンド 138
TXIDROP コマンド 139
TXSAMPLE コマンド 140
TXSTART コマンド 141
TXSTATUS コマンド 142
TXSTOP コマンド 143
TXVERIFY コマンド 152

索引

一致語句 33
概説 21
外部ファイルの更新 78
基本語形 31
共通索引表 34
計画 21
現行設定値の表示 85
現行設定値の変更 79
更新頻度 35
更新頻度の変更 79
サイズの計算 35
再編成 80
索引タイプの変更 34
作成、テキスト列に対するさまざまな
索引タイプ 51
周期的な索引更新 35
即時の索引更新 78
タイプ 31

索引 (続き)

テキスト構成設定内のデフォルト・タイプ 10
テキスト構成の変更 11
バックアップおよび復元 12
複数、使用 34
保守 77
CASE_ENABLED オプション 33
CHANGE INDEX SETTINGS コマンド 101
CHANGE TEXT CONFIGURATION コマンド 103
CHANGE TEXT CONFIGURATION 中の INDEXTYPE 103
CHANGE TEXT CONFIGURATION 中の TABLESPACE 104
ENABLE TEXT COLUMN 中の INDEXOPTION 117
ENABLE TEXT COLUMN 中の INDEXTYPE 116
ENABLE TEXT TABLE 中の INDEXOPTION 124
ENABLE TEXT TABLE 中の INDEXTYPE 124
GET INDEX SETTINGS コマンド 127
GET INDEX STATUS コマンド 129
GET TEXT CONFIGURATION コマンド 131
Ngram 33
UPDATE INDEX コマンド 136
索引状況の表示
構文 129
例および出力 82
索引状況のリセット
構文 135
例 79
索引タイプの変更
作成、テキスト列に対するさまざまな索引タイプ 51
変更 34
索引付け、言語的な処理 197
索引付けイベント
理由コード 243
索引付けイベントの削除
構文 107
例 80
索引特性
テキスト構成設定内のデフォルト 10
表示 85
ENABLE TEXT COLUMN 中の 113
ENABLE TEXT FILES 中の 120
ENABLE TEXT TABLE 中の 123
作成、サンプル表の
TXSAMPLE コマンド 140
作成、サンプル・データベースの 152

作成、DB2 テキスト・エクステンダー・インスタンスの
TXICRT コマンド 138
サンプル API プログラム 195
サンプル表
削除 90
説明 57
TXSAMPLE コマンド 140
サンプル・データベース
作成 (TXVERIFY) 9
辞書ファイル名 40
シソーラスの検索
概念 206
構文 168
シソーラス定義ファイルのコンパイル 144
シソーラスの作成 210
例 69
IMOTHESC コマンド 144
IMOTHESN コマンド 146
Ngram シソーラス定義ファイルのコンパイル 146
Ngram シソーラスの作成 213
THESAURUS キーワード 168
終了、ブラウザ・セッションの、API 関数
使用法 96
説明 177
障害の検出 13
状況、索引の
エラー後のリセット 79
現在状況の表示 129
表示 82
リセット 79
状況、DB2 テキスト・エクステンダー・サーバーの 142
情報、テキスト文書に関する
形式 23
言語 26
現行設定の表示 85
タイプ 23
CCSID 26
GET TEXT INFO コマンド 132
省略語
省略語ファイルの編集 41
リスト 40
除去、インスタンスの
方法 11
TXIDROP コマンド 139
処理特性
テキスト構成設定内のデフォルト 10
スペース所要量、索引の 35
正規化、用語の 198
制約事項、検索引き数の 173
セキュリティー 14

セクション、文書中の
検索構文中の MODEL キーワード 167
検索構文中の SECTION キーワード 167
検索構文中の属性値 168
検索例 66
セクション・サポートの使用可能化 37
フラット・ファイル文書 37
文書モデル・ファイル、内容 37
DESMODEL.INI 37
HTML 文書 37
XML 文書 38
接続、データベースへの
CONNECT コマンド 106

[タ行]

タイプ、テキスト索引の
一致語句 33
基本語形 31
検索オプションのデフォルト 170
テキスト構成設定内のデフォルト 10
CASE_ENABLED オプション 33
CHANGE TEXT CONFIGURATION コマンド 103
CHANGE TEXT CONFIGURATION 中の INDEXTYPE 103
ENABLE TEXT COLUMN 中の INDEXTYPE 116
ENABLE TEXT TABLE 中の INDEXTYPE 124
GET INDEX SETTINGS コマンド 127
Ngram 33
直前の検索の詳細化 70
突き合わせ、一致
検索結果 63
DesGetMatches 関数 182
DesGetSearchResultTable 93
NUMBER_OF_MATCHES 関数 161
データ・ストリーム構文 183
データ・タイプ、テキスト文書の
サポートされる 115
変換用の関数 52
停止、DB2 テキスト・エクステンダー・サーバーの 143
停止ワード
基本テキスト分析の一部として 202
説明 21
停止ワード・ファイルの編集 41
リスト 40
ディスク・スペース、索引の 35

ディレクトリー、索引用の
GET INDEX SETTINGS コマンド
127
テキスト構成設定値
インストール時のデフォルト 10
表示 82
変更 11
テキスト特性
形式 23
言語 26
テキスト構成内のデフォルト 10
CCSID 26
ENABLE TEXT COLUMN 中の 113
ENABLE TEXT FILES 中の 120
テキスト表
バックアップおよび復元 12
ラージ表における列の使用可能化 51
DISABLE TEXT TABLE コマンド
111
ENABLE TEXT TABLE コマンド
123
テキスト・エクステンダー・プロファイル
9
同義語
検索引き数 67
説明 203
SYNONYM FORM OF キーワード
170
特殊タイプ 153
トリガー
説明 22
トレース障害
セットアップ 13
IMOTRACE コマンド 148

[ナ行]

入手、一致語の、API 関数
使用法 95
説明 182
入手、検索結果表の、API 関数
使用法 93
説明 187
入手、ブラウズ情報の、API 関数
使用法 94
説明 179
認識、文の 199
ノード
ノード・グループおよび表スペース
50

[ハ行]

はじめに 19

発音の拡張
説明 204
例 69
バックアップおよび復元 12
パフォーマンスの向上 75
ハンドル
外部ファイルの 61
形式と言語の変更 74
情報の設定と抽出 72
説明 60
特殊タイプ DB2TEXTFH 153
特殊タイプ DB2TEXTH 153
リストを使用したパフォーマンスの向
上 75
CCSID 関数 155
CCSID、形式、および言語の抽出 73
FORMAT 関数 158
LANGUAGE 関数 159
ハンドル・リストのポインター (特殊タイ
プ DB2TEXTFHLISTP) 154
ハンドル・リストのポインター (特殊タイ
プ DB2TEXTHLISTP) 154
表スペースおよびノード・グループ 50
ブール演算子
NOT 68
& (AND) および ! (OR) 64
ブール検索引き数 169
ファジー検索 31
ファジー検索、例 68
深さを指定する、シソーラスに含まれる用
語の 168
複数索引の使用 34
ブラウズ
言語的な処理 204
独自のブラウザーの使用 94
プログラムの例 195
ブラウズ関数 91
フラットな ASCII、文書形式 23
フラット・ファイル文書、セクション・サ
ポート 37
フリー・テキスト検索
例 70
文書
関連情報 85
形式、説明 23
形式の変換 24
言語 26
構造 37
索引付け 21
サポートされる形式 23
サポートされるデータ・タイプ 115
設定値の表示 85
データ・タイプの変換 52
CCSID 26
CHANGE TEXT CONFIGURATION 中
の形式 105

文書 (続き)
ENABLE TEXT COLUMN 中の形式
115
GET TEXT INFO コマンド 132
文書モデル
検索構文中の MODEL キーワード
167
検索構文中の SECTION キーワード
167
検索構文中の属性値 168
説明 37
文書モデル・ファイルの変更 37
文書モデル・ファイル、内容 37
分析、テキストの
索引付け 198
ブラウズ 204
文の認識 199
文の分割 33
ヘッダー・ファイル des_ext.h 91
ヘルプ、コマンドの 45

[マ行]

マスキング
言語的な処理 204
検索項目内での 65
マッチング、拡張 205
メッセージ 227
文字のマスキング 204
戻りコード 221

[ヤ行]

ユーザー出口、文書形式の変換 24
用語拡張、強調表示のための 204
用語の基本形への変形 202
呼び出し、DB2 テキスト・エクステンダ
ー・プログラムの 6
AS400 固有の 6
QSHELL 6

[ラ行]

ラージ表、使用可能化 51
ランク
検索結果 63
DesGetSearchResultTable 93
列
さまざまな索引タイプの使用可能化
51
使用可能化 49
使用不可にする 88
ラージ表における使用可能化 51
DISABLE TEXT COLUMN コマンド
109

列 (続き)

ENABLE TEXT COLUMN コマンド
113

ログ表

エラー・イベントの抽出 84

説明 22

表スペースへの割り当て 50

ログ用スペース、使い尽くす 51

[ワ行]

ワイルドカード文字

検索項目内での 65

語のマスキング 204

A

AmiPro、文書形式 23

AND

検索引き数内のキーワード 170

ブール演算子 64

ASCII、文書形式 23

AS/400 オペレーション・ナビゲーター

77

B

BOUND キーワード 170

C

CASE_ENABLED キーワード

ENABLE TEXT COLUMN 中の 117

ENABLE TEXT TABLE 中の 124

CCSID

関数 155

コード・ページの問題の回避 28

説明 26

テキスト構成内のデフォルト 10

ハンドルからの抽出 73

リスト 26

CHANGE TEXT CONFIGURATION 中
の 104

ENABLE TEXT COLUMN 中の 115

GET TEXT INFO コマンド 132

CHANGE INDEX SETTINGS コマンド

構文 101

使用法 79

CHANGE TEXT CONFIGURATION コマ
ンド

構文 103

使用法 11

COMMITCOUNT 構成パラメーター

説明 51

テキスト構成設定内のデフォルト 11

COMMITCOUNT 構成パラメーター (続
き)

ログ用スペースの確保 52

CHANGE TEXT CONFIGURATION 中
の 104

ENABLE TEXT COLUMN 中の 119

ENABLE TEXT TABLE 中の 136

CONNECT コマンド

構文 106

CONTAINS 関数

構文 156

例 62

COUNT キーワード 168

D

DB2 Extenders

使用例 3

DB2 テキスト・エクステンダーの概説 3

DB2 テキスト・エクステンダーの関数に

より戻される SQL の状態 227

DB2 テキスト・エクステンダーのサンプ
ル関数

実行 57

DB2TEXTFH 特殊タイプ 153

DB2TEXTFHLISTP 特殊タイプ 154

DB2TEXTH 特殊タイプ 153

DB2TEXTHLISTP 特殊タイプ 154

DB2TX、コマンド行プロセッサ

構文 99

使用法 44

DB2TX.SAMPLE 表

削除 90

作成用ユーティリティ 9

説明 57

DBCS 文書、検索 33

DELETE INDEX EVENTS コマンド

構文 107

例 80

DesCloseDocument 関数

使用法 95

説明 176

DESCL.INI 217

DesEndBrowseSession 関数

使用法 96

説明 177

DesFreeBrowseInfo 関数

使用法 96

説明 178

DesGetBrowseInfo 関数

使用法 94

説明 179

DesGetMatches 関数

使用法 95

説明 182

DesGetSearchResultTable 関数

使用法 93

説明 187

DESMODEL.INI 37

DesOpenDocument 関数

使用法 94

説明 191

DESRESTB、結果表の作成 188

DESSAMP1、サンプル・プログラム 195

DESSRV.INI 218

DesStartBrowseSession 関数

使用法 94

説明 193

DES_BROWSE、DesGetSearchResultTable

中のオプション 188

DES_EXTENDED、DesOpenDocument 中

のオプション 191

DES_EXT.H ヘッダー・ファイル 91

DES_FAST、DesOpenDocument 中のオプ
ション 191

DES_MATCH、DesGetSearchResultTable

中のオプション 188

DES_NOBROWSE、

DesGetSearchResultTable 中のオプション

188

DES_RANKANDMATCH、

DesGetSearchResultTable 中のオプション

188

DES_RANK、DesGetSearchResultTable 中

のオプション 188

DES_TEXTHANDLEONLY、

DesGetSearchResultTable 中のオプション

188

DIRECTORY キーワード

現行設定の表示 85

テキスト構成設定内のデフォルト 10

CHANGE TEXT CONFIGURATION 中

の 104

ENABLE TEXT COLUMN 中の 118

ENABLE TEXT TABLE 中の 125

DISABLE SERVER FOR DB2TEXT コマ
ンド

構文 108

使用法 90

DISABLE TEXT COLUMN コマンド

構文 109

使用法 88

DISABLE TEXT FILES コマンド

構文 110

使用法 89

DISABLE TEXT TABLE コマンド

構文 111

使用法 89

E

- ENABLE SERVER FOR DB2TEXT コマンド
 - 構文 112
 - 使用法 45
- ENABLE TEXT COLUMN コマンド
 - 構文 113
 - 使用法 49
- ENABLE TEXT FILES コマンド
 - 構文 120
 - 使用法 53
- ENABLE TEXT TABLE コマンド
 - 構文 123
 - 使用法 46
- EXPAND キーワード 169

F

- FFT、文書形式 23
- FILE 関数
 - 構文 157
 - 例 73
- FORMAT 関数
 - 構文 158
 - 例 73
- FUNCTION キーワード
 - 説明 52
 - ENABLE TEXT COLUMN 中の 115
- FUZZY FORM OF キーワード 170

G

- GET INDEX SETTINGS コマンド
 - 構文 127
 - 例および出力 85
- GET INDEX STATUS コマンド
 - 構文 129
 - 例および出力 82
- GET STATUS コマンド
 - 構文 130
 - 例および出力 81
- GET TEXT CONFIGURATION コマンド
 - 構文 131
 - 例および出力 82
- GET TEXT INFO コマンド
 - 構文 132
 - 例および出力 85

H

- HANDLE 関数
 - 使用法 75
- HANDLE_LIST 関数
 - 使用法 75

- HTML 構造化文書 37
- HTML 文書、セクション・サポート 37
- HTML、文書形式 23

I

- IMOTHESC コマンド
 - 構文 144
- IMOTHESN コマンド
 - 構文 146
- IMOTRACE
 - 構文 148
 - 使用法 13
- IN SAME PARAGRAPH AS キーワード 169
- IN SAME SENTENCE AS キーワード 170
- INDEXOPTION キーワード
 - CHANGE TEXT CONFIGURATION 中の 104
 - ENABLE TEXT COLUMN 中の 117
 - ENABLE TEXT TABLE 中の 124
- INDEXPROPERTY キーワード
 - ENABLE TEXT COLUMN 中の 117
 - ENABLE TEXT TABLE 中の 124
- INDEXTYPE キーワード
 - CHANGE TEXT CONFIGURATION 中の 103
 - ENABLE TEXT COLUMN 中の 116
 - ENABLE TEXT TABLE 中の 124

L

- LANGUAGE 関数
 - 構文 159
 - 例 73
- LANGUAGE キーワード 105, 115
- LOGPRIMARY、LOGSECOND、および LOGFILSIZ パラメーター、DB2 UDB for iSeries 内の 51

M

- Microsoft、文書形式 23

N

- Ngram 索引
 - 検索オプションのデフォルト 170
 - 説明 33
 - CASE_ENABLED オプション 33
- NODE キーワード
 - ENABLE TEXT COLUMN 中の 118
 - ENABLE TEXT TABLE 中の 125

- NORMALIZED キーワード
 - ENABLE TEXT COLUMN 中の 117
- NOT
 - 検索引き数内のキーワード 169
 - ブール演算子 68
- NO_OF_DOCUMENTS 関数
 - 構文 160
- NUMBER_OF_MATCHES 関数、構文 161

O

- ON NODE キーワード
 - ENABLE TEXT COLUMN 中の 118
 - ENABLE TEXT TABLE 中の 125
- OR ブール演算子 64

P

- PRECISE FORM OF キーワード 170

Q

- QDESADM 14
- QDESUSR 14
- QUIT コマンド
 - 構文 133
 - 使用法 55

R

- RANK 関数
 - 構文 162
 - 例 63
- REFINE 関数
 - 構文 163
 - 例 70
- REORGANIZE INDEX コマンド
 - 構文 134
 - 例 80
- RESET INDEX STATUS コマンド
 - 構文 135
 - 例 79
- RESULT LIMIT キーワード 169

S

- SEARCH_RESULT 関数
 - 構文 164
 - 例 75
- SET CURRENT FUNCTION PATH ステートメント 61
- SQL 表の準備 14
- STEMMED FORM OF キーワード 170

T

- TERM OF キーワード 169
- TEXTINDEXES カタログ・ビュー
 - 削除 90
 - 作成 45
 - 内容 86
- TXICRT コマンド
 - 構文 138
 - 作成、DB2 テキスト・エクステンダー・インスタンスの 11
- TXIDROP コマンド
 - 構文 139
- TXPROFILE 9
- TXSAMPLE コマンド
 - 構文 140
 - 使用法 9
- TXSAMPLE.UDF
 - 実行 57
- TXSTART コマンド
 - 構文 141
 - 使用法 12
- TXSTATUS コマンド
 - 構文 142
 - 使用法 12
- TXSTOP コマンド
 - 構文 143
 - 使用法 12
- TXVERIFY
 - サンプル・データベースの作成 9
- TXVERIFY コマンド
 - 構文 152

U

- UDT 153
- UPDATE INDEX コマンド
 - 構文 136
 - 例 78
- UPDATEFREQ キーワード
 - CHANGE INDEX SETTINGS 中の 101
 - CHANGE TEXT CONFIGURATION 中の 104
 - ENABLE TEXT COLUMN 中の 118
 - ENABLE TEXT TABLE 中の 125
- UPDATEINDEX キーワード
 - 現行設定の表示 85
 - テキスト構成設定内のデフォルト 10
 - CHANGE TEXT CONFIGURATION 中の 104
 - ENABLE TEXT COLUMN 中の 118
 - GET INDEX SETTINGS コマンド 127

W

- WordPerfect、文書形式 23

X

- XML 構造化文書 37
- XML 文書、セクション・サポート 38
- XML、文書形式 23

[特殊文字]

- & (AND) 演算子、検索引き数内の検索引き数の構文 169
 - 使用方法 64
- | (OR) 演算子、検索引き数内の検索引き数の構文 169
 - 使用方法 64



プログラム番号: 5722-DE1

Printed in Japan

SH88-8521-01



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12