

Замечание

Прежде чем использовать эту информацию и описанный в ней продукт, прочтите общие сведения в разделе Приложение Е, “Замечания”, на стр. 205.

Замечание по изданию

Информация в этом документе является собственностью IBM. Она предоставляется в соответствии с лицензионным соглашением и защищена законами об авторском праве. Информация, содержащаяся в этой публикации, не содержит никаких гарантий, и никакое утверждение в этом руководстве не должно рассматриваться как гарантия.

Заказать публикации IBM можно через Интернет или через вашего местного представителя IBM.

- Чтобы заказать публикации через Интернет, откройте страницу IBM Publications Center по адресу www.ibm.com/shop/publications/order
- Чтобы найти вашего местного представителя IBM откройте страницу IBM Directory of Worldwide Contacts по адресу www.ibm.com/planetwide

Чтобы заказать публикации DB2 publications в отделе DB2 Marketing and Sales в Соединенных Штатах или Канаде, позвоните по телефону 1-800-IBM-4YOU (426-4968).

Посылая информацию IBM, вы даете IBM неисключительное право использовать или распространять эту информацию тем способом, каким компания сочтет нужным, без каких-либо обязательств перед вами.

Содержание

Глава 1. Текстовый поиск DB2 (DB2 Text Search) 1

Глава 2. Обзор текстового поиска DB2 3

Основные возможности и понятия	3
Сценарий: Индексирование и поиск	3
Создание, обновление и изменение индекса текстового поиска.	6
Лингвистическая обработка для DB2 Text Search.	7
Инструменты командной строки для DB2 Text Search.	9
Вызов команд текстового поиска	9
Поддержка документов в формате расширенного текста	10
Планирование решений текстового поиска	10
Особенности защиты для DB2 Text Search	10
Роли пользователей.	11
Планирование мощности DB2 Text Search	12
Форматы документов, поддерживаемые для DB2 Text Search.	24
Поддерживаемые типы данных	24
Преобразование неподдерживаемых форматов и типов данных	25
Поддерживаемые языки и кодовые страницы	25
О размерах документов	26

Глава 3. Установка и конфигурирование DB2 Text Search. . 27

Установка и конфигурирование текстового поиска DB2 при помощи мастера по установке DB2	28
Установка серверов DB2 (Windows).	29
Установка серверов DB2 при помощи мастера по установке DB2 (Linux и UNIX)	32
Установка и конфигурирование DB2 Text Search при помощи файла ответов.	36
Установка продукта DB2 при помощи файла ответов (Windows)	37
Установка продукта баз данных DB2 при помощи файла ответов (Linux и UNIX)	38
Установка текстового поиска DB2 при помощи db2_install (Linux и UNIX)	39
Установка продукта баз данных DB2 при помощи команды db2_install или doce_install (Linux и UNIX)	39
Конфигурирование текстового поиска DB2	42
Конфигурирование DB2 Text Search при помощи инструмента конфигурирования.	43
Обновление файла служб на сервере для соединений TCP/IP	45

Глава 4. Обновление текстового поиска. 47

Обновление DB2 Text Search для установок от имени администратора или пользователя root	47
--	----

Обновление установки DB2 Text Search от имени рядового пользователя (не root) (Linux и UNIX).	48
---	----

Глава 5. Администрирование и обслуживание индексов текстового поиска. 51

Запуск службы экземпляра текстового поиска DB2.	51
Включение базы данных для DB2 Text Search	51
Установка DB2 Accessories Suite для DB2 Text Search Версии 9.7 FP3 или более ранних версий	52
Установка DB2 Accessories Suite для DB2 Text Search Версии 9.7 FP4 или более новых версий	54
Деинсталляция DB2 Accessories Suite для DB2 Text Search	55
Конфигурирование DB2 Text Search для поддержки документов в расширенном текстовом формате	56
Включение поддержки документов в расширенном текстовом формате для DB2 Text Search	58
Остановка службы экземпляра текстового поиска DB2	59
Выключение базы данных для DB2 Text Search	59
Удаление бесхозных собраний текстового поиска	60
Выключение поддержки документов в расширенном текстовом формате для DB2 Text Search	61
Создание индексов текстового поиска	62
Создание индекса текстового поиска	63
Повышение производительности индексов текстового поиска	66
Обслуживание индексов текстового поиска	66
Обновление индекса текстового поиска	67
Очистка событий индекса текстового поиска.	68
Изменение индекса текстового поиска	69
Просмотр состояния индексов текстового поиска	69
Отбрасывание индекса текстового поиска.	70
Резервное копирование и восстановление индексов текстового поиска	71
Словари синонимов для DB2 Text Search	71
Добавление словаря синонимов для DB2 Text Search	72
Удаление словаря синонимов для DB2 Text Search	73

Глава 6. Поиск с использованием индексов текстового поиска. 75

Функции поиска для DB2 Text Search	75
Планирование запросов DB2 Text Search	77
Аргументы текстового поиска DB2 (DB2 Text Search)	77
Множественные предикаты DB2 Text Search	78
Функция SCORE DB2 Text Search	78
Функция RESULTLIMIT DB2 Text Search	79
Поиск в индексах текстового поиска	79
Поиск в индексах текстового поиска при помощи SCORE	80
Повышение производительности для полнотекстовых запросов	81

Синтаксис аргументов текстового поиска	81
Поиск документов XML при помощи DB2 Text Search	85
Конфигурирование поиска XML для DB2 Text Search	87
Синтаксис поиска для документов XML	88

Глава 7. Встроенные функции поиска SQL и XML 93

функция CONTAINS	93
функция SCORE	95
Функция xmlcolumn-contains	98

Глава 8. Команды управления для DB2 Text Search 103

db2ts START FOR TEXT,	104
db2ts STOP FOR TEXT	104
db2ts CLEANUP FOR TEXT,	105
db2ts ENABLE DATABASE FOR TEXT	106
db2ts DISABLE DATABASE FOR TEXT	107
db2ts CLEAR COMMAND LOCKS	109
db2ts CREATE INDEX,	111
db2ts DROP INDEX,	117
db2ts ALTER INDEX,	119
db2ts UPDATE INDEX,	123
db2ts CLEAR EVENTS FOR INDEX,	127
db2ts HELP	128

Глава 9. Подпрограммы DB2 Text Search 131

Хранимая процедура SYSTS_ADMIN_CMD - Запуск команд управления текстовым поиском	131
Процедура SYSTS_ENABLE - Включение текущей базы данных для текстового поиска	132
Процедура SYSTS_DISABLE - Выключение текущей базы данных для текстового поиска	134
Процедура SYSTS_CREATE - Создание индекса текстового поиска для столбца	136
Процедура SYSTS_DROP - Отбрасывание индекса текстового поиска	144
Процедура SYSTS_ALTER - Изменение характеристик обновления индекса	146
Процедура SYSTS_UPDATE - Обновление индекса текстового поиска	151
Процедура SYSTS_CLEAR_EVENTS - Удаление событий индексации из таблицы событий индекса	154
Процедура SYSTS_CLEAR_COMMANDLOCKS - Удаление блокировок команд для индексов текстового поиска	156

Глава 10. Управляющие производные таблицы текстового поиска 159

Производная таблица SYSIBMTS.TSDEFAULTS	159
---	-----

Производная таблица SYSIBMTS.TSLOCKS	160
SYSIBMTS.TSCOLLECTIONNAMES view	160
Производная таблица SYSIBMTS.TSINDEXES	161
Производная таблица SYSIBMTS.TSCONFIGURATION	162
Производная таблица SYSIBMTS.TSEVENT	162
Производная таблица SYSIBMTS.TSSTAGING	163

Глава 11. Инструменты текстового поиска 165

Инструмент конфигурирования для DB2 Text Search	165
Инструмент управления для DB2 Text Search	167
Инструменты синонимов для DB2 Text Search	169
Инструмент форматирования журналов для DB2 Text Search	170

Приложение А. Сравнение DB2 Text Search и Net Search Extender 173

Перенастройка в DB2 Text Search	175
---	-----

Приложение В. Локали, поддерживаемые DB2 Text Search. . . 177

Приложение С. Команды DB2. 179

db2iupgrade - обновление экземпляра	179
db2icrt - Создание экземпляра	182
db2idrop - удаление экземпляра	186
db2iupdt - Обновление экземпляров	188

Приложение D. Обзор технической информации DB2 193

Техническая библиотека DB2 в печатном виде или в формате PDF	193
Заказ печатных копий книг DB2	196
Вызов справки по SQLSTATE из командной строки	197
Доступ к различным версиям Информационного центра DB2	197
Вывод тем Информационного центра DB2 на предпочитаемом вами языке	198
Обновление Информационного центра DB2, установленного на вашем компьютере или на сервере интранет	199
Обновление Информационного центра DB2, установленного на вашем компьютере или на сервере интранет, вручную	200
Учебные материалы по DB2.	202
Информация об устранении неисправностей DB2	202
Положения и условия	203

Приложение Е. Замечания 205

Индекс 209

Глава 1. Текстовый поиск DB2 (DB2 Text Search)

DB2 Text Search позволяет применять операторы SQL и XQuery для выполнения запросов текстового поиска данных, хранящихся в базе данных DB2.

DB2 Text Search предоставляет большие возможности для поиска данных в текстовых столбцах, хранящихся в таблице DB2. Поисковая система обеспечивает быстрый отклик на запрос и консолидированный и ранжированный набор результатов, что позволяет вам легко и без промедления найти необходимую информацию. При помощи включения функций DB2 Text Search в состав ваших операторов SQL и XQuery можно создать мощные и многоцелевые программы поиска текстов.

DB2 Text Search достигает высокой производительности и масштабируемости путем использования потокового режима передачи данных, что исключает высокие ресурсные затраты во время поиска.

DB2 Text Search поддерживается на следующих платформах:

- Платформы AIX (64-битные)
- Платформы Linux на x64 (64-битные)
- Платформы HP-UX для HP Integrity Series на основе Itanium (64-битные)
- Платформы Solaris для AMD (64-битные)
- Платформы Solaris для UltraSPARC (64-битные)
- Платформы Windows на x86 (32-битные)
- Платформы Windows на x64 (64-битные)
- Платформы Linux на x86 (32-битные) поддерживаются только для целей разработки и тестирования.

Примечание: DB2 Text Search не поддерживает Microsoft Failover Clustering.

DB2 Text Search включает в себя следующие основные возможности:

Тесную интеграцию с DB2 Database для Linux, UNIX и Windows

- Интерфейс хранимых процедур для команд администрирования
- Выполняемые установщиком DB2 установку и конфигурирование
- Скрытая аутентификация
- Коды SQL для обработки ошибок

Индексацию документов

- Быструю индексацию больших объемов данных
- Поддержку pureXML
- Поддержку многих форматов документов
- Инкрементное и асинхронное изменение индексов

Технологию расширенного поиска

- Поддержку SQL, SQL/XML и XQuery
- Функции SQL CONTAINS и SCORE
- Встроенные функциональные возможности SQL в сочетании с DB2 Optimizer
- Функцию XML xmlcolumn-contains

- Фильтрацию XML
- Лингвистическую обработку на всех поддерживаемых языках
- Поддержку весовых коэффициентов, символов подстановки и необязательных термов
- Поддержку словаря синонимов

Требования аппаратного обеспечения для DB2 Text Search следующие:

Требования к аппаратному обеспечению

- Два двухъядерных процессора по 2,66 ГГц
- 4 Гбайта оперативной памяти
- Скрытая аутентификация
- Достаточно дискового пространства для хранения исходного текста в четырехкратном объеме, умноженного на число индексов текстового поиска. Например, если сервер текстового поиска поддерживает пять индексов с 1 миллионом строк по 1 Кбайту текста в каждой, требуется минимальное дисковое пространство 20 Гбайт.

Глава 2. Обзор текстового поиска DB2

DB2 Text Search позволяет выполнять поиск в текстовых столбцах таблицы DB2. Включив поддержку текстового поиска, можно использовать функции CONTAINS, SCORE и xmlcolumn-contains, встроенные в механизм DB2, для поиска по индексам текстового поиска на основе заданных аргументов поиска.

DB2 Text Search работает путем сбора данных из различных источников и их индексирования для последующего быстрого извлечения. Кроме того, механизм поиска использует лингвистический анализ, чтобы обеспечить возврат только подходящих результатов запроса.

Как показано на рис. 1, сервер DB2 Text Search и серверы DB2 устанавливаются в одной системе, но сервер DB2 Text Search запускает собственную виртуальную Java-машину (JVM). Службы DB2 Text Search запускаются и останавливаются явным образом после запуска экземпляра DB2. Можно установить DB2 Text Search и DB2 Net Search Extender в той же копии установки DB2.

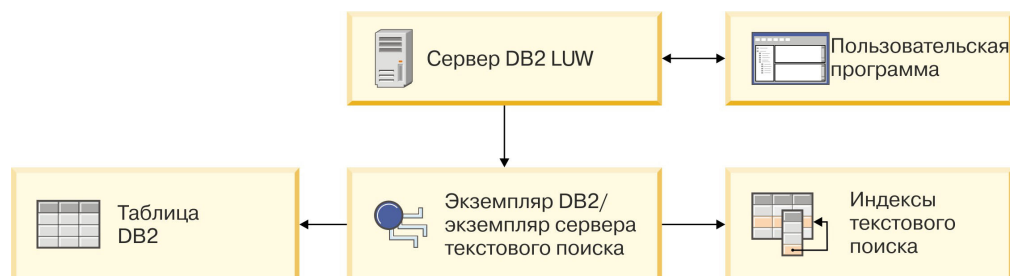


Рисунок 1. Схема внедрения для сервера текстового поиска DB2

Начиная с версии 9.7 с пакетом Fix Pack 3 и более новыми пакетами исправлений, для одного столбца таблицы могут одновременно существовать текстовые индексы DB2 Text Search и Net Search Extender. Теперь можно переключаться между текстовыми индексами DB2 Text Search и Net Search Extender, просто изменяя состояние индекса.

У DB2 Text Search нет графического пользовательского интерфейса. Вместо этого для таких задач, как конфигурирование и администрирование сервера DB2 Text Search, создание словаря синонимов для собрания и диагностика проблем, доступны инструменты командной строки. Кроме того, можно использовать интерфейс хранимых процедур для некоторых распространенных задач администрирования.

Основные возможности и понятия

Сценарий: Индексирование и поиск

После установки и конфигурирования DB2 Text Search надо выполнить четыре действия, прежде чем приступить к поиску.

1. Запустите службы экземпляра DB2 Text Search.

Процесс, поддерживающий расписание изменений индексов текстового поиска, запускается при вводе команды **db2ts START FOR TEXT**. Служба экземпляра текстового поиска автоматически обновляет индекс текстового поиска в заданные моменты времени.

2. Подготовьте базу данных, которую будет использовать DB2 Text Search.

Базу данных нужно включить для DB2 Text Search только один раз. Обратите внимание на то, что нельзя включить базу данных для Net Search Extender, если она уже включена для DB2 Text Search.

3. Создайте индекс текстового поиска для столбца, который содержит или будет содержать текст для поиска.
4. Заполните индекс текстового поиска. При этом данные добавляются в пустой, вновь созданный индекс текстового поиска.

Когда индекс текстового поиска будет содержать данные, можно проводить поиск индекса при помощи оператора SQL, а если индекс содержит данные XML - при помощи XQuery.

Как показано на рис. 2, вам нужно вручную или автоматически изменить существующие индексы текстового поиска, чтобы учесть изменения в текстовом столбце, с которым связан индекс.



Рисунок 2. Конфигурирование индексов тестового поиска для поиска

Начиная с Версии 9.7 Fix Pack 3 и более новых пакетов Fix Pack, текстовые индексы DB2 Text Search и Net Search Extender могут одновременно существовать для одного столбца таблицы. Теперь текстовый индекс DB2 Text Search можно создать для того же столбца, для которого уже существует индекс Net Search Extender. Вы можете использовать хранимую процедуру SYSPROC.SYSTS_ALTER для переключения состояния текстового индекса с ACTIVE на INACTIVE или наоборот, тем самым переключаясь между текстовыми индексами DB2 Text Search и Net Search Extender.

Тип текстового индекса определяется на уровне запросов. Это подразумевает, что все текстовые индексы в предикатах одного запроса должны быть одинакового типа, то

есть или типа DB2 Text Search, или типа Net Search Extender. Если несколько предикатов в одном запросе используют текстовые индексы, некоторые из которых принадлежат к типу индексов DB2 Text Search, а другие - к Net Search Extender, такой поиск окончится неудачно с сообщением об ошибке SQL20425N.

Примечание: Переключение текстового индекса должно сопровождаться переключением прикладной программы, чтобы активировать соответствующий синтаксис запросов.

Простые сценарии

1. Предположим, что вы хотите сделать продукты в таблице PRODUCT базы данных SAMPLE доступными для поиска DB2 Text Search. В предположении, что вы уже создали базу данных sample (запустив команду **db2samp1**) и задали для переменной среды **DB2DBDFT** значение SAMPLE, можно ввести следующие команды:

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

Теперь имена продуктов и описания, содержащиеся в столбце NAME таблицы PRODUCT, проиндексированы и доступны для поиска. Если вы хотите найти ID продуктов всех снеговых лопат (snow shovels), введите следующий запрос поиска:

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

2. Включите базу данных Net Search Extender для DB2 Text Search и используйте возможность совместного существования индексов для запроса в базе данных.

Запустите базу данных для текстового поиска.

```
db2ts start for text
DB20000I Команда SQL выполнена успешно.
```

Включите базу данных Net Search Extender для DB2 Text Search.

```
db2ts enable database for text
C1E00001 Операция завершена успешно
```

Создайте и измените индекс DB2 Text Search для столбца, у которого есть существующий индекс Net Search Extender.

```
db2ts "CREATE INDEX db2ts.title_idx FOR TEXT ON books(title)"
C1E00001 Операция завершена успешно.
```

```
db2ts "UPDATE INDEX db2ts.title_idx FOR TEXT"
C1E00001 Операция завершена успешно.
```

Активируйте новый индекс DB2 Text Search.

```
db2 "ALTER INDEX db2ts.title_idx FOR TEXT SET ACTIVE"
C1E00001 Операция завершена успешно.
```

Введите запрос для использования индекса DB2 Text Search.

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

```
ISBN          TITLE
123-014014014 Climber's Mountain Tops
```

Выбрано 2 записи.

Введите запрос, который ссылается на оба индекса в таблице, DB2 Text Search и Net Search Extender. У столбца title есть активный индекс DB2 Text Search, и одновременно у столбца bookinfo - активный индекс Net Search Extender. Поиск возвратит ошибку, так как все текстовые индексы в одном запросе должны быть одного типа.

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

```
ISBN          TITLE
```

SQL20425N Столбец "BOOKINFO" в таблице "BOOKS" был задан как аргумент для функции текстового поиска, но для этого столбца не существует индекса текстового поиска.
SQLSTATE=38H12

Чтобы избежать такой ошибки, создайте индекс DB2 Text Search для столбца bookinfo и активируйте его.

```
db2ts "CREATE INDEX db2ts.bookinfo_idx FOR TEXT ON books( bookinfo )"
CIE00001 Операция завершена успешно.
```

```
db2ts alter index db2ts.bookinfo_idx for text set active
CIE00001 Операция завершена успешно.
```

Создание, обновление и изменение индекса текстового поиска

Создание индекса текстового поиска - это процесс определения и объявления свойств индекса. После создания индекса текстового поиска можно обновить его, добавив новые данные из связанной с ним таблицы. Можно также изменить свойства индекса текстового поиска.

С помощью DB2 Text Search можно создать индекс текстового поиска, чтобы данные текстового столбца можно было искать с помощью функций поиска. Индекс текстового поиска состоит из значимых термов, которые извлекаются из текстовых документов. Первичный ключ строки используется для уникальной идентификации текстового документа в этой строке. При создании индекса текстового поиска для него можно задать различные свойства, в том числе частоту обновления и каталог для хранения.

Непосредственно после создания индекс текстового поиска не содержит данных. Данные в индекс текстового поиска добавляются с помощью специальной для текстового поиска команды db2ts **UPDATE INDEX** или административной подпрограммы SQL SYSTS_UPDATE. При первом обновлении индекса все текстовые документы добавляются из текстового столбца в индекс. Это обновление обычно называют *начальным обновлением*. Все последующие обновления называют *инкрементными обновлениями*.

Индексы текстового поиска реорганизуются автоматически по мере необходимости; нельзя явным образом запустить реорганизацию или пересоздать индекс при обновлении. Однако можно отбросить индекс текстового поиска, создать новый и выполнить для него начальное обновление.

Можно изменить некоторые свойства индекса текстового поиска после его создания, например частоту обновлений или минимальное число изменений, необходимое для запуска обновления.

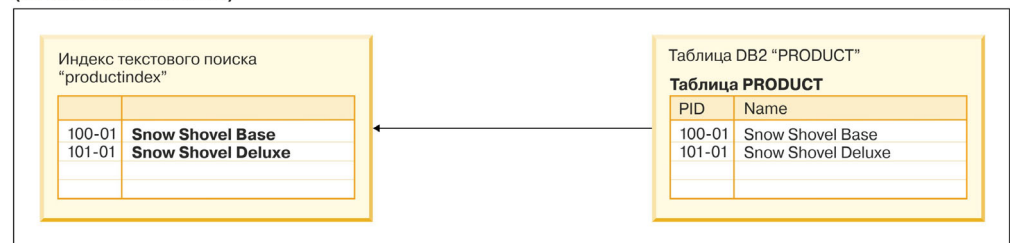
Пример

В этом примере пользователь создает индекс текстового поиска с именем `myschema.productindex` для таблицы `PRODUCT` из базы данных `SAMPLE`. На рис. 3 показано, что индекс пуст до выполнения начального обновления, а когда в `PRODUCT` добавляется дополнительная информация о продукте, необходимо выполнить инкрементное обновление.

Создание индекса для текста



Обновление текстового индекса (начальное обновление)



Обновление текстового индекса (инкрементное обновление)

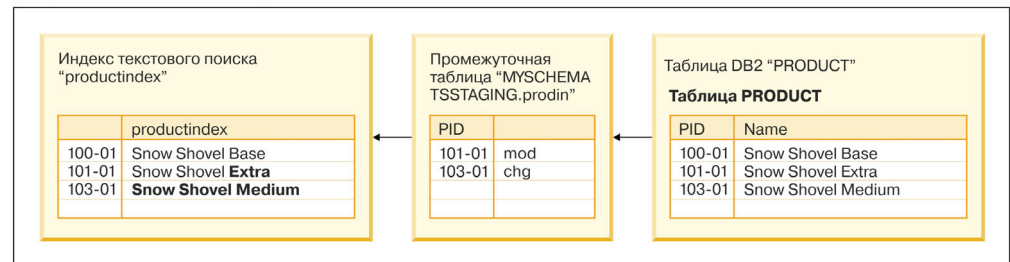


Рисунок 3. Два типа обновлений индекса текстового поиска

Лингвистическая обработка для DB2 Text Search

Сервер DB2 Text Search предоставляет пакеты словарей для поддержки лингвистической обработки документов и запросов на национальных языках. В качестве альтернативы основанного на словарях разделения по словам для китайского, японского и корейского языков механизм поиска использует сегментацию энграмм.

Для документа на одном из поддерживаемых языков лингвистическая обработка выполняется на этапе деления текста на лексемы (это означает, что текст разбивается на отдельные слова). Для неподдерживаемых языков документ анализируется при помощи сегментации по пробелам или же сегментации энграмм. Лемматизация (это обнаружение нормализованной формы слова с анализом слов по частям речи) для неподдерживаемых языков не выполняется.

При выполнении поиска с помощью индекса текстового поиска совпадение отмечается, если текст содержит лингвистические словоформы терминов запроса. Эти словоформы зависят от языка запроса.

Лингвистическая обработка документов на китайском, японском и корейском языках

Документы на китайском, японском или корейском языке обрабатываются с использованием сегментации энграмм.

Для механизма поиска получение хороших результатов поиска большей частью зависит от приемов, используемых для обработки текста. После извлечения текста из документа первым шагом его обработки является идентификация в тексте отдельных слов. Идентификация в тексте отдельных слов называется сегментацией. Во многих языках для опознания границ слов можно использовать пробельные символы (знаки пробела, конца строки и некоторые знаки пунктуации). Однако в корейском, китайском и японском языках пробельные символы для разделения слов не используются, поэтому здесь нужны другие приемы.

В настоящее время DB2 Text Search не предоставляет сегментации на базе словарей для китайского, японского и корейского языков.

Сегментация энграмм

Сегментация энграмм решает задачу определения границ слов путем индексации перекрывающихся пар символов. Так как используются два символа, эта технология также называется сегментацией диграфов.

Сегментация энграмм всегда возвращает все документы, удовлетворяющие условию поиска, однако эта технология может возвращать документы, не удовлетворяющие запросу.

По умолчанию DB2 Text Search поставляется с предварительной конфигурацией индекса, использующей сегментацию энграмм для китайского, японского и корейского языков.

Чтобы показать, как работают оба типа лингвистической обработки, рассмотрим следующий текст в документе: выборы губернатора префектуры Канагава. На японском этот текст содержит восемь символов. Для этого примера представим эти восемь символов как A B C D E F G H. Пример запроса, который может ввести пользователь - выборы губернатора; на японском этот текст содержит четыре символа E F G H. (В тексте документа и в примере запроса используются одинаковые символы).

После индексации документа механизм поиска выполнит сегментацию текста выборы губернатора префектуры Канагава, преобразовав его в следующие наборы символов: AB BC CD DE EF FG GH.

Пример запроса 'выборы губернатора' сегментируется в следующие наборы символов: DE EF FG GH. Если вы выполняете поиск с примером запроса выборы губернатора, документ будет найден в этом запросе, поскольку маркеры и для текста документа, и для запроса, идут в одном и том же порядке.

При помощи сегментации энграмм можно получить и больше результатов, но они могут оказаться менее точными. Например, если выполняется поиск на японском языке при помощи запроса Киото, а документ в индексе содержит текст город Токио, запрос Киото вернет документ с текстом город Токио. Причина в том, что в город

Токио и Киото подряд идут два одних и тех же иероглифа, хотя в Киото они составляют слово-название города, а в словосочетании "город Токио" "кио" (столица) - часть слова "Токио", а "то" - слово "город".

Инструменты командной строки для DB2 Text Search

В DB2 Text Search включены четыре инструмента командной строки, помогающие при работе с продуктом.

Инструмент конфигурирования

Для выполнения первоначального и последующих конфигурирования DB2 Text Search

Инструмент управления

Для выполнения различных административных задач, относящихся к серверу DB2 Text Search

Инструмент синонимов

Для добавления словарей синонимов к индексам текстового поиска и удаления словарей синонимов из индексов текстового поиска

Инструмент Log Formatter

Для просмотра и сохранения системных сообщений и сообщений трассировки

Вызов команд текстового поиска

Вы можете передавать на выполнение команды, запустив командную оболочку **db2ts** или вызвав одну из управляющих подпрограмм SQL, представляющую собой хранимую процедуру для DB2 Text Search.

Об этой задаче

Для использования командной оболочки **db2ts** строку команды надо передать как параметр. Командная оболочка **db2ts** похожа на командную оболочку DB2 тем, что, если используется удаленная база данных, команда должна содержать информацию о соединении. Однако **db2ts**, в отличие от командной оболочки DB2, не поддерживает сеанс; вместо этого каждая команда представляет собой отдельную единицу и поэтому должна устанавливать соединение отдельно. Если команда запускается локально для базы данных по умолчанию, заданной при помощи переменной среды **DB2DBDFT**, задавать соединение с базой данных не требуется. Задайте переменную среды **DB2DBDFT** на уровне операционной системы. Кроме того, если она задана помощи команды **db2set**, убедитесь, что используется одно и то же значение.

Используя управляющую подпрограмму SQL, можно передавать управляющие вызовы с клиента DB2, на котором не установлен текстовый поиск DB2. Можно вызвать либо универсальную управляющую подпрограмму SQL SYSTS_ADMIN_CMD, указав строку команды в качестве параметра, либо конкретную управляющую подпрограмму SQL для нужной команды.

Примечание: Сообщения об ошибках, возвращаемые командами **db2ts**, записываются в локали клиента, но сообщения, возвращаемые управляющими подпрограммами, записываются в локали, заданной аргументом локали сообщений (message-locale), или в локали en_US, если локаль не задана.

Поскольку некоторые команды, такие как **START FOR TEXT** и **STOP FOR TEXT**, не связаны с конкретной базой данных, их можно вызвать только при помощи командной оболочки **db2ts**.

Поддержка документов в формате расширенного текста

DB2 Text Search Версии 9.7 поддерживает документы в в формате расширенного текста. Благодаря этому DB2 Text Search может индексировать и искать документы в формате расширенного текста с помощью правильно сконфигурированного экземпляра DB2 Text Search.

DB2 Text Search поддерживает форматы текстового индекса TEXT, XML и HTML для подготовки индексов полнотекстового поиска в текстовых данных. DB2 версии 9.7 и поддержка документов в формате расширенного текста добавляют формат текстового индекса INSO для подготовки индексов полнотекстового поиска по данным в формате расширенного текста .

Документы в формате расширенного текста - это документы, содержащие текст и сведения о форматировании, такие как тип шрифта (жирный, курсив), размер шрифта, отступы и т.д.

Основной пример документов в формате расширенного текста - документы, которые создают распространенные офисные программы. Более конкретный пример - резюме. Такой тип документа обычно создается с помощью различных программных пакетов для офиса. DB2 Text Search теперь может хранить и индексировать такие документы, а также выполнять их поиск с большей легкостью и гибкостью. Можно корректно искать и извлекать ключевые слова не из английского словаря (например, название улицы, адрес электронной почты, компьютерные языки).

Дополнительную информацию о включении и конфигурировании функции поддержки документов в формате расширенного текста можно найти в следующих разделах.

Планирование решений текстового поиска

Знакомство с основными понятиями (например, поддерживаемыми типами документов и языками, ролями пользователей) поможет вам лучше использовать преимущества DB2 Text Search.

Особенности защиты для DB2 Text Search

Для защиты конфиденциальных данных требуется выполнить несколько действий.

Таблица 1. Проблемы, связанные с защитой, и их решения

Проблема	Решение
Политики доступа и аудита для базовой таблицы данных не наследуются для промежуточной таблицы.	Если в первичном ключе базовой таблицы используются конфиденциальные данные, промежуточная таблица должна быть соответствующим способом защищена.
Если конфиденциальные данные индексированы, они представлены в текстовом индексе, и некоторые данные могут быть доступны для чтения.	Нужно проверить разрешения доступа к папке и файлу, чтобы убедиться в их соответствии политике компании.

Таблица 1. Проблемы, связанные с защитой, и их решения (продолжение)

Проблема	Решение
Текстовые индексы расположены в файловой системе и доступны пользователям с привилегиями чтения и записи в каталоге, содержащем файлы текстового индекса. Привилегии доступа для каталогов текстового индекса автоматически не ограничиваются системой текстового поиска, кроме ограничений, наследованных для положения по умолчанию в пути экземпляра базы данных.	Проверьте (и при необходимости измените) разрешения доступа для каталогов пользовательских собраний, чтобы ограничить доступ владельцу процесса текстового поиска на сервере. Этот процесс запускается службой DB2TS, связанной с экземпляром DB2, от имени того пользователя, который задан для этой службы Windows.
ID изолированного пользователя и ID пользователя экземпляра должны входить в одну первичную группу, чтобы ID пользователя экземпляра был предоставлен правильный доступ к файлам, созданным ID изолированного пользователя, и наоборот.	Назначьте правильное вхождение в группы и разрешения доступа к файлам.

Роли пользователей

Для пользователей DB2 Text Search существуют различные роли и полномочия.

К типичным пользователям относятся:

- Владельцы экземпляров DB2
- Администраторы базы данных
- Владельцы пользовательских таблиц
- Пользователи, выполняющие текстовый поиск при помощи запросов

Владелец экземпляра

ID пользователя владельца экземпляра используется для запуска и остановки служб экземпляра DB2 Text Search на сервере DB2. Обратите внимание на то, что экземпляр DB2 и службы DB2 Text Search должны быть процессами одного и того же ID пользователя.

В Версии 9.7 у владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS.

В дополнение к выполнению своей основной задачи управления службами экземпляра ID пользователя владельца экземпляра DB2 действует как центр управления для всех изменений базы данных, инициированных DB2 Text Search. Другая важная задача - очистка всех неиспользуемых индексов текстового поиска.

Владелец экземпляра различным образом определяется в операционных системах UNIX и Windows:

- В операционных системах UNIX принадлежащие владельцу экземпляра имя и ID пользователя экземпляра задаются для команды **db2icrt**.
- В операционных системах Windows владелец экземпляра - это ID пользователя, запускающего службу экземпляра DB2.

Администратор баз данных

Администратор баз данных подключает и отключает базы данных для использования с DB2 Text Search. Другая задача администратора баз данных - очистка блокировок команд.

Эта роль соответствует полномочиям DBADM, позволяющим работать со всеми объектами баз данных, в том числе с индексами текстового поиска.

Владелец пользовательской таблицы

Владелец пользовательской таблицы определяет и поддерживает индексы текстового поиска.

Обычные задачи владельца пользовательской таблицы:

- Создание индексов текстового поиска и определение их характеристик
- Обновление индексов текстового поиска
- Изменение характеристик обновления индексов текстового поиска
- Отбрасывание индексов текстового поиска
- Периодическая очистка таблицы событий

Роль владельца пользовательской таблицы соответствует привилегии CONTROL таблицы, для которой создан индекс текстового поиска.

Пользователь, выполняющий текстовый поиск при помощи запросов

Пользователи, выполняющие поиск при помощи запросов, могут использовать в запросе SQL функции CONTAINS и SCORE DB2 Text Search для пользовательской таблицы. Они могут использовать также функцию xmlcolumn-contains в XQuery, которая обращается к таблице с индексом текстового поиска.

Специальных полномочий поиска для DB2 Text Search не существует. В зависимости от прав доступа, предоставленных пользователю для таблицы, на которой создан индекс, запрос разрешается или отклоняется. Если пользователи могут ввести оператор SELECT для данной таблицы, они могут также выполнить текстовый поиск для этой таблицы.

Пользователи, выполняющие поиск при помощи запросов, могут включить в свои запросы следующие функциональные возможности:

- Ограничить текстовый поиск конкретным документом (используя SQL или XQuery)
- Возвратить оценку, показывающую, насколько хорошо документ по сравнению с другими документами подходит для данного аргумента поиска (используя SQL)

Планирование мощности DB2 Text Search

На производительность и использование ресурсов DB2 Text Search влияет ряд факторов. При планировании емкости системы для DB2 Text Search следует учесть число параллельных обновлений индекса, планируемый размер текстовых индексов и время обработки для индексируемых документов. Для полнотекстового поиска следует учитывать различные факторы конфигурации системы и ее емкости.

DB2 Text Search теперь дает возможность выполнять запросы полнотекстового поиска для большинства типов данных в базе данных DB2, включая поддержку документов XML, документов в формате расширенного текста и в пользовательских форматах. Полнотекстовый поиск поддерживается с помощью экземпляра сервера текстового поиска, интегрированного в установку экземпляра базы данных и подключаемого через TCP/IP. Производительность полнотекстового индексирования и поиска зависит от параметров текстового индекса, конфигурации сервера текстового поиска и доступных системных ресурсов.

Размер индексов текстового поиска

На сервере текстового поиска DB2 индекс текстового поиска обычно имеет размер, равный 50-70% размера исходных данных. Начиная с DB2 версии 9.7 с пакетом Fix Pack 4, средний размер индекса текстового поиска составляет 50-150% размера исходных данных.

Размер индекса текстового поиска на интегрированном сервере текстового поиска зависит от следующих факторов:

- среднего размера документа;
- размера индексируемого столбца;
- размера первичного ключа;
- отношения размера первичного ключа к размеру данных индекса;
- количества и распределения терминов.

Ограничений на абсолютный размер какого-либо индекса текстового поиска нет. Однако при добавлении существенного объема данных в индекс текстового поиска или удаления их из него выполняется реорганизация структуры индекса текстового поиска для повышения производительности. Время выполнения этой операции зависит от размеров индекса.

Абсолютная пропускная способность зависит от типа данных и формата индекса. Сочетание условий, от которых зависит абсолютная пропускная способность, и условий, от которых зависит время выполнения, приводит к тому, что практически существуют ограничения на общий размер индекса текстового поиска.

Внедрение и конфигурирование сервера текстового поиска

Отдельный сервер текстового поиска конфигурируется для экземпляра базы данных. Минимальный рекомендуемый объем памяти для сервера текстового поиска - 4 Гбайта для производственного режима; это число увеличивается в соответствии с числом параллельных обновлений индексов.

Обновление индекса текстового поиска - ресурсоемкая операция как в отношении дискового ввода-вывода, так и в отношении ресурсов памяти и процессора. Для управления использованием ресурсов сервера текстового поиска есть несколько параметров конфигурации.

Факторы, влияющие на пропускную способность

Абсолютная пропускная способность обновления индекса текста зависит от типа данных и формата индекса. Оптимально обработка происходит при объеме 10-100 Кбайт текста на документ. Пропускная способности снижается для текстов меньше 1 кбайта или больше 1 Мбайта.

Время, требуемое для выполнения обновления текстового индекса, зависит преимущественно от следующих факторов:

- число документов для индексирования
- размер документа
- тип индекса
- параллелизм обновления индекса
- конфигурация сервера текстового поиска

Время обработки каждого документа - это сумма приблизительно фиксированного времени и переменного времени. На фиксированное время влияет тип документа, например простой текст, XML или INSO. Фиксированное время приблизительно, так как возможны незначительные отклонения во времени, требуемом для выделения или повторного использования памяти. Переменное время определяется в первую очередь размером документа и вариациями лингвистической обработки.

Для индексов документов INSO на время обработки может также повлиять обработка разных типов MIME.

Число документов, которые можно обработать за заданный промежуток времени, больше для документов меньших размеров. Однако общая пропускная способность для небольших документов меньше, чем для больших, из-за фиксированной части стоимости обработки каждого документа.

Изменение производительности запросов больше всего заметно при наличии большого количества результатов, а не большого размера индекса текстового поиска. Например, запрос с одним предикатом и одним термом поиска по индексу текстового поиска размером 100 Гбайт будет работать почти так же, как и поиск по индексу текстового поиска объемом 800 Гбайт, если они дают одинаковое число результатов.

Конфигурация сервера текстового поиска DB2

Можно настраивать конфигурацию текстового поиска DB2, меняя размер очереди, размер кучи, количество потоков индексирования и другие параметры. Сбалансировав эти параметры, можно добиться оптимальной производительности системы.

Рекомендуется, чтобы в конфигурации сервера текстового поиска DB2 количество потоков индексирования не превышало количество процессоров, а количество потоков параллельного обновления не превышало количество потоков индексирования.

При работе с интегрированным сервером текстового поиска DB2 остановите экземпляры службы текстового поиска DB2 с помощью команды **STOP FOR TEXT** перед изменением конфигурации, после чего используйте для изменения значений утилиту configurationTool. Например:

```
configTool configureParams -configPath <путь-конфигурации> -numberOfIndexingThreads 3
```

Чтобы изменения вступили в силу, перезапустите сервер текстового поиска DB2.

Конфигурирование максимального размера кучи:

Когда документ получен из потока входных документов, его содержимое помещается в очередь документов. Документы, помещенные в очередь документов, остаются в ней, пока их не проиндексирует активный поток индексирования.

В обычных случаях скорость помещения документов в очередь выше, чем скорость их анализа и индексирования. Следовательно, в какой-то момент времени очередь документов будет заполнена и поток приема документов будет заблокирован, пока в очереди не освободится место.

По мере заполнения очереди документов необработанными документами очередь расходует память кучи. Память используется также для обработки документов - синтаксического анализа и индексирования. Общее потребление памяти должно быть меньше максимального размера кучи для процесса. По умолчанию размер кучи

skonфигурирован равным 1500 Мбайт. Если планируется постоянно обрабатывать большие документы, размер кучи следует настроить соответствующим образом.

Кроме того, следует учитывать соотношение между размерами памяти входящей и исходящей очереди и памяти кучи. Размер очереди определяется потреблением памяти документами в этой очереди. Если требуется обрабатывать большие документа, например по 20 Мбайт, и решено увеличить размер памяти очереди, следует также рассмотреть возможность увеличения кучи.

Переменная *startupHeapSize* (до DB2 Версии 9.7 Fix Pack 3 она называлась *maximumHeapSize*) задает максимально разрешенный размер кучи для интегрированного сервера текстового поиска DB2. Начальный размер кучи по умолчанию - 1,5 Гбайта. Можно задать объем от 1,5 Гбайт до максимального количества памяти, которое позволяет использовать операционная система и JVM. Рассмотрим следующие примеры:

- В Windows с 32-битной JVM максимальный объем памяти кучи для процесса - 2 Гбайта. Поэтому значение параметра *startupHeapSize* должно быть не больше 2 Гбайт. Например, задайте 1,8 Гбайт.
- В системе AIX с 64-битной JVM максимальный размер кучи ограничен только объемом виртуальной памяти системы. Если используется интегрированный сервер текстового поиска и требуется постоянно обрабатывать большие документы со средним размером 20 Мбайт, следует увеличить значение *startupHeapSize* примерно до 4 Гбайт.

Для конфигурирования максимального размера очереди введите следующую команду.

- до DB2 Версии 9.7 Fix Pack 3:
`configTool configureParams -configPath <полный_путь_к_папке_конфигурации>
-maximumHeapSize <значение>`
- Начиная с DB2 версии 9.7 Fix Pack 4:
`configTool configureParams -configPath <полный_путь_к_папке_конфигурации>
-startupHeapSize <значение>`

Где <значение> - размер кучи, а <полный-путь-к-папке-конфигурации> - полный путь к файлу `config.xml` для сервера текстового поиска DB2.

Потоки индексирования DB2 Text Search:

Несколько потоков индексирования работают параллельно над синтаксическим анализом и индексом документов. Обычно это уменьшает общее время обновления индекса текстового поиска.

Потоки индексирования выбирают документы из очереди и управляют процессом индексирования. Они используют потоки предварительной обработки индекса для подготовки содержимого документа к индексированию и записывают результат в собрание текстового индекса.

Потоки предварительной обработки извлекают текст, определяют язык, разбивают текст на элементы и анализируют этот документ.

До DB2 Версии 9.7 Fix Pack 3 количество потоков предварительной обработки совпадало с количеством потоков индексирования. Начиная с DB2 Версии 9.7 Fix Pack 4 стало возможно отдельное конфигурирование потоков индексирования и потоков предварительной обработки. Обычно конфигурируется равное их

количество. Но в некоторых случаях, при обработке больших документов, увеличение количества потоков предварительной обработки может дать выигрыш в производительности.

Использование потоков индексирования

Если несколько потоков индексирования работают с одной коллекцией, эффект уменьшается из-за координации, необходимой для синхронизации обработки между потоками. При синтаксическом анализе однопоточная работа потока индексирования выгодна, но производительность может снижаться при объединении или записи на диск. Например, четыре потока индексирования, работающие с четырьмя разными текстовыми индексами, покажут лучший результат, чем четыре потока индексирования, работающие с одним текстовым индексом.

Количество потоков индексирования

Потоков индексирования должно быть не меньше двух, при этом их количество не должно превышать количество доступных процессоров. Максимальное количество параллельных обновлений индекса не должно превышать количество потоков индексирования во избежание объединения потоков. При слишком большом количестве потоков индексирования или параллельных обновлений индекса общая производительность системы снижается из-за увеличения использования памяти на переключение контекстов процесса.

Например, если 40 текстовых индексов часто обновляются на системе с 8 процессорами, не следует использовать больше 8 потоков индексирования. Следует также использовать расписание с чередованием обновлений текстовых индексов для минимизации конфликтов между потоками индексирования.

По умолчанию значение параметра количества потоков индексирования равно 4, такое же значение применяется к потокам предварительной обработки индекса.

В DB2 Версии 9.7 Fix Pack 3 или более ранним для конфигурирования количества потоков индексирования в настройках интегрированного текстового поиска измените следующие параметры в файле *полный-путь-к-папке-конфигурации/config.xml*:

```
<numberOfIndexers>4</numberOfIndexers>
```

Начиная с DB2 Версии 9.7 Fix Pack 4, для конфигурирования количества потоков индексирования введите следующую команду:

```
configTool configureParams -configPath полный_путь_к_папке_конфигурации  
-numberOfIndexingThreads значение
```

Где *значение* - количество потоков, а *полный-путь-к-папке-конфигурации* - полный путь к файлу *config.xml* для сервера текстового поиска DB2.

Для конфигурирования количества потоков предварительной обработки введите следующую команду:

```
configTool configureParams -configPath полный_путь_к_папке_конфигурации  
-numberOfPreprocessingThreads значение
```

Где *значение* - количество потоков, а *полный-путь-к-папке-конфигурации* - полный путь к файлу *config.xml* для сервера текстового поиска DB2.

Размер памяти очереди DB2 Text Search:

Для оптимальной обработки обновлений индексов надо правильно задать размер памяти очереди DB2 Text Search.

Для DB2 Версии 9.7 Fix Pack 3 или более ранних пакетов Fix Pack, для сервера текстового поиска сконфигурируйте размер очереди документов, отредактировав файл `<полный_путь_к_папке_конфигурации>/config.xml`, где `<полный_путь_к_папке_конфигурации>` - полный путь к папке, содержащей файл `config.xml`. Измените следующий параметр:
`<documentQueueSize>120</documentQueueSize>`

Это значение должно быть числом от 4 (или числа потоков индексирования) до 120. Значение по умолчанию - 120.

Начиная с DB2 Версии 9.7 Fix Pack 4, можно управлять выделением памяти очередей и для базы данных, и для текстового сервера. Память очереди базы данных определяет число документов, которые можно за раз отправить текстовому серверу для обработки обновлений. Чтобы управлять объемом памяти очереди базы данных, обновите производную таблицу управления SYSIBMTS.TSDEFAULTS и задайте значение для параметра **DocumentResultQueueSize**. Значение по умолчанию - 30000. Это значение используется для ограничения объема памяти базы данных, резервируемой для каждой операции обновления для собрания.

Чтобы изменить это значение, введите команду:

```
db2 "update sysibmts.tsdefaults set defaultvalue='10000' where  
lower(defaultname)='documentresultqueuesize'"
```

Второй механизм управления памятью очередей применяется к текстовому серверу. Использование памяти очередей определяется двумя значениями конфигурации.

- *inputQueueMemorySize*:
Задает объем памяти для входной очереди на сервере индексирования. Входная очередь содержит документы, ожидающие предварительной обработки. Увеличение объема памяти ускоряет работу, но увеличивает потребление ресурсов. Размер по умолчанию - 15 Мбайт.
- *outputQueueMemorySize*:
Задает объем памяти для выходной очереди на сервере индексирования. Входная очередь содержит документы, ожидающие индексирования после предварительной обработки. Увеличение объема памяти ускоряет работу, но увеличивает потребление ресурсов. Размер по умолчанию - 15 Мбайт.

Обратите внимание на соотношение между размером входной и выходной очередей и размером памяти кучи. Размер очереди определяется потреблением памяти документами в этой очереди. Если вы собираетесь обрабатывать длинные документы, например, по 20 Мбайт каждый, рассмотрите возможность увеличения объема памяти очередей и размера кучи.

Чтобы изменить, например, размер *inputQueueMemory*, введите команду:

```
configTool configureParams -configPath <полный_путь_к_папке_конфигурации>  
-inputQueueMemorySize <значение>
```

где `<значение>` - объем памяти, а `<полный_путь_к_папке_конфигурации>` - полный путь к файлу `config.xml` для DB2 Text Search.

Планирование и оптимизация индексов DB2 Text Search

Характеристики источников данных имеют большое влияние на производительность.

Время, требуемое для выполнения обновления текстового индекса, зависит преимущественно от следующих факторов:

- число документов для индексирования
- размер документа
- тип индекса
- параллелизм обновления индекса
- конфигурация сервера текстового поиска

Время обработки каждого документа - это сумма приблизительно фиксированного времени и переменного времени. Фиксированное время зависит от типа документа, например, простой текст, XML или INSO. Фиксированное время приблизительно, так как возможны незначительные отклонения во времени, требуемом для выделения или повторного использования памяти. Переменное время определяется в первую очередь размером документа и вариациями лингвистической обработки.

Для индексов документов INSO на время обработки может также повлиять обработка разных типов MIME.

Число документов, которые можно обработать за заданный промежуток времени, больше для документов меньших размеров. Однако общая пропускная способность для небольших документов меньше, чем для больших, из-за фиксированной части стоимости обработки каждого документа.

Характеристики источников индексов DB2 Text Search:

Характеристики источников данных имеют большое влияние на производительность.

Используйте следующие приемы для повышения производительности индексирования или поиска:

- Для столбцов первичных ключей используйте числовые типы данных, такие как INTEGER, а не тип VARCHAR. Чтобы минимизировать объем передаваемых данных для результатов поиска, избегайте первичных ключей, составленных из нескольких столбцов VARCHAR.
- Убедитесь, что в системе достаточный объем доступной реальной памяти для операций обновления индекса. Для обновлений индексов требуется память сверх того, что требуется для пулов буферов баз данных. Если памяти недостаточно, операционная система использует пространство подкачки, что значительно снижает производительность поиска.
- Если во время обновлений индекса сервера текстового поиска должны обрабатываться много небольших документов, рассмотрите возможность уменьшения числа параллельных обновлений индекса и увеличения размеров очередей, чтобы увеличить максимальный поток документов к текстовому серверу. Подробную информацию смотрите в разделах о планировании мощности.
- Убедитесь, что индексируемое содержимое доступно и находится в нужном формате, так как если при обновлении индекса в таблицу событий записывается много сообщений об ошибках и предупреждений, производительность может снизиться.

Асинхронные изменения индексов:

Для повышения производительности индекс текстового поиска не синхронизируется с пользовательской таблицей внутри области транзакции DB2, в ходе которой текстовые документы обновляются, удаляются или добавляются в эту таблицу. Индекс текстового поиска обновляется асинхронно.

Для облегчения асинхронного обновления индекса текстового поиска для каждого индекса текстового поиска создается промежуточная таблица (другое название - таблицей журнала). Триггеры для текстовых таблиц захватывают любые изменения, например вставки, удаления или обновления текстового столбца, с которым связан индекс текстового поиска, и записывают их в промежуточную таблицу. Обновления индекса текстового поиска применяются позже во время ручного или автоматического обновления. Обновление вносится в копию очень маленького фрагмента индекса. Во время обновления можно по-прежнему выполнять поиск по индексу, но до окончания синхронизации нельзя получить доступ к обновленному индексу текстового поиска.

При использовании опции `updateautocommit` следует учитывать, что каждый цикл принятия приводит к значительному использованию памяти, что увеличивает общее время, необходимое для выполнения обновления индекса. Использование опции `commitcount` особенно полезно для начальных обновлений или обновлений, затрагивающих все строки или большую часть строк, если общее затраченное время велико, чтобы избежать потери сделанной работы в связи с откатом вследствие сбоя системы или сервера.

Для вычисления значения `commitcount` следует учитывать, что расчеты для обработки индекса текста сильно отличаются от обработки транзакций в базе данных. Каждый цикл рассматривается как полное обновление индекса, поэтому выполняются шаги подготовки и завершения, в том числе необходимые слияния сегментов индекса или реорганизация индекса. Таким образом, очень короткие циклы принятия существенно увеличат общее время, и этого следует избегать.

Примеры

Пример 1. При среднем размере документа 10 Кбайт на строку, пропускной способности конкретной системы 12 Гбайт/час и общем объеме данных для обработки 240 Гбайт для обновления индекса без циклов принятия потребуется около 20 часов при обработке 1 200 000 строк в час. Можно задать для опции `commitcount` значение от 2000000 до 2500000, чтобы принятие выполнялось каждые 2 часа.

Пример 2. При среднем размере документа 100 байт на строку, пропускной способности 1 Гбайт/час на конкретном компьютере и общем объеме данных для обработки 60 Гбайт обновление индекса без циклов принятия займет около 60 часов при обработке 10 000 000 строк в час. В этом случае подходящим значением для опции `commitcount` будет 30 000 000.

Оптимизация индекса DB2 Text Search:

Оптимизация индекса DB2 Text Search компактифицирует индекс текстового поиска и ускоряет индексирование и поиск. Оптимизация удаляет из индекса текстового поиска удаленные документы и сливает файлы сегментов индекса на диске.

Оптимизацию и индексирование одного и того же индекса нельзя выполнять параллельно. Учитывайте это при составлении расписания сеансов оптимизации и индексирования. Но оптимизацию можно выполнять параллельно с поиском. Во время оптимизации индекса может наблюдаться повышенное потребление дискового пространства, особенно если параллельно выполняется поиск по тому же самому индексу.

Вы можете оптимизировать индекс после полного индексирования набора документов или после инкрементных обновлений индекса. Оптимизация индекса

может занять значительное время в зависимости от размера индекса. Если при инкрементных обновлениях часто добавляются новые документы, выполняйте оптимизацию реже, чтобы минимизировать использование памяти на процесс оптимизации.

Чтобы оптимизировать индекс, запустите инструмент управления с командой **optimizeIndex**. Например:

```
adminTool.bat optimizeIndex -configPath c:\mydbinst\db2tss\config -collectionName MyCollection
```

Чтобы проверить состояние последнего запущенного процесса оптимизации, запустите инструмент управления с командой **optimizeIndexStatus**.

Потребление дискового пространства:

Размер текстового индекса

Объем дискового пространства, используемого индексом текстового поиска, сильно зависит от характера текста документа. При этом между размером индекса текстового поиска и размером исходных данных существует приблизительно линейная зависимость. Как правило, размер индекса на диске составляет от 50% до 150% размера исходного текста. Например, индекс текстового поиска для 100 000 документов размером 200 Кбайт, вероятно, потребует около 1100 Мбайт (100000 x 20 Кбайт x 55%). Отношение размера индекса текстового поиска к размеру исходных документов зависит от следующих факторов:

- среднего размера документа;
- размера индексируемого столбца;
- размера первичного ключа;
- отношения размера первичного ключа к размеру данных индекса;
- количества и распределения терминов.

В ходе обновления индекса требуется дополнительное место. В среднем требуемый размер свободного пространства в 2-3 раза больше окончательного размера индекса текстового поиска, если не достигнут предел размера сегмента. Требуемое свободное пространство в 2-3 раза больше максимального размера сегмента. Дисковое пространство резервируется даже после объединения сегментов, если старые сегменты использовались в поиске.

Абсолютного предела размера индекса текстового поиска не существует, но сочетание производительности и времени завершения работы диктует практические пределы размера индекса. Например, при удалении из индекса или добавлении в индекс заметного количества данных структура индекса текстового поиска объединяется для улучшения производительности при выполнении запроса, и время, необходимое для объединения, зависит от размера индекса.

Файлы журналов

Кроме файла `db2diag.log`, текстовый поиск DB2 создает файлы трассировки и журнала утилиты конфигурирования, содержащие сообщения сервера текстового поиска DB2.

По умолчанию файлы журнала сервера текстового поиска расположены в каталоге `db2tss/log`. Если требуется хранить базу данных DB2 и файлы журнала текстового поиска в одном месте, задайте положение *<домашний-каталог-экземпляра>/sqllib/db2dump/tslog* в UNIX или *<путь-профиля-экземпляра>\<имя-экземпляра>\db2tss\tslog* в Windows.

Убедитесь, что в указанном месте достаточно свободного дискового пространства для файлов журнала. Требуется как минимум 100 Мбайт свободного места. Если места недостаточно, служба текстового поиска останавливается и выдает ошибку переполнения диска.

Начиная с DB2 Версии 9.7 Fix Pack 4, можно конфигурировать запись в журнал сервера текстового поиска, изменяя файлы свойств записи в журнал, расположенные в каталоге `db2tss\config`. Файл `ecmts_logging.properties` определяет свойства записи трассировки. Файл `ecmts_config_logging.properties` определяет свойства записи в журнал инструментов командной строки. Это файлы свойств `java.util.Logger`. Комментарии внутри файлов объясняют, как их редактировать.

Административные таблицы

Если при выполнении команды **CREATE INDEX FOR TEXT** не указано табличное пространство для административных таблиц для текстового индекса, административная таблица создается в том же табличном пространстве, что и базовая таблица. При определении нужного расположения учитывайте следующее:

- Промежуточная таблица для текстового индекса

Промежуточная таблица содержит ссылку на строки, которые были обновлены в базовой таблице для инкрементного обновления текстового индекса. Эта таблица автоматически очищается при каждом обновлении:

Размер =
количество строк обновления индекса * (длина первичного ключа базовой таблицы + 18)

- Таблица событий для текстового индекса

Таблица событий содержит сведения о состоянии обработки текстового индекса, включая ошибки и предупреждения в ходе обновления индекса. В худшем случае, если каждый документ отклоняется вследствие нефатальной ошибки, количество событий равно количеству документов с прибавлением нескольких начальных и конечных сообщений процесса обновления. Таблица событий не очищается автоматически, и до выполнения операции **CLEAR EVENTS FOR INDEX** ее размер продолжает увеличиваться.

Размер таблицы событий
= количество событий * (длина первичного ключа базовой таблицы + 1036)

Положение индексов DB2 Text Search:

Для интегрированного сервера Text Search метаданные конфигурации и сбораний хранятся в *домашний_каталог_экземпляра/sql1lib/db2tss/config* в UNIX или в *путь_профиля_экземпляра\instance_name\db2tss\config* в Windows.

Если вы планируете создать несколько больших индексов, рассмотрите возможность их хранения на отдельных дисковых устройствах или массивах, в особенности если будут выполняться одновременные обновления индексов по расписанию.

Параметры, специфичные для индексов, для обновлений индексов DB2 Text Search:

Начиная с DB2 Версии 9.7 Fix Pack 4, для повышения производительности можно сконфигурировать следующие параметры, специфичные для собрания:

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

Чтобы изменить параметры индексирования в конкретном собрании, отредактируйте файл `\config\collections\<имя_собрания>\collection.xml`. Чтобы изменить параметры по умолчанию для собраний, которые будут создаваться в дальнейшем, задайте значения этих параметров в файле `\config\defaults\collection.xml`.

MaxMergeDocs

Параметр **MaxMergeDocs** определяет самый большой сегмент (по числу документов), который можно слить с другими сегментами в индексе. Надо соблюсти баланс между общей пропускной способностью индексации и временем слияния сегментов.

Если задать для параметра **MaxMergeDocs** маленькое значение (например, 100 000 документов), размер сегментов будет ограничен. В этом случае слияние сегментов проходит быстрее, и индексация выполняется более гладко, без задержек. Однако если вы работаете с большим содержимым, получится слишком много сегментов, что со временем вызовет ухудшение пропускной способности индексации.

Если указать для параметра **MaxMergeDocs** большое значение (например, 100 000 000 или 500 000 000 документов), сегментов будет меньше (если только индекс не будет очень большим), и общая пропускная способность индексации увеличится. Однако слияние сегментов будет отнимать больше времени, и вы можете столкнуться с задержками при индексации.

Обычно значение параметра **MaxMergeDocs** должно быть больше для собраний небольших документов и меньше для собраний больших документов.

MergeFactor

Параметр **MergeFactor** определяет число сегментов, сливаемых за один раз, а также управляет общим числом сегментов, которое можно накопить в одном индексе. Здесь стоит выбор между частыми небольшими слияниями (например, два за один раз) и редкими большими слияниями (например, 10 за один раз). Чтобы избежать задержек, можно указать меньшее значение для параметра **MergeFactor**. Изменение параметра слияния обычно не влияет на производительность.

BufferSize

Параметр **BufferSize** задает объем оперативной памяти, который можно использовать для буферизации добавляемых документов перед тем, как они будут сброшены на диск в качестве нового сегмента. Здесь стоит выбор между частыми небольшими сбросами на диск и редкими большими сбросами на диск. В некоторых случаях можно повысить производительность, увеличив значение параметра **BufferSize**. Например, при индексировании одного собрания небольших документов увеличение размера буфера повысит производительность, особенно для первых 100 000 документов в индексе.

Настройка системы DB2 Text Search

На производительность обновления текстовых индексов и выполнения запросов текстового поиска влияют различные характеристики системы.

Рассмотрите следующие факторы:

- О портах TCP/IP для Windows
- Дескрипторы файлов

О портах TCP/IP для DB2 Text Search и Windows:

В 32-битных операционных системах Windows на возможность обработки больших нагрузок запросов влияет число портов TCP/IP и время ожидания повторного использования порта.

Назначения портов в Windows (32-битной)

Интегрированная версия DB2 Text Search работает как отдельный процесс на том же хосте, что и сервер баз данных. Сервер баз данных и текстовый сервер связываются через соединение TCP/IP.

На число доступных портов для соединений TCP/IP влияет число портов и время ожидания повторного использования порта после закрытия соединения. Значения этих параметров по умолчанию могут быть недостаточны, чтобы обеспечить достаточное число доступных портов для обслуживания значительной нагрузки запросов. Если у вас слишком мало портов TCP/IP, может возвращаться ошибка CIE00756 Соединение не удалось установить.

Если возникает ошибка CIE00756 Соединение не удалось установить, введите следующую команду, чтобы проверить использование портов на сервере:

```
netstat -n  
netstat -n | c:\windows\system32\find /I номер_порта
```

Если выводится много соединений TCP/IP и локальных адресов 127.0.0.1:номер_порта в состоянии TIME_WAIT, высока вероятность, что у сервера заканчиваются порты TCP/IP.

Чтобы определить номера портов DB2 Text Search, введите следующую команду:
configTool printAdminHTTPPort -configPath %INSTPROF%\%DB2INSTANCE%\db2tss\config

где для INSTPROF задано значение переменной реестра **DB2INSTPROF**, применяемой к интегрированным установкам сервера DB2 Text Search.

Параметры портов

Параметрами портов управляют следующие записи реестра, находящиеся в HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TCP/IP\Parameters:

- **TcpTimedWaitDelay**

Значение DWORD в диапазоне от 30 до 300, определяющее время в секундах, которое должно пройти до того, как TCP/IP сможет освободить закрытое соединение и повторно использовать его ресурсы. Задайте для значения **TcpTimedWaitDelay** более низкое значение, чтобы уменьшить время, в течение которого гнезда остаются в состоянии TIME_WAIT.

- **MaxUserPort**

Значение DWORD, определяющее максимальный номер порта, который может быть назначен TCP/IP, когда прикладная программа запрашивает свободный пользовательский порт. Задайте для **MaxUserPort** большее значение, чтобы увеличить общее число гнезд, которые можно соединить с портом.

Система, генерирующая много требований соединения, может заработать быстрее, если задать для **TcpTimedWaitDelay** значение 30 секунд, а для **MaxUserPort** - 32678.

После добавления или изменения этих записей реестра перезагрузите компьютер Windows, чтобы изменения вступили в силу.

Дескрипторы файлов DB2 Text Search:

При обновлении индексов и выполнении запросов DB2 Text Search для обработки множественных требований обновления индекса и поиска используются такие системные ресурсы, как дескрипторы файлов.

Обычно в системе число открытых дескрипторов файлов для каждого процесса может быть ограничено сравнительно небольшим числом, например, 1024, из-за чего у сервера текстового поиска могут закончиться дескрипторы файлов. В этом случае требования поиска и обновления завершаются неудачно.

Для исправления этой ошибки:

- Поищите в журналах сервера исключительную ситуацию со строкой сообщения вроде `too many open files` (слишком много открытых файлов).
- В системах UNIX проверьте ограничения системы с помощью команды `ulimit -a`.

Чтобы увеличить число дескрипторов файлов, выполните следующие действия:

1. Остановите сервер текстового поиска.
2. Увеличьте число дескрипторов файлов, разрешенных для одного процесса, как описано в руководстве по вашей операционной системе. Это увеличение числа дескрипторов файлов должно быть достаточным для всех требований во всех сеансах регистрации.
3. Перезапустите сервер текстового поиска.

Форматы документов, поддерживаемые для DB2 Text Search

Надо задать формат (или тип) текстовых документов, в которых собираетесь выполнять текстовый поиск с использованием DB2 Text Search. Эта информация необходима для индексации текстовых документов.

Данные в текстовом столбце могут быть простым текстом, текстом расширенном текстовом формате, документом HTML или документом XML. Для документов выполняется синтаксический анализ, чтобы извлечь для индексации и, тем самым, сделать доступными для поиска только релевантные части. Например, теги и метаданные в документах HTML не индексируются, поэтому они не извлекаются.

Поддерживаемые типы данных

Типы данных в текстовых столбцах, которые вы хотите индексировать для поиска, могут быть двоичными или символьными.

DB2 Text Search поддерживает следующие типы данных:

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

Преобразование неподдерживаемых форматов и типов данных

Можно использовать собственную функцию для преобразования неподдерживаемого формата или типа данных в поддерживаемый формат или тип данных.

Создав текстовый индекс с использованием пользовательской функции, можно преобразовать неподдерживаемый формат в поддерживаемый формат, который может быть обработан при индексации, путем фильтрации неподдерживаемых символов.

Этот подход можно использовать также для индексации документов, хранящихся на внешних неподдерживаемых складах данных. В этом случае, когда столбец DB2 содержит ссылки на документы, можно использовать пользовательскую функцию, возвращающую содержимое документов по соответствующей ссылке на документ.

Поддерживаемые языки и кодовые страницы

При первом создании индекса текстового поиска вы можете указать, что синтаксический анализ текстовых документов будет проводиться для конкретного языка. Можно задать также, что при поиске термины запроса будут интерпретироваться для некоторого языка. Дополнительно при создании индекса текстового поиска для столбца двоичного типа данных можно указать кодовую страницу.

Спецификация языка

Локаль - это сочетание информации о языке и территории (регионе или стране), которое представляется пятисимвольным кодом локали. Вы определяете локаль сообщений для процедуры администрирования текстового поиска, передавая этой процедуре код локали. Уточнение кодов локалей возможно в зависимости от локалей, установленных на сервере DB2.

Есть важное различие между заданием языка при создании индекса текстового поиска и заданием языка при вводе запроса поиска:

- Локаль, заданная в вашей команде **db2ts CREATE INDEX**, определяет язык выполнения лексического или другого анализа документов для индексирования. Если известно, что все документы в столбце для индексирования используют некоторый конкретный язык, укажите применимую локаль при создании индекса текстового поиска. Если вы не указали локаль, для задания значения по умолчанию для параметра **LANGUAGE** будет использована территория базы данных. Чтобы просматривать ваши документы автоматически для определения локали, задайте в производной таблице SYSIBMTS.TSDEFAULTS для атрибута **LANGUAGE** значение AUTO. Производная таблица SYSIBMTS.TSDEFAULTS представляет значения по умолчанию в базе данных для текстового поиска при помощи пар атрибут-значение.
- Заданная вами в запросе поиска локаль используется при лингвистической обработке запроса и помогает определить основные формы термина запроса. После того, как локаль основной формы определена, она не играет роли в самом процессе поиска. Таким образом, можно использовать английский язык для запроса, но получать в результате поиска документы на русском языке, если в этих документах присутствует термин поиска в своей основной форме.

Список поддерживаемых локалей можно найти [здесь](#).

Спецификация кодовой страницы

Документы можно индексировать, если они используют одну из поддерживаемых кодовых страниц DB2. Хотя указание кодовой страницы при создании текстового поиска необязательно, это поможет идентифицировать кодировку символов в двоичных столбцах. Если для двоичных столбцов кодовая страница не указана, используется кодовая страница из свойств столбца. .

О размерах документов

DB2 Text Search накладывает ограничения на размер индексируемых документов и на число символов в документе.

Максимальный размер документа, который можно успешно обработать - 100 Мбайт. Если размер документа превышает этот предел, он отвергается и в таблице событий создается запись с этой информацией, в том числе с первичным ключом для ее идентификации. Обработка продолжается для других документов, участвующих в этой операции изменения.

Усечение документов

DB2 Text Search ограничивает число символов Unicode, которые можно индексировать для каждого текстового документа. Иногда это ограничение числа символов приводит к усечению больших текстовых документов в индексе текстового поиска.

Значение по умолчанию для числа символов Unicode, разрешенных для каждого текстового документа, составляет 10 000 000. Для каждого документа в расширенном текстовом формате этот предел применяется после преобразования документа в простой текст.

Если на стадии анализа документ усекается, вы получаете предупреждение, что некоторый текст не был обработан правильно или полностью. Текст, который обработан неправильно или не полностью, не индексируется и не может участвовать в операции поиска. Подробности об этом предупреждении записываются в таблицу событий, созданную для каждого индекса текстового поиска.

При желании вы можете удалить усеченный документ, чтобы исключить непредсказуемое поведение во время поиска. Удалить документ можно, стерев его в таблице DB2 или изменив значение для документа на пустое или null.

Максимальное число индексируемых для документов XML символов такое же, как и для других документов (не XML), но при расчете количества символов принимается во внимание не только сам текст, но и теги, имена атрибутов и значения атрибутов. Директивы и комментарии XML не индексируются и не учитываются при вычислении предела отсечения.

Глава 3. Установка и конфигурирование DB2 Text Search

DB2 Text Search - это необязательный устанавливаемый компонент, установка и конфигурация которого полностью интегрированы с установкой всех продуктов серверов баз данных DB2.

Программа установки DB2 может полностью автоматически установить и сконфигурировать DB2 Text Search. Требуемые шаги отличаются для различных платформ. На рис. 4 показан процесс установки и конфигурирования в ОС Windows, а на рис. 5 на стр. 28 показан этот процесс для ОС Linux и UNIX.

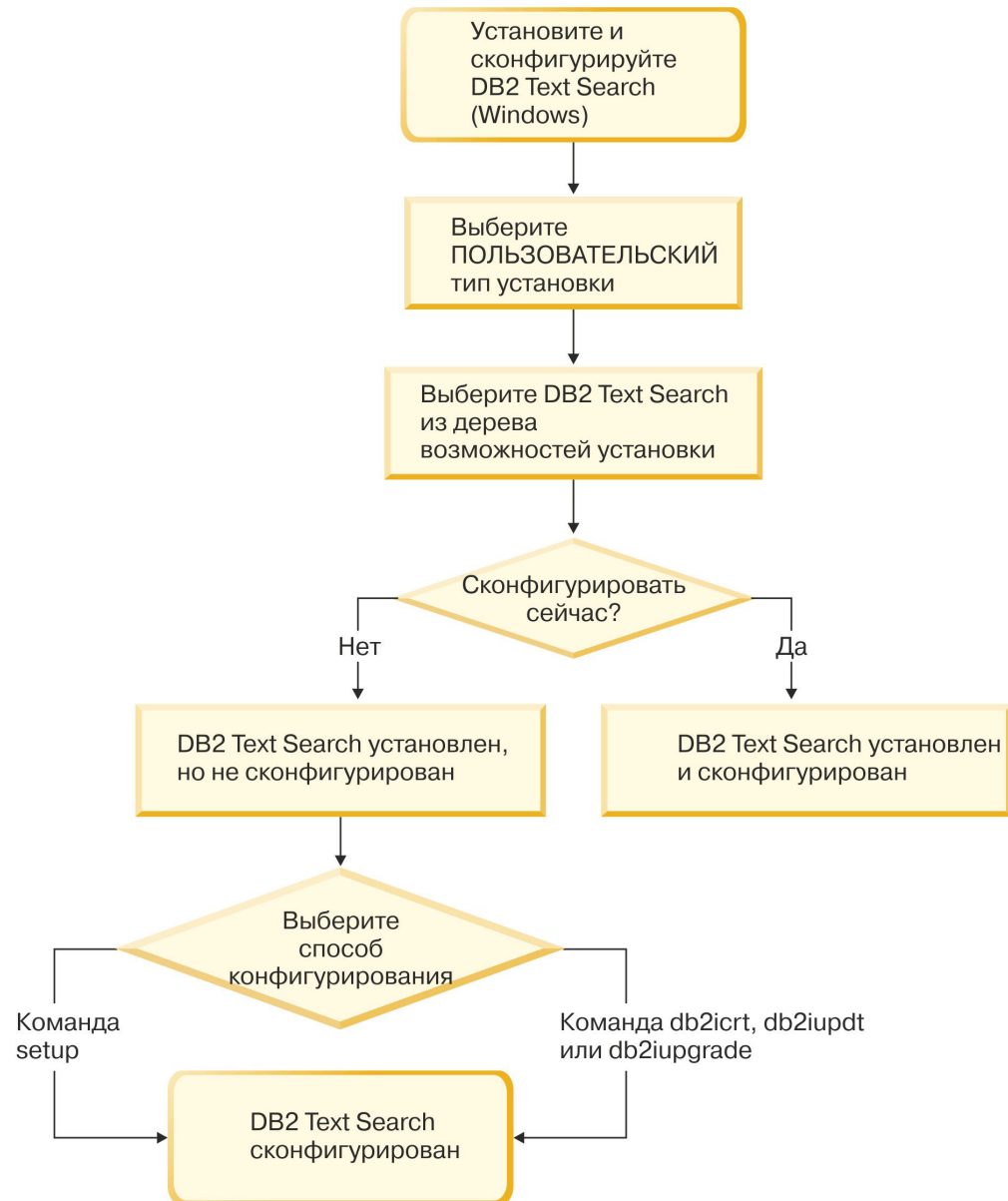


Рисунок 4. Установка и конфигурирование на платформах Windows

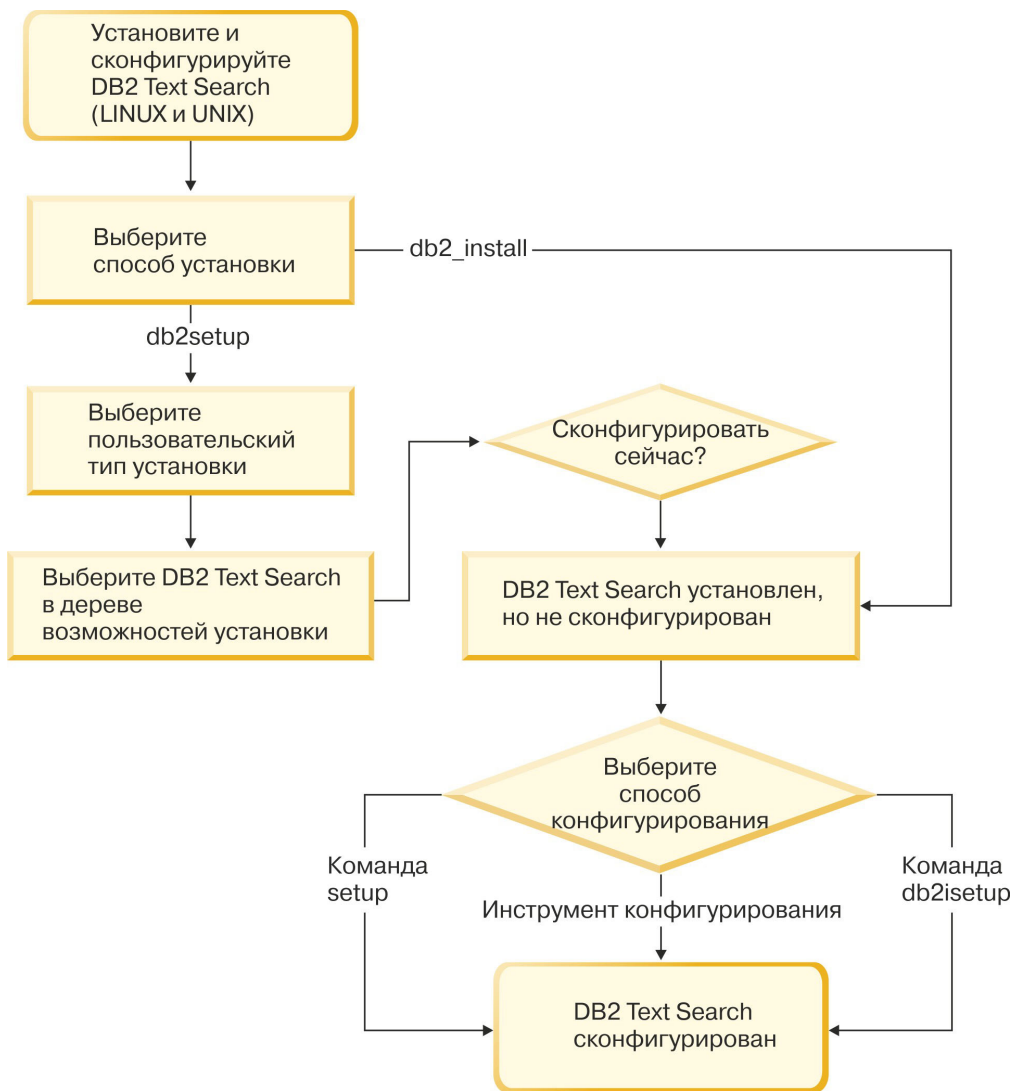


Рисунок 5. Установка и конфигурирование на платформах Linux и UNIX

1. При запуске **db2setup** от имени пользователя без полномочий root необходимо, чтобы системный администратор (пользователь с полномочиями SYSADM) затем запустил команду **db2rfe** для резервирования номера порта, который требуется использовать в файле служб.

Ограничения

DB2 Text Search не поддерживается в среде многораздельных баз данных.

Установка и конфигурирование текстового поиска DB2 при помощи мастера по установке DB2

Установить DB2 Text Search можно при помощи мастера по установке DB2 в составе пользовательской установки продукта баз данных DB2.

Об этой задаче

Выполните пользовательскую установку продукта баз данных DB2 и выберите в дереве возможностей DB2 Text Search. Возможность DB2 Text Search можно сконфигурировать автоматически либо позднее вручную.

Процедура

Чтобы выполнить пользовательскую установку DB2 Text Search при помощи **setup** или **db2setup**:

1. Установите сервер DB2, выполнив инструкции для используемой платформы:
 - "Установка серверов DB2 при помощи мастера по установке DB2 (Windows)", *Установка серверов DB2*
 - "Установка серверов DB2 при помощи мастера по установке DB2 (Linux и UNIX)", *Установка серверов DB2*

Вы можете выбрать компонент DB2 Text Search в дереве возможностей. При установке есть возможность сконфигурировать DB2 Text Search для экземпляра по умолчанию. Если вы не хотите конфигурировать DB2 Text Search, пропустите шаг 2.

2. Чтобы сконфигурировать DB2 Text Search самостоятельно, введите правильные имя службы и номер порта, если в этих полях еще нет значений. Конфигурировать возможность DB2 Text Search сразу же после ее установки не обязательно; ее можно сконфигурировать и позднее. Инструкции по конфигурированию после установки смотрите в разделе "Конфигурирование текстового поиска DB2" на стр. 42.

Установка серверов DB2 (Windows)

В этом разделе описана процедура запуска мастера по установке DB2 в Windows. Мастер по установке DB2 позволяет задать параметры установки и установить продукт баз данных DB2 в системе.

Прежде чем начать

Перед запуском мастера по установке DB2:

- Если вы собираетесь сконфигурировать среду многораздельных баз данных, прочитайте тему "Настройка среды многораздельных баз данных".
- Убедитесь, что в системе выполнены предварительные требования к установке и достаточно оперативной и дисковой памяти.
- Если вы собираетесь использовать LDAP для регистрации сервера DB2 в Active Directory операционных систем Windows, перед установкой надо расширить схему каталога, иначе надо будет регистрировать узел и каталогизировать базы данных вручную. Дополнительную информацию смотрите в теме "Расширение схемы Active Directory для служб каталога LDAP (Windows)".
- Если вы собираетесь использовать агент DB2 IBM® Tivoli Monitoring for Databases, перед установкой продукта DB2 посмотрите в теме "Установка агента DB2 IBM Tivoli Monitoring for Databases при помощи программы установки DB2" подробности и ограничения.
- Для выполнения установки требуется учетная запись локального администратора с необходимыми правами доступа. На серверах баз данных DB2, где в качестве пользователя экземпляра DAS и DB2 можно использовать LocalSystem и если не используется возможность разделения баз данных, эту установку может выполнить обычный пользователь (не администратор) с расширенными привилегиями.

Примечание: Если планируется установка продукта от имени обычного пользователя (не администратора), перед попыткой установки продукта баз данных DB2 надо установить библиотеку времени выполнения VS2005. Библиотека времени выполнения VS2005 должна быть в операционной системе до начала установки продукта баз данных DB2. Эту библиотеку можно получить на сайте загрузки библиотек времени выполнения Microsoft. Существует два варианта: `vcredist_x86.exe` для 32-битных систем и `vcredist_x64.exe` для 64-битных систем.

- Хотя это и не обязательно, рекомендуется закрыть все программы, чтобы программа установки могла изменять любые файлы на компьютере, не требуя перезагрузки.
- Установка продуктов DB2 с виртуального диска или с неотображенного сетевого диска (такого как `\\hostname\sharename` в Windows Explorer) не поддерживается. Перед попыткой установить продукты DB2 products, надо отобразить этот сетевой диск, назначив ему буквы диска Windows (например, Z:).

Об этой задаче

Ограничения

- Под каждой учетной записью пользователя может работать только один экземпляр мастера по установке DB2.
- Имя копии DB2 и имя экземпляра не могут начинаться с цифры. Имя копии DB2 может содержать не более 64 символов; разрешены символы A-Z, a-z и 0-9.
- Имя копии DB2 и имя экземпляра должны быть уникальными среди всех копий DB2.
- Возможности XML можно использовать только для однораздельных баз данных.
- Никакой другой продукт баз данных DB2 нельзя установить в тот же самый путь, если в нем уже установлен один из следующих продуктов:
 - IBM Data Server
 - пакет драйверов IBM Data Server
 - *Информационный центр DB2*
- В поля мастера по установке DB2 нельзя вводить символы национальных алфавитов.
- Если вы включили расширенную защиту в Windows Vista или в Windows 2008, пользователи должны входить в группу DB2ADMNS или DB2USERS, чтобы иметь возможность запускать локальные команды и программы DB2, так как возможность расширенной защиты (User Access Control) ограничивает по умолчанию привилегии локального администратора. Если пользователь не входит в одну из этих групп, у него не будет доступа для чтения локальной конфигурации DB2 или данных прикладных программ.

Процедура

Для запуска мастера по установке DB2:

1. Войдите в систему с помощью учетной записи локального администратора, созданной для установки DB2.
2. Если у вас есть диск DVD с продуктом баз данных DB2, вставьте его в дисковод. Если включена функция автоматического запуска, появится Панель запуска программы установки DB2. Если функция автоматического запуска не работает, откройте диск DVD продукта баз данных DB2 в Проводнике Windows и дважды щелкните по значку установки, чтобы запустить Панель установки DB2.

3. Если вы скачали продукт баз данных DB2 с Passport Advantage, запустите исполняемый файл, чтобы распаковать файлы установки продукта баз данных DB2. Используйте Проводник Windows для просмотра файлов установки DB2 и щелкните дважды по значку установки для запуска Панели установки DB2.
4. С Панели запуска установки DB2 можно просмотреть предварительные требования для установки, замечания по выпуску, а также перейти к процедуре установки. Ознакомьтесь со свежей информацией, приведенной в описании предварительных требований к установке и информации о выпуске.
5. Выберите **Установить продукт** - откроется окно Установить продукт со списком доступных для установки продуктов.

Если на данном компьютере не установлены продукты баз данных DB2, запустите установку, выбрав **Установить новую**. Выполните установку, следуя подсказкам мастера по установке DB2.

Если на данном компьютере установлен хотя бы один продукт баз данных DB2, можно:

- Выбрать **Установить новую**, чтобы создать новую копию DB2.
 - Выбрать **Работать с существующей**, чтобы обновить существующую копию DB2, добавить функции к существующей копии DB2, обновить существующую копию DB2 Версии 8, Версии 9.1 или Версии 9.5 или установить дополнительный продукт.
6. Мастер по установке DB2 определит, какой язык установлен в системе, и запустит программу установки на этом языке. Электронная справка поможет выполнить остальные действия. Для просмотра электронной справки нажмите кнопку **Справка** или клавишу **F1**. В любой момент можно нажать кнопку **Отмена**, чтобы прервать установку.

Результаты

Ваш продукт баз данных DB2 будет установлен по умолчанию в каталог *Program_Files\IBM\sql1lib*, где *Program_Files* - положение каталога Program Files.

Если при установке в системе этот каталог уже используется, будет использован путь установки продукта DB2 с добавлением *_xx*, где *xx* - две цифры, начиная с 01 по возрастанию, в зависимости от того, сколько копий DB2 у вас установлено.

Кроме того, вы можете задать свой собственный путь для установки продукта баз данных DB2.

Дальнейшие действия

- Проверьте свою установку.
- Выполните необходимые задачи после установки.

Информацию об ошибках, возникших во время установки, смотрите в файле журнала установки в каталоге Мои документы\DB2LOG\. Для имени этого файла журнала используется следующий формат: *DB2-КраткоеИмяПродукта-ДатаВремя.log*, например, *DB2-ESE-Tue Apr 04 17_04_45 2008.log*.

Если это новая установка продукта DB2 в 64-битной системе Vista, и вы используете 32-битный провайдер OLE DB, надо вручную зарегистрировать библиотеку DLL IBM DADB2. Чтобы зарегистрировать эту библиотеку DLL, введите команду:

```
c:\windows\SysWOW64\regsvr32 /s c:\Program_Files\IBM\SQLLIB\bin\ibmdadb2.dll
```

где *Program_Files* - положение каталога Program Files.

Чтобы предоставить продукту баз данных DB2 доступ к документации по DB2 на локальном компьютере или на другом компьютере в вашей локальной сети, нужно установить *Информационный центр DB2*. *Информационный центр DB2* содержит документацию по системе баз данных DB2 и связанным с ней продуктами DB2. По умолчанию, если *Информационный центр DB2* не установлен локально, система будет обращаться к информации о DB2 в Web.

Ограничения на память в DB2 Express Edition и DB2 Workgroup Server Edition

При установке DB2 Express Edition максимально допустимый объем памяти для экземпляра - 4 Гбайта.

При установке DB2 Workgroup Server Edition максимально допустимый объем памяти для экземпляра - 64 Гбайт.

Выделяемый для экземпляра объем памяти определяется параметром конфигурации менеджера баз данных **INSTANCE_MEMORY**.

Важные замечания при обновлении из Версии 9.1 или 9.5:

- Если объем памяти, сконфигурированный для продукта баз данных DB2 Версии 9.1, превышает допустимый предел, продукт DB2 после обновления в текущую версию может не запуститься.
- Менеджер автоматической настройки памяти не станет увеличивать общий предельный объем памяти экземпляров до значения, превышающего лицензионные ограничения.

Установка серверов DB2 при помощи мастера по установке DB2 (Linux и UNIX)

В этом разделе описана процедура запуска мастера по установке DB2 в операционных системах Linux и UNIX. Мастер по установке DB2 позволяет задать параметры установки и установить продукт баз данных DB2 в системе.

Прежде чем начать

Перед запуском мастера по установке DB2:

- Если вы собираетесь сконфигурировать среду многораздельных баз данных, посмотрите раздел “Задание среды многораздельных баз данных” в книге *Установка серверов DB2*
- Убедитесь, что в системе выполнены предварительные требования к установке и достаточно оперативной и дисковой памяти.
- Убедитесь, что у вас установлен поддерживаемый браузер (Firefox 2.0+, Mozilla 1.7+ или SeaMonkey 1.1.4).
- Установку сервера баз данных DB2 можно выполнять с полномочиями root или без полномочий root. Дополнительную информацию об установке без полномочий root смотрите в разделе “Обзор установки без полномочий root (Linux и UNIX)” в книге *Установка серверов DB2*.
- Образ продукта баз данных DB2 должен быть доступен. Вы можете получить образ установки DB2 либо купив физический диск DVD продукта баз данных DB2, либо скачав образ установки с Passport Advantage.
- Для установки неанглийской версии продукта баз данных DB2 нужны соответствующие пакеты для национальных языков.
- Мастер по установке DB2 - это программа установки с графическим интерфейсом. Для запуска мастера по установке DB2 на компьютере должна быть доступна система X windows. Убедитесь, что сервер X windows работает. Убедитесь, что вы правильно экспортировали дисплей. Например, `export DISPLAY=9.26.163.144:0`.

- Если в вашей среде применяется программное обеспечение систем защиты, перед запуском мастера по установке DB2 нужно вручную создать требуемых пользователей DB2.
- Если вы собираетесь использовать агент IBM Tivoli Monitoring for Databases: DB2, перед установкой продукта баз данных DB2 посмотрите в теме “Установка агента DB2 IBM Tivoli Monitoring for Databases при помощи программы установки DB2” подробности и ограничения.

Об этой задаче

Ограничения

- Под каждой учетной записи пользователя может работать только один экземпляр мастера по установке DB2.
- Возможности XML можно использовать только для базы данных, которая определена с кодовым набором UTF-8 и содержит только один раздел базы данных.
- В поля мастера по установке DB2 нельзя вводить символы национальных алфавитов.
- Для HP-UX 11i V2 в системах HP Integrity Series Systems на основе Itanium пользователи, созданные мастером по установке в качестве владельца экземпляра DB2, изолированного пользователя или DAS, недоступны по паролю, заданного в мастере по установке DB2. После завершения работы мастера надо перезадать пароль для этих пользователей. Это не повлияет на создание экземпляра или DAS в мастере по установке, то есть пересоздавать экземпляр или DAS не требуется.

Процедура

Для запуска мастера по установке DB2:

1. Если у вас есть физический диск DVD с продуктом баз данных DB2, перейдите в каталог, в котором смонтирован диск DVD продукта баз данных DB2, введя команду:

```
cd /dvdrom
```

где */dvdrom* - точка монтирования диска DVD продукта баз данных DB2.

2. Если образ продукта баз данных DB2 загружен по сети, нужно извлечь и распаковать файл tar продукта.

- a. Извлеките файл продукта:

```
gzip -d продукт.tar.gz
```

где *продукт* - имя загруженного вами продукта.

- b. Распакуйте файл tar продукта:

В операционных системах Linux

```
tar -xvf продукт.tar
```

В операционных системах AIX, HP-UX и Solaris

```
guntar -xvf продукт.tar
```

где *продукт* - имя загруженного вами продукта.

- c. Перейдите в другой каталог:

```
cd ./продукт
```

где *продукт* - имя загруженного вами продукта.

Примечание: Если вы скачали пакет для национальных языков, распакуйте его в тот же каталог. При этом будут созданы подкаталоги (такие как `./n1pack`) в одном каталоге, и программа установки сможет автоматически найти установочные образы, не выводя запрос пользователю.

3. Из каталога, где находится образ продукта баз данных, введите команду `./db2setup` для запуска мастера по установке DB2.
4. Откроется Панель запуска IBM DB2. В этом окне можно просмотреть предварительные требования для установки, информацию о выпуске, а также перейти к процедуре установки. Вы можете также ознакомиться со свежей информацией, приведенной в описании предварительных требований к установке и замечаниях по выпуску.
5. Выберите **Установить продукт** - откроется окно **Установить продукт** со списком доступных для установки продуктов.

Запустите установку, выбрав **Установить новый**. Выполните установку, следуя подсказкам мастера по установке DB2.

После запуска программы установки вам потребуется задать необходимые значения на панелях мастера по установке DB2. Выполните остальные действия, руководствуясь справкой по установке. Для просмотра справки нажмите кнопку **Справка** или клавишу F1. В любой момент можно нажать кнопку **Отмена**, чтобы прервать установку.

Результаты

Для установок без полномочий root продукты баз данных DB2 всегда устанавливаются в каталог `$HOME/sql1ib`, где `$HOME` представляет домашний каталог пользователя без полномочий root.

Для установок с полномочиями root продукты баз данных DB2 устанавливаются в один из следующих каталогов:

AIX, HP-UX и Solaris

`/opt/IBM/db2/V9.7`

Linux `/opt/ibm/db2/V9.7`

Если при установке в системе этот каталог уже используется, будет использован путь установки продукта баз данных DB2 с добавлением `_xx` где `xx` - две цифры, начиная с 01 по возрастанию, в зависимости от того, сколько копий DB2 у вас установлено.

Кроме того, вы можете задать свой собственный путь для установки продукта баз данных DB2.

К путям установки DB2 применяются следующие правила:

- Могут содержать латинские буквы нижнего регистра (a-z), верхнего регистра (A-Z) и символ подчеркивания (`_`)
- Не могут быть длиннее 128 символов
- Не могут содержать пробелы
- Не могут содержать символы национальных алфавитов
- Не могут быть символическими ссылками

При установке создаются следующие файлы журналов:

- Файл журнала установки DB2. В этот файл записывается вся информация об установке DB2, включая сообщения об ошибках.

- Для установок с полномочиями root имя файла журнала установки DB2 - `db2setup.log`.
- Для установок без полномочий root имя файла журнала установки DB2 - `db2setup_имя-пользователя.log`, где *имя-пользователя* - ID пользователя без полномочий root, под которым выполнена установка.
- Файл журнала ошибок DB2. В этот файл записываются все сообщения об ошибках, возвращаемых Java (например, исключительные ситуации и данные перехвата).
 - Для установок с полномочиями root имя файла журнала ошибок DB2 - `db2setup.err`.
 - Для установок без полномочий root имя файла журнала ошибок DB2 - `db2setup_имя-пользователя.err`, где *имя-пользователя* - ID пользователя без полномочий root, под которым выполнена установка.

По умолчанию эти файлы журналов находятся в каталоге `/tmp`. Можно задать другое положение файлов журналов.

Файл `db2setup.his` более не используется. Вместо него программа установки DB2 сохраняет копию файла журнала установки DB2 в каталоге `DB2_DIR/install/logs/` и переименовывает ее в `db2install.history`. Если файл с таким именем уже существует, программа установки DB2 переименовывает файл журнала в `db2install.history.xxxx`, где `xxxx` - число от 0000 до 9999, в зависимости от числа установок на данном компьютере.

У каждой копии установки отдельный список файлов хронологии. Если копия установки удаляется, также удаляются и файлы хронологии в ее пути установки. Операция копирования файла журнала в файл хронологии выполняется в конце процесса установки; если программа установки остановлена или прервана до завершения процесса установки, файл хронологии не создается.

Дальнейшие действия

- Проверьте свою установку.
- Выполните необходимые задачи после установки.

Пакет для национальных языков можно установить также, введя команду `./db2setup` из каталога, в котором находится пакет, после того, как установлен продукт баз данных DB2.

В Linux x86, чтобы предоставить продукту баз данных DB2 доступ к документации по DB2 на локальном компьютере или на другом компьютере в вашей локальной сети, нужно установить *Информационный центр DB2*. *Информационный центр DB2* содержит документацию по системе баз данных DB2 и связанным с ней продуктами DB2.

Ограничения на память в DB2 Express Edition и DB2 Workgroup Server Edition

При установке DB2 Express Edition максимально допустимый объем памяти для экземпляра - 4 Гбайта.

При установке DB2 Workgroup Server Edition максимально допустимый объем памяти для экземпляра - 64 Гбайт.

Выделяемый для экземпляра объем памяти определяется параметром конфигурации менеджера баз данных `INSTANCE_MEMORY`.

Важные замечания при обновлении из Версии 9.1 или 9.5:

- Если объем памяти, сконфигурированный для продукта баз данных DB2 Версии 9.1 или 9.5, превышает допустимый предел, продукт DB2 после обновления в текущую версию может не запуститься.
- Менеджер автоматической настройки памяти не станет увеличивать общий предельный объем памяти экземпляров до значения, превышающего лицензионные ограничения.

Установка и конфигурирование DB2 Text Search при помощи файла ответов

Установить и сконфигурировать DB2 Text Search можно в составе пользовательской установки продукта баз данных DB2 в режиме без вывода сообщений. При установке этого типа используется команда **setup** или **db2setup** с файлом ответов.

Об этой задаче

Чтобы установить DB2 Text Search, выполните пользовательскую установку продукта баз данных DB2. Для установки и конфигурирования DB2 Text Search в файл ответов нужно добавить ряд ключевых слов.

Процедура

Чтобы выполнить пользовательскую установку:

1. Добавьте в файл ответов, используемый для установки продукта баз данных DB2, следующую строку:

```
COMP = TEXT_SEARCH
```

2. Чтобы сконфигурировать DB2 Text Search во время установки, добавьте в файл ответов следующую строку:

- Только для установок с полномочиями root:

```
имя_экземпляра_db2.TEXT_SEARCH_HTTP_SERVICE_NAME = имя_экземпляра_db2_db2j
```

где *имя_экземпляра_db2* - имя экземпляра DB2, а *имя_экземпляра_db2_db2j* - имя службы.

- Для установок с полномочиями root и без полномочий root:

```
имя_экземпляра_db2.TEXT_SEARCH_HTTP_PORT_NUMBER = номер-порта
```

Если задать значение для ключевого слова **TEXT_SEARCH_HTTP_SERVICE_NAME** при установке без полномочий root, будет возвращена ошибка.

Можно указать любые допустимые свободные имя и номер порта. Если не указать никаких значений, для конфигурации будут использованы значения по умолчанию при условии, что для ключевого слова файла ответов *имя_экземпляра_db2*.**CONFIGURE_TEXT_SEARCH** задано значение YES.

3. Установите продукт баз данных DB2, выполнив инструкции для используемой платформы:
 - "Установка продукта DB2 при помощи файла ответов (Windows)" в Установка серверов DB2
 - "Установка продукта (DB2 при помощи файла ответов (Linux и UNIX))" в Установка серверов DB2

Дальнейшие действия

Конфигурировать возможность DB2 Text Search сразу же после ее установки не обязательно; ее можно сконфигурировать и позднее. Инструкции по конфигурированию после установки смотрите в разделе “Конфигурирование текстового поиска DB2” на стр. 42.

Установка продукта DB2 при помощи файла ответов (Windows)

Прежде чем начать

Перед началом установки убедитесь, что:

- В системе выполнены требования к памяти, аппаратному обеспечению и программному обеспечению, предъявляемые продуктом DB2.

Об этой задаче

Для установки продукта DB2 с той рабочей станции, на которой он будет установлен:

Процедура

1. Соединитесь с совместно используемым каталогом на сетевом диске или диске DVD, где находятся установочные файлы DB2, введя в командной строке команду:

```
net use x: \\имя_комп\имя_к-га_совм_доступа /USER:домен\имя_польз
```

где:

- *x*: - локальное обозначение совместно используемого каталога.
- *имя_комп* - имя удаленного сервера, где находятся установочные файлы DB2.
- *имя_к-га_совм_доступа* - имя совместно используемого каталога на сетевом диске или диске DVD, где находятся установочные файлы DB2.
- *домен* - домен, где задана учетная запись.
- *имя_польз* - пользователь, имеющий доступ к этой системе.

Например, чтобы присвоить локальному диску *x*: удаленный каталог *db2prods*, который используется совместно под именем *DB2_Installation_Images* и расположен на удаленном сервере *Server*, введите следующую команду:

```
net use x: \\Server\DB2_Installation_Images
```

В зависимости от того, как настроена сетевая защита, может понадобиться задать параметр */USER*.

2. Введите команду **setup**:

```
setup -u файл_ответов
```

где *файл_ответов* - полный путь и имя используемого файла ответов.

Если вы используете файл ответов, созданный генератором файла ответов, все профили экземпляра должны быть расположены на том же диске и в том же каталоге, что и указанный файл ответов.

3. После завершения установки просмотрите сообщения в файле журнала. Информацию об ошибках, возникших во время установки, смотрите в файле журнала установки в каталоге Мои документы\DB2LOG\. Для имени этого файла журнала используется следующий формат: DB2-КраткоеИмяПродукта-ДатаВремя.log, например, DB2-ESE-Tue Apr 04 17_04_45 2007.log.

Дальнейшие действия

Чтобы предоставить продукту DB2 доступ к документации по DB2 на локальном компьютере или на другом компьютере в вашей локальной сети, нужно установить Информационный центр DB2.

Установка продукта баз данных DB2 при помощи файла ответов (Linux и UNIX)

Прежде чем начать

Перед началом установки убедитесь, что:

- Для установки с полномочиями root зарегистрируйтесь под ID пользователя с привилегиями root. Для установки без полномочий root зарегистрируйтесь под ID пользователя, который будет владельцем этой установки DB2.
- В системе выполнены требования к памяти, аппаратному обеспечению и программному обеспечению, предъявляемые продуктом баз данных DB2.
- Установку продукта баз данных DB2 с помощью файла ответов можно выполнять с полномочиями root или без полномочий root.
- Остановлены все процессы DB2, связанные с копией DB2, с которой вы работаете.

Процедура

Чтобы выполнить установку с помощью файла ответов:

1. Введите команду **db2setup**:

```
dvd/db2setup -r каталог_файла_ответов/файл_ответов
```

где:

- *dvd* - положение установочного образа DB2;
 - *каталог_файла_ответов* - каталог, где расположен настроенный файл ответов; и
 - *файл_ответов* - имя файла ответов.
2. После завершения установки просмотрите сообщения в файле журнала. Журналы установки расположены по умолчанию в каталоге /tmp:
 - *db2setup.log* (*db2setup_имя_пользователя.log* для установок без полномочий root, где *имя_пользователя* - ID пользователя без полномочий root, под которым выполнялась установка)
 - *db2setup.err* (*db2setup_имя_пользователя.err* для установок без полномочий root, где *имя_пользователя* - ID пользователя без полномочий root, под которым выполнялась установка)

Можно задать другое положение файла журнала. Журнал хронологии установок *db2install.history* находится в каталоге *DB2DIR/install/logs*, где *DB2DIR* задает путь установки продукта DB2. Каталог установки по умолчанию:

- Для AIX, HP-UX и Solaris - /opt/IBM/db2/V9.7
- Для Linux - /opt/ibm/db2/V9.7

Если в одном положении установлено несколько продуктов баз данных DB2, у этого журнала будет имя *db2install.history.xxxx*, где *xxxx* - число, начиная с 0001, которое увеличивается в зависимости от того, сколько копий DB2 установлено.

Дальнейшие действия

Чтобы предоставить продукту баз данных DB2 доступ к документации по DB2 на локальном компьютере или на другом компьютере в вашей локальной сети, нужно установить *Информационный центр DB2*. *Информационный центр DB2* содержит документацию по базам данных DB2 и связанными с ними продуктами.

Установка текстового поиска DB2 при помощи `db2_install` (Linux и UNIX)

При вводе команды `db2_install` устанавливается также и DB2 Text Search.

Об этой задаче

Чтобы установить DB2 Text Search, выполните действия, описанные в теме "Установка продукта DB2 при помощи `db2_install`", *Установка серверов DB2*. Возможность DB2 Text Search будет автоматически установлена в составе установки продукта баз данных DB2.

В случае установки без полномочий root будет создан экземпляр DB2 и установлена возможность DB2 Text Search. В случае установки с полномочиями root нужно будет создать экземпляр DB2 и сконфигурировать DB2 Text Search одним из доступных способов.

Конфигурировать возможность DB2 Text Search сразу же после ее установки не обязательно. Инструкции по выполнению конфигурирования смотрите в разделе "Конфигурирование текстового поиска DB2" на стр. 42.

Установка продукта баз данных DB2 при помощи команды `db2_install` или `doce_install` (Linux и UNIX)

Чтобы установить продукты и возможности баз данных DB2 или Информационный центр DB2, извлеките образ продукта, если он находится в сжатом файле, и введите команду `db2_install` или `doce_install`.

Прежде чем начать

Перед установкой продуктов и возможностей баз данных DB2 или *Информационного центра DB2*:

- Необходимо ознакомиться с документацией по установке того продукта баз данных DB2, который вы собираетесь устанавливать. Например, если вы планируете установить DB2 Enterprise Server Edition, ознакомьтесь с документом *Установка серверов DB2*, в котором приведены предварительные требования и другая полезная информация об установке.
- Продукт баз данных DB2 и *Информационный центр DB2* можно устанавливать как с полномочиями, так и без полномочий пользователя root.
- Образ продукта баз данных DB2 должен быть доступен. Вы можете получить образ установки DB2 либо купив физический диск DVD продукта баз данных DB2, либо скачав образ установки с Passport Advantage.

Об этой задаче

Команда `db2_install` устанавливает продукты и возможности баз данных DB2 в поддерживаемых операционных системах Linux и UNIX.

Команда **doce_install** устанавливает *Информационный центр DB2* в поддерживаемых операционных системах Linux.

Ограничения

Продукты или возможности баз данных DB2 *нельзя* установить вручную с помощью собственных средств установки операционной системы, таких как **rpm**, **SMIT**, **swinstall** или **pkgadd**. Потребуется изменить все существующие сценарии, содержащие собственные утилиты установки, которые вы используете для взаимодействия и запросов к установкам баз данных DB2.

Команда **db2_install** не поддерживается для DVD-диска с пакетом национальных языков.

В системе можно установить только одну копию *Информационного центра DB2* для текущего выпуска. Информационный центр нельзя установить в то же положение, в которое установлен продукт баз данных DB2. Если вы устанавливаете *Информационный центр DB2* в системе с брандмауэром и собираетесь разрешить остальным системам доступ к *Информационному центру DB2*, необходимо открыть порт при установке брандмауэра.

Процедура

Чтобы установить продукт баз данных или утилиту DB2 при помощи команды **db2_install** или установить *Информационный центр DB2* с использованием команды **doce_install**:

1. Если у вас есть физический диск DVD с продуктом баз данных DB2, вставьте и смонтируйте необходимый диск DVD или перейдите в файловую систему, в которой был сохранен образ установки.
2. Если образ продукта баз данных DB2 загружен по сети, нужно извлечь и распаковать файл tar продукта.

- a. извлеките файл продукта:

```
gzip -d продукт.tar.gz
```

Например,

```
gzip -d ese.tar.gz
```

- b. Распакуйте файл tar продукта:

В операционных системах Linux

```
tar -xvf продукт.tar
```

Например,

```
tar -xvf ese.tar
```

В операционных системах AIX, HP-UX и Solaris

```
guntar -xvf продукт.tar
```

Например,

```
guntar -xvf ese.tar
```

- c. Перейдите в каталог продукта:

```
cd продукт
```

Например,

```
cd ese
```

3. Введите команду **./db2_install** или **./doce_install**:

```
./db2_install -b DB2DIR -p короткое_имя_продукта -с положение_языков_пакета  
-L язык... -n
```

где:

- *DB2DIR* задает путь установки продукта баз данных DB2. Если путь не указан, вам будет предложено либо выбрать путь по умолчанию, либо ввести необходимый путь. Путь установки по умолчанию:
 - для операционных систем AIX, HP-UX или Solaris: /opt/IBM/db2/V9.7
 - для операционных систем Linux: /opt/ibm/db2/V9.7
 - для *Информационного центра DB2*: /opt/ibm/db2ic/V9.7. У Версия Информационного центра DB2 для рабочей станции нет пути установки по умолчанию, и положение установки должны задать вы. Однако по умолчанию Версия Информационного центра DB2 для рабочей станции устанавливается на порт 51097.

Если вы задаете свой собственный путь, необходимо указывать полное имя.

К путям установки DB2 применяются следующие правила:

- Могут содержать латинские буквы нижнего регистра (a-z), верхнего регистра (A-Z) и символ подчеркивания (_)
- Не могут быть длиннее 128 символов
- Не могут содержать пробелы
- Не могут содержать символы национальных алфавитов
- Не могут быть символическими ссылками

Примечание: Чтобы продукты и компоненты баз данных DB2 работали совместно, их нужно установить в один путь. Продукты баз данных DB2 можно устанавливать и в разные пути. Но чтобы продукты и компоненты работали совместно, они должны быть установлены в один путь и должны быть одного уровня выпуска.

- *короткое_имя_продукта* задает устанавливаемый продукт баз данных DB2. Этот параметр регистронезависим и обязателен, если указан параметр **-n**. Короткое имя продукта (*productShortName*) можно найти в файле ComponentList.htm (под полным именем продукта), находящемся на вашем носителе в каталоге /db2/платформа, где платформа - платформа, на которой выполняется установка. Можно установить только один продукт за раз.
- *положение_пакета_национальных_языков* задает положение пакета национальных языков (NLPACK).
- *язык* задает поддержку национального языка. Можно установить версию продукта баз данных DB2, отличную от английской. Однако эту команду необходимо запускать с DVD-диска с продуктом, а не с DVD-диска с пакетом национальных языков.

По умолчанию английский язык устанавливается всегда, поэтому указывать его не надо. Если требуется несколько языков, этот параметр обязателен. Чтобы задать несколько языков, укажите этот параметр несколько раз. Например, чтобы установить французский и русский языки, задайте **-L FR -L RU**.

- Параметр *n* задает неинтерактивный режим установки. Если указан этот параметр, необходимо также указать **-b** и **-p**. Параметры **-с** и **-L** задаются только в случае необходимости.

Если при установке *Информационного центра DB2* указать номер порта, отличный от используемого по умолчанию, можно получить сообщение об ошибке. Указанное имя службы уже используется. Эту ошибку можно исправить либо выбрав номер порта по умолчанию, либо указав другое имя службы.

Дальнейшие действия

После установки нужно вручную сконфигурировать сервер баз данных DB2. При этом нужно создать и сконфигурировать пользователей и экземпляры.

Конфигурирование текстового поиска DB2

Опции для конфигурирования возможности DB2 Text Search зависят от того, выполняется ли исходное конфигурирование или переконфигурирование, и от используемой платформы.

Прежде чем начать

Перед переконфигурированием DB2 Text Search остановите службы экземпляра текстового поиска, как описано в разделе “Запуск службы экземпляра текстового поиска DB2” на стр. 51.

Процедура

- Определите, сконфигурирована ли возможность DB2 Text Search.
Запустите инструмент конфигурирования, введя следующую команду:

```
configTool printAll -configPath абсолютный-путь-к-папке-config
```

В выводе опции **printAll**, если возможность DB2 Text Search не сконфигурирована, маркер аутентификации - это пустая строка.
- Выполните исходное конфигурирование возможности DB2 Text Search.
В операционных системах Linux и UNIX используйте для конфигурирования DB2 Text Search один из следующих способов:
 - Повторите установку без вывода сообщений, как описано в разделе “Установка и конфигурирование DB2 Text Search при помощи файла ответов” на стр. 36.
 - Повторите установку из графического пользовательского интерфейса, как описано в разделе “Установка и конфигурирование текстового поиска DB2 при помощи мастера по установке DB2” на стр. 28.
 - Используйте инструмент конфигурирования. Смотрите раздел “Конфигурирование DB2 Text Search при помощи инструмента конфигурирования” на стр. 43. Учтите, что при использовании инструмента конфигурирования для выполнения установки вручную все параметры нужно сконфигурировать вручную, тогда как при использовании программы установки нужно сконфигурировать всего два параметра.
 - Введите команду **db2isetup**. Учтите, что в операционных системах UNIX, если экземпляр создается при помощи **db2icrt**, DB2 Text Search не конфигурируется автоматически.

В операционных системах Windows используйте для конфигурирования возможности DB2 Text Search один из следующих способов:

 - Повторите установку без вывода сообщений, как описано в разделе “Установка и конфигурирование DB2 Text Search при помощи файла ответов” на стр. 36.
 - Повторите установку из графического пользовательского интерфейса, как описано в разделе “Установка и конфигурирование текстового поиска DB2 при помощи мастера по установке DB2” на стр. 28.
 - Введите команду **db2icrt**, **db2iupdt** или **db2iupgrade**.
- Определите, используется ли Java Developer Kit разработки IBM.
DB2 Text Search использует для внутренней обработки Java Developer Kit, на положение которого указывает параметр **JDK_PATH** команды `db2 get dbm cfg`, и

этот Java Developer Kit должен быть разработки IBM. Чтобы проверить, используется ли Java Developer Kit разработки IBM, введите следующую команду:
JDK_PATH/jre/bin/java -version

Эта команда выведет информацию о версии Java, и если используется Java Developer Kit разработки IBM, в строке будет указана IBM.

- Переконфигурируйте DB2 Text Search.

Для переконфигурирования уже установленной возможности DB2 Text Search программу установки из графического интерфейса использовать нельзя. Все изменения в конфигурацию нужно вносить вручную.

В операционных системах Linux и UNIX для переконфигурирования возможности DB2 Text Search используйте один из следующих способов:

- Повторите установку без вывода сообщений, как описано в разделе “Установка и конфигурирование DB2 Text Search при помощи файла ответов” на стр. 36.
- Используйте инструмент конфигурирования. Смотрите раздел “Конфигурирование DB2 Text Search при помощи инструмента конфигурирования”.
- Введите команду **db2isetup** с опцией **-r**.

В операционных системах Windows для переконфигурирования возможности DB2 Text Search используйте один из следующих способов:

- Повторите установку без вывода сообщений, как описано в разделе “Установка и конфигурирование DB2 Text Search при помощи файла ответов” на стр. 36.
- Используйте инструмент конфигурирования. Смотрите раздел “Конфигурирование DB2 Text Search при помощи инструмента конфигурирования”.
- Введите команду **db2icrt**, **db2iupdt** или **db2iupgrade**, задав, как показано, опцию **-j** нужным вам образом:
 - Команда **-j "TEXT_SEARCH"** пытается сконфигурировать DB2 Text Search с именем службы по умолчанию и сгенерированным значением для номера порта.
 - Команда **-j "TEXT_SEARCH, [имя_службы]"** резервирует указанное имя службы с автоматически сгенерированным номером порта или с одним и тем же номером порта, назначенным для этого имени службы, если имя службы уже зарезервировано в файле служб.
 - Команда **-j "TEXT_SEARCH, [номер_порта]"** резервирует порт с именем службы по умолчанию.
 - Команда **-j "TEXT_SEARCH, [имя_службы], [номер_порта]"** резервирует указанные имя службы и номер порта.

Примечание: В операционных системах Windows параметр **PATH** в командном окне DB2 указывает на *путь-установки-текущей-копии-по-умолчанию* \db2tss\bin, поэтому, чтобы сконфигурировать экземпляр, находящийся не в текущей копии DB2, переключитесь сначала на командное окно DB2, соответствующее нужной копии.

Конфигурирование DB2 Text Search при помощи инструмента конфигурирования

Инструмент конфигурирования - это инструмент командной строки, который можно использовать для создания исходной конфигурации DB2 Text Search или для изменения текущей конфигурации.

Прежде чем начать

Для настройки большинства параметров конфигурации требуется остановить службы экземпляра DB2 Text Search.

Об этой задаче

Исходное конфигурирование рекомендуется выполнять автоматически программой установки DB2. Но если вы хотите переконфигурировать DB2 Text Search вручную, это легко сделать при помощи инструмента конфигурирования.

Процедура

Чтобы создать исходную конфигурацию DB2 Text Search:

1. Введите команду **configTool** с опцией **configureParams**, чтобы задать значения для пути конфигурации, пути журнала, пути временного каталога и пути установки.

- Задайте опции следующим образом:

```
configTool configureParams
-configPath абсолютный-путь-к-папке-config
-logPath путь
-tempDirPath путь
-installPath путь
-maxHeapSize значение
```

Где **logPath** задает используемый DB2 Text Search путь журнала, **tempDirPath** задает используемый DB2 Text Search путь временного каталога, **installPath** задает путь к каталогу установки текстового поиска DB2 (а именно, DB2PATH\db2tss в Windows или каталог *DB2DIR/db2tss* в Linux и UNIX), а **maxHeapSize** задает размер используемой кучи. *DB2DIR* - это положение копии DB2.

Примечание: Начиная с DB2 Версии 9.7 Fix Pack 4, параметр **maxHeapSize** переименован в **startupHeapSize**.

- В операционных системах Windows введите команду указанным ниже способом. Необходимо задать только **configPath**; все остальные параметры - это пути и значения, назначаемые по умолчанию.

```
configTool
-configPath абсолютный-путь-к-папке-config
```

2. Сгенерируйте маркер аутентификации, введя команду **configTool** с опцией **generateToken** следующим образом:

```
configTool generateToken
-configPath абсолютный-путь-к-папке-config
-seed имя-моего-экземпляра
```

DB2 Text Search выполняет аутентификацию требований администрирования индексов текстового поиска и требований текстового поиска при помощи маркера аутентификации. Запоминать маркер или ключ аутентификации не требуется, поскольку службы экземпляра текстового поиска используют его во внутренней обработке.

3. Задайте порт HTTP, введя команду **configTool** с опцией **configureHTTPListener** следующим образом:

```
configTool configureHTTPListener
-configPath абсолютный-путь-к-папке-config
-adminHTTPPort номер-порта
```

Примечание: Можно использовать значение номера порта от 1024 до 65535. Управляющий порт HTTP позволяет устанавливать связь между процессами текстового поиска при помощи TCP/IP. При установке продукта баз данных DB2 или создании экземпляра, если у вас есть полномочия root, можно задать имя службы и порт. Они будут использоваться для обновления файла служб.

4. Обновите файл служб.

Смотрите раздел “Обновление файла служб на сервере для соединений TCP/IP”.

Если для конфигурирования используется инструмент конфигурирования, он не обновляет файл служб. Поэтому вы должны обновить файл служб вручную.

Примечание: Обновлять файл служб могут только пользователи root. Пользователи без полномочий root должны сначала попросить администратора системы ввести команду **db2rfe**.

Обновление файла служб на сервере для соединений TCP/IP

Эта задача - часть большей задачи *Конфигурирование связи TCP/IP для экземпляра DB2*.

Об этой задаче

В файле служб TCP/IP заданы номера портов, которые могут применяться программами серверов для получения запросов от клиентов. Если в поле *svcsname* файла конфигурации менеджера баз данных было задано имя службы, то в файле служб необходимо задать номер порта или протокол, соответствующий этому имени службы. Если в поле *svcsname* файла конфигурации менеджера баз данных был задан номер порта, то файл служб обновлять *не* нужно.

Укажите в файле *services* порты, через которые сервер принимает входящие требования от клиентов. Положение по умолчанию файла *services* зависит от операционной системы:

Операционные системы Linux и UNIX

`/etc/services`

Операционные системы Windows

`%SystemRoot%\system32\drivers\etc\services`

Процедура

С помощью текстового редактора добавьте запись для соединения в файл служб. Например:

```
db2c_db2inst1 3700/tcp # порт службы соединения DB2
```

где:

db2c_db2inst1

имя службы соединения

3700

номер порта соединения

tcp

используемый протокол связи

Результаты

Глава 4. Обновление текстового поиска

Обновление DB2 Text Search для установок от имени администратора или пользователя root

Чтобы получить самые последние функциональные возможности, обновите свой экземпляр DB2 Text Search. При обновлении экземпляра текстового поиска вам нужно обновить сервер DB2, экземпляр DB2 и все базы данных.

Об этой задаче

Перечисленные ниже действия представляют процесс обновления установок от имени пользователя root DB2 Text Search Версии 9.5 или Версии 9.7 для платформ Linux или UNIX или от имени администратора для платформы Windows.

Процедура

1. Создайте резервные копии всех конфигурируемых свойств DB2 Text Search и серверов баз данных. Используйте следующие команды:

```
configTool printAll -configPath каталог-конфигурации > db2tss_config.out
adminTool version -configPath каталог-конфигурации >> db2tss_config.out
```
2. Если для DB2 Text Search включена поддержка документов в расширенном текстовом формате и вы выполняете обновление из DB2 Версии 9.5 Fix Pack 3 или более новой, отключите поддержку документов в расширенном текстовом формате. Введите команду:

```
richtextTool disable <каталог-установки-DB2>
```
3. Войдите как пользователь - владелец экземпляра или пользователь с полномочиями SYSADM.
4. Остановите экземпляр DB2 и службу экземпляра DB2 Text Search.
5. Создайте резервную копию каталога конфигурации DB2 Text Search в папке *INSTHOME/db2tss/config*.
6. Войдите на сервер DB2 как пользователь с полномочиями root в операционных системах Linux и UNIX или с полномочиями локального администратора в операционных системах Windows. Если вы обновляете многораздельный экземпляр, обновление экземпляра надо выполнять из раздела - владельца экземпляра.
7. Установите новую копию той версии DB2, до которой вы проводите обновление, и выполните пользовательскую установку. Убедитесь, что выбран компонент текстового поиска. Текстовый поиск DB2 - это дополнительный компонент, который доступен только при выборе пользовательской установки.
8. Обновите ваши экземпляры, введя команду **db2iupgrade**:

```
db2 iupgrade /j "text_search [[,имя-службы]] [[,номер-порта]]"
```
9. Обновите существующие базы данных с помощью команды **DB2 UPGRADE DATABASE**.
10. Обновите сервер DB2 Text Search для ваших экземпляров с помощью команды **configTool upgradeInstance**.
 - Для операционных систем Linux и UNIX:

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

где *INSTHOME* - домашний каталог экземпляра, а *DB2DIR* - положение копии DB2 Версии 9.7.

- Для операционных систем Windows:

```
configTool upgradeInstance
  -installedConfigPath "<DB2PATH>\CFG\DB2TSS\CONFIG"
  -configPath "<INSTPROFDIR>\имя-экземпляра\DB2TSS\CONFIG"
```

, где *DB2PATH* - положение самой новой копии DB2, а *INSTPROFDIR* - каталог профиля экземпляра.

11. Просмотрите значения всех конфигурируемых свойств DB2 Text Search. Сравните их со значениями, для которых было выполнено резервное копирование, и убедитесь, что все значения правильны. Введите следующую команду для проверки значений конфигурации:

```
configTool printAll -configPath <каталог-конфигурации>
```

12. Начиная с DB2 Версии 9.7 Fix Pack 4, необходимо сконфигурировать пакет для поддержки расширенного текстового или проприетарного формата.
 - a. Скачайте и установите пакет DB2 Accessories Suite (Oracle's Outside In Technology (OIT)). Дополнительную информацию смотрите в теме об установке DB2 Accessories Suite.
 - b. Запустите утилиту **richtextTool** для конфигурирования поддержки документов в расширенном текстовом формате от имени системного администратора или пользователя с разрешениями администратора.

13. Если ранее вы отключили поддержку расширенных текстовых документов для DB2 Text Search, сконфигурируйте и включите такую поддержку сейчас. Используйте следующие команды для конфигурирования и включения поддержки документов в расширенном текстовом формате:

```
richtextTool setup <DB2DIR> <каталог-файлов-zip-OIT>
richtextTool enable <DB2DIR>
```

, где *DB2DIR* - положение копии DB2 Версии 9.7, а *каталог-файлов-zip-OIT* - каталог, содержащий сжатые файлы zip библиотеки OIT (фильтр OIT).

14. Проверьте, успешно ли выполнено обновление, запустив службу экземпляра DB2 Text Search. Если вы отключали поддержку текстового поиска, проверьте, что теперь поддержка документов в расширенном текстовом формате включена, запустив запросы текстового поиска и сравнив результаты с полученными до обновления.

Обновление установки DB2 Text Search от имени рядового пользователя (не root) (Linux и UNIX)

Чтобы получить самые последние функциональные возможности, обновите свой экземпляр DB2 Text Search. При обновлении экземпляра текстового поиска вам нужно обновить сервер DB2, экземпляр DB2 и все базы данных.

Об этой задаче

Перечисленные ниже действия представляют процесс обновления установок от имени рядового пользователя (не root) DB2 Text Search Версии 9.5 или Версии 9.7 для платформ Linux или UNIX.

Процедура

1. Создайте резервные копии всех конфигурируемых свойств DB2 Text Search и серверов баз данных. Можно использовать две следующих команды:


```
configTool printAll -configPath каталог-конфигурации > db2tss_config.out
adminTool version -configPath каталог-конфигурации >> db2tss_config.out
```

2. Если для DB2 Text Search включена поддержка документов в расширенном текстовом формате и вы выполняете обновление из DB2 Версии 9.5 Fix Pack 3 или более новой, отключите поддержку документов в расширенном текстовом формате. Введите команду:

```
richtextTool disable <каталог-установки-DB2>
```
3. Войдите как пользователь - владелец экземпляра или пользователь с полномочиями SYSADM. Затем остановите экземпляр DB2 и службу экземпляра DB2 Text Search.
4. Создайте резервную копию каталога конфигурации DB2 Text Search в папке `sqllib/db2tss/config`.
5. Зарегистрируйтесь на сервере DB2 как пользователь без полномочий root. Проверьте тип экземпляра базы данных, чтобы убедиться, можно ли ее обновлять как установку от имени рядового пользователя (не root).
6. Установите новую копию DB2 и выберите опцию обновления. Если вы задали параметр **-f nobackup** и установка продукта базы данных DB2 не удалась, вручную установите продукт базы данных DB2 и запустите команду **db2nrupgrade** для обновления экземпляра пользователя без полномочий root. Из дерева возможностей вы должны выбрать компонент DB2 Text Search. Введите команду:

```
db2nrupg -b <каталог-резервной-копии>
```

для установки от имени рядового пользователя (не root) до обновления.
каталог-резервной-копии - каталог резервной копии для файла конфигурации.

7. Обновите существующие базы данных с помощью команды **upgrade database**.
8. Включите возможности, доступные для пользователя root. Вы можете попросить системного администратора с правами доступа root ввести команду **db2rfe**
9. Обновите сервер DB2 Text Search для ваших экземпляров с помощью команды **configTool upgradeInstance**.

- Для операционных систем Linux и UNIX:

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

где *INSTHOME* - домашний каталог экземпляра, а *DB2DIR* - положение копии DB2 Версии 9.7.

10. Просмотрите значения всех конфигурируемых свойств DB2 Text Search. Сравните их со значениями, для которых было выполнено резервное копирование, и убедитесь, что все значения правильны. Введите следующую команду для проверки значений конфигурации:

```
configTool printAll -configPath <каталог-конфигурации>
```
11. Начиная с DB2 Версии 9.7 Fix Pack 4, необходимо сконфигурировать пакет для поддержки расширенного текстового или проприетарного формата.
 - a. Скачайте и установите пакет DB2 Accessories Suite (Oracle's Outside In Technology (OIT)). Дополнительную информацию смотрите в теме об установке DB2 Accessories Suite.
 - b. Запустите утилиту **richtextTool** для конфигурирования поддержки документов в расширенном текстовом формате от имени системного администратора или пользователя с разрешениями администратора.
12. Если ранее вы отключили поддержку расширенных текстовых документов для DB2 Text Search, сконфигурируйте и включите такую поддержку сейчас.

Используйте следующие команды для конфигурирования и включения поддержки документов в расширенном текстовом формате:

```
richtextTool setup <DB2DIR> <каталог-файлов-zip-OIT>  
richtextTool enable <DB2DIR>
```

, где *DB2DIR* - положение копии DB2 Версии 9.7, а *каталог-файлов-zip-OIT* - каталог, содержащий сжатые файлы zip библиотеки OIT (фильтр OIT).

13. Проверьте, успешно ли выполнено обновление, запустив службу экземпляра DB2 Text Search. Если вы отключали поддержку текстового поиска, проверьте, что теперь поддержка документов в расширенном текстовом формате включена, запустив запросы текстового поиска и сравнив результаты с полученными до обновления.

Глава 5. Администрирование и обслуживание индексов текстового поиска

Запуск службы экземпляра текстового поиска DB2

Чтобы включить для базы данных текстовый поиск, сначала нужно запустить службу экземпляра DB2 Text Search.

Об этой задаче

Чтобы запустить службу экземпляра DB2 Text Search, введите следующую команду:
`db2ts "START FOR TEXT"`

При вызове команды `db2ts START FOR TEXT` будет запущена служба обновления индексов.

Включение базы данных для DB2 Text Search

Включение надо выполнить для каждой базы данных, содержащей столбцы текста, в котором будет нужен поиск. Включить базу данных для DB2 Text Search можно командой `db2ts ENABLE DATABASE FOR TEXT` или хранимой процедурой `SYSPROC.SYSTS_ENABLE`.

Прежде чем начать

Нужно запустить службу экземпляра DB2.

ID авторизации оператора должен обладать полномочиями DBADM или DATAACCESS.

Об этой задаче

Включив базу данных, вы сможете использовать следующие производные таблицы для получения информации об индексах текстового поиска в базе данных и о свойствах этих индексов:

SYSIBMTS.TSDEFAULTS

Содержит значения по умолчанию для характеристик индекса, текста и обработки

SYSIBMTS.TSLOCKS

Содержит информацию о блокировках команд, заданных на уровне базы данных и на уровне индекса

SYSIBMTS.TSINDEXES

Содержит все индексы текстового поиска и значения их параметров

SYSIBMTS.TSCONFIGURATION

Содержит параметры конфигурации индекса

SYSIBMTS.TSCOLLECTIONNAMES

Содержит имена собраний для каждого индекса

После включения базы данных для текстового поиска она останется включенной, пока вы не выключите ее явным образом.

Чтобы подготовить базу данных к использованию с DB2 Text Search, воспользуйтесь одним из следующих способов:

- Введите команду:

```
db2ts "ENABLE DATABASE FOR TEXT CONNECT TO имя-базы-данных"
```

Это действие нужно выполнить только один раз для каждой базы данных. Не нужно включать базу данных каждый раз, когда вы останавливаете и перезапускаете службы экземпляра.

Например, чтобы включить базу данных с именем SAMPLE, введите следующую команду:

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

- Вызовите одну из подпрограмм SQL администрирования, как показано ниже:
 - CALL SYSPROC.SYSTS_ADMIN_CMD ('ENABLE DATABASE FOR TEXT', 'en_US', ?)
 - CALL SYSPROC.SYSTS_ENABLE('en_US', ?)

Установка DB2 Accessories Suite для DB2 Text Search Версии 9.7 FP3 или более ранних версий

DB2 Accessories Suite предоставляет библиотеки фильтров, требуемые для функциональности поддержки документов в расширенном текстовом формате для DB2 Text Search.

Прежде чем начать

Для установки DB2 Accessories Suite необходимо:

- Скачать DB2 Accessories Suite (V8.2.2 of Oracle's Outside In Technology Search Export SDK и HTML Export SDK). Ссылку для скачивания смотрите в теме <https://www.ibm.com/services/forms/preLogin.do?source=swg-dm-db2accsuite>

Об этой задаче

Библиотеки фильтров, поставляемые в составе DB2 Accessories Suite, обязательно надо устанавливать в соответствии с инструкциями в этой задаче. Возможность поддержки расширенного текстового формата будет работать только в случае наличия нужных библиотек фильтров в соответствующем месте.

Ограничения

Чтобы установить DB2 Accessories Suite для Linux и UNIX, необходимо зарегистрироваться на сервере DB2 в качестве системного администратора. Для Windows вы должны зарегистрироваться как пользователь с полномочиями локального администратора.

Процедура

Чтобы установить DB2 Accessories Suite:

1. Остановите службу экземпляра текстового поиска DB2. Для остановки службы введите команду **db2ts "STOP FOR TEXT"**. Дополнительную информацию об этой команде смотрите в разделе “Остановка службы экземпляра текстового поиска DB2” на стр. 59.
2. Зарегистрируйтесь на сервере DB2 как пользователь с разрешениями, описанными в разделе Ограничения.
3. Распакуйте пакет. Там содержатся следующие файлы:

- Для операционных систем Linux и UNIX:
 - db2_accsuite_license
 - db2-install_directory
 - db2-install_directory/db2tss
 - db2-install_directory/db2tss/lib
 - db2-install_directory/db2tss/lib/ciemsg
 - db2-install_directory/db2tss/bin
 - db2-install_directory/db2tss/bin/richtextTool.sh
 - oit_sdk
 - oit_sdk/<hx_sdk_file>
 - oit_sdk/<sx_sdk_file>
 - readme.txt
 - setup.sh

Файлы в каталоге oit_sdk зависят от платформы.

Платформа	<hx_sdk_file>	<sx_sdk_file>
AIX 64-битная	V13087-01.zip	V13119-01.zip
Linux 32-битная	V13125-01.zip	V13132-01.zip
Linux 64-битная	V13094-01.zip	V13126-01.zip
HP-UX IA64	V13088-01.zip	V13121-01.zip
SunOS x86 64-битная	V13096-01.zip	V13128-01.zip
SunOS SPARC	V13095-01.zip	V13127-01.zip

- Для операционных систем Windows:
 - db2_accsuite_license
 - db2-install_directory
 - db2-install_directory\db2tss
 - db2-install_directory\db2tss\lib
 - db2-install_directory\db2tss\lib\ciemsg.exe
 - db2-install_directory\db2tss\bin
 - db2-install_directory\db2tss\bin\richtextTool.bat
 - oit_sdk\<hx_sdk_file>
 - oit_sdk\<sx_sdk_file>
 - readme.txt
 - setup.bat

Файлы в каталоге oit_sdk зависят от платформы:

Платформа	<hx_sdk_file>	<sx_sdk_file>
Windows 32-битная	V13098-01.zip	V13130-01.zip
Windows 64-битная	V13099-01.zip	V13131-01.zip

4. Проверьте требуемые файлы и пакеты.

- Для операционных систем Linux и UNIX проверьте, что существуют:
 - DB2DIR*/db2tss/lib/ciemsg
 - DB2DIR*/db2tss/bin/richtextTool.sh

, где *DB2DIR* - положение копии DB2 Версии 9.7.

Если каталог db2tss отсутствует, это значит, что DB2 Text Search не был правильно установлен и сконфигурирован.

Если каталог db2tss существует, но требуемые файлы и пакеты отсутствуют, скопируйте ciemsg из пакета в *DB2DIR*/db2tss/lib и скопируйте richtextTool.sh в *DB2DIR*/db2tss/bin, где *DB2DIR* - положение копии DB2 Версии 9.7. После копирования ciemsg нужно ввести команду **db2chgpath**.

Пример:

```
cp ./db2-install_directory/db2tss/lib/ciemsg DB2DIR/db2tss/lib/
cp ./db2-install_directory/db2tss/bin/richtextTool.sh DB2DIR/db2tss/bin/
DB2DIR/install/db2chgpath -f db2tss/lib/ciemsg
```

- Для операционных систем Windows проверьте, что существует:
 - DB2PATH*\db2tss\lib\ciemsg.exe
 - DB2PATH*\db2tss\bin\richtextTool.bat

, где *DB2PATH* - положение копии DB2 Версии 9.7.

Если каталог *db2tss* отсутствует, это значит, что DB2 Text Search не был правильно установлен и сконфигурирован.

Если каталог *db2tss* существует, но требуемые файлы и пакеты отсутствуют, скопируйте *ciemsg.exe* из пакета в *DB2PATH\db2tss\lib* и скопируйте *richtextTool.bat* в *DB2PATH\db2tss\bin*, где *DB2PATH* - положение копии DB2 Версии 9.7.

Пример:

```
copy .\каталог-установки-db2\db2tss\lib\ciemsg.exe DB2PATH\db2tss\lib\  
copy .\каталог-установки-db2\db2tss\bin\richtextTool.bat DB2PATH\db2tss\bin\  

```

5. Для операционных систем Linux и UNIX убедитесь, что разрешение управления было задано для *ciemsg* и *richtextTool.sh*, то есть владелец экземпляра может их запустить.
6. Запустите программу *setup*.

- Для операционных систем Linux и UNIX:
каталог_извлечения_OIT/setup.sh <DB2DIR>

, где *каталог_извлечения_OIT* - это положение, в которое вы распаковали DB2 Accessories Suite, а *<DB2DIR>* - положение копии DB2 Версии 9.7.

- Для операционных систем Windows:
каталог_извлечения_OIT/setup.bat "<DB2PATH>"

, где *каталог_извлечения_OIT* - это положение, в которое вы распаковали DB2 Accessories Suite, а *<DB2PATH>* - положение копии DB2 Версии 9.7.

Результаты

Вы успешно установили DB2 Accessories Suite.

Дальнейшие действия

Теперь можно сконфигурировать поддержку документов в расширенном текстовом формате для DB2 Text Search. Подробности смотрите в теме “Конфигурирование DB2 Text Search для поддержки документов в расширенном текстовом формате” на стр. 56.

Установка DB2 Accessories Suite для DB2 Text Search Версии 9.7 FP4 или более новых версий

DB2 Accessories Suite включает индексирование и поиск для документов в расширенном текстовом формате и в проприетарном формате при помощи DB2 Text Search.

Прежде чем начать

Для установки DB2 Accessories Suite необходимо:

- Скачать DB2 Accessories Suite для Версии 9.7 FP4. Ссылку для скачивания смотрите в теме <https://www.ibm.com/services/forms/preLogin.do?source=swg-dm-db2accsuite>

Убедитесь, что вы скачиваете самую свежую версию комплекта.

Об этой задаче

Для правильного функционирования возможности необходимо, чтобы был установлен соответствующий пакет исправлений DB2 Accessories Suite.

Ограничения

Чтобы установить DB2 Accessories Suite для Linux и UNIX, необходимо зарегистрироваться на сервере DB2 в качестве системного администратора. Для Windows вы должны зарегистрироваться как пользователь с полномочиями локального администратора.

Процедура

Чтобы установить DB2 Accessories Suite:

1. Остановите службу экземпляра текстового поиска DB2. Для остановки службы введите команду **db2ts "STOP FOR TEXT"**. Дополнительную информацию об этой команде смотрите в разделе “Остановка службы экземпляра текстового поиска DB2” на стр. 59.
2. Зарегистрируйтесь на сервере DB2 как пользователь с разрешениями, описанными в разделе Ограничения.
3. Распакуйте пакета.
4. Есть два режим установки. Один из них - установка с консоли, другой - установка без вывода сообщений.
 - Чтобы выполнить установку с консоли:
 - a. Запустите программу InstallDB2AccSuite97.
 - b. Приняв условия лицензии, введите положение подкаталога /db2tss в копии DB2 Версии 9.7, когда система запросит путь установки.
 - c. Каталог db2tss уже должен существовать. Если он отсутствует, это значит, что DB2 Text Search не был правильно установлен и сконфигурирован.
 - d. Проверьте сводку данных и подтвердите установку.
 - Чтобы выполнить установку без вывода сообщений:
 - a. Можно установить DB2 Accessories Suite без вывода сообщений, запустив программу установки с опцией **-i**, указав в опции **-f** файл ответов с параметрами.
 - b. Задайте в файле ответов YES, чтоб принять условия лицензии.

Результаты

Вы успешно установили DB2 Accessories Suite.

Дальнейшие действия

Теперь вы можете включить поддержку документов в расширенном текстовом формате для DB2 Text Search. Подробности смотрите в теме “Включение поддержки документов в расширенном текстовом формате для DB2 Text Search” на стр. 58.

Деинсталляция DB2 Accessories Suite для DB2 Text Search

Деинсталлировать автономную систему DB2® Text Search можно при помощи команды **Uninstall_DB2AS**.

Прежде чем начать

Чтобы деинсталлировать DB2 Accessories Suite для Linux и UNIX, необходимо зарегистрироваться на сервере DB2 в качестве системного администратора. Для платформах Windows вы должны зарегистрироваться как пользователь с полномочиями локального администратора.

Процедура

Чтобы деинсталлировать DB2 Accessories Suite:

1. Остановите службу экземпляра текстового поиска DB2. Для остановки службы введите команду db2ts "STOP FOR TEXT".
2. Зарегистрируйтесь на сервере баз данных DB2 как пользователь с достаточными привилегиями операционной системы.
3. Отключите поддержку расширенных текстовых документов для всех экземпляров текстового поиска, для которых она была включена ранее. Подробности смотрите в теме об отключении поддержки расширенных текстовых документов в DB2 Text Search.
4. Деинсталлируйте программу установки DB2 Accessories Suite. Для деинсталляции программы установки:
 - В операционных системах Linux и UNIX:
`<DB2DIR>/db2tss/Uninstall_DB2AS/Uninstall_DB2AS.bin`

где `<DB2DIR>` - положение последней копии DB2.
 - В операционной системе Windows:
`<DB2PATH>\db2tss\Uninstall_DB2AS\Uninstall_DB2AS.exe`

, где `<DB2PATH>` - это положение, где установлена последняя копия DB2.

Результаты

Вы деинсталлировали DB2 Accessories Suite.

Конфигурирование DB2 Text Search для поддержки документов в расширенном текстовом формате

Серверы DB2 Text Search должны быть правильно сконфигурированы для поддержки документов в расширенном текстовом формате.

Прежде чем начать

Для конфигурирования поддержки документов в расширенном текстовом формате для серверов DB2 Text Search необходимо:

1. Скачать и установить пакет DB2 Accessories Suite (Oracle's Outside In Technology (OIT)). Дополнительную информацию смотрите в разделе "Установка DB2 Accessories Suite для DB2 Text Search Версии 9.7 FP3 или более ранних версий" на стр. 52.
2. От имени администратора или пользователя с разрешениями администратора запустить утилиту richtextTool, чтобы сконфигурировать поддержку документов в расширенном текстовом формате.

Об этой задаче

Утилита `richtextTool` конфигурирует, включает и выключает поддержку документов в расширенном текстовом формате для DB2 Text Search. Этот инструмент нужно запустить до того, как могут быть созданы полнотекстовые индексы для обработки документов в расширенном текстовом формате.

Ограничения

Чтобы запустить **richtextTool setup** в Linux и UNIX, необходимо зарегистрироваться на сервере DB2 как системный администратор или пользователь с разрешениями записи и выполнения в каталоге установки DB2. Чтобы запустить **richtextTool setup** в Windows, необходимо зарегистрироваться как пользователь с полномочиями локального администратора. Чтобы запустить **richtextTool setup**, необходима установленная команда распаковки. Как получить пакет `unzip` для AIX, описано в публикации <http://www-03.ibm.com/systems/power/software/aix/linux/toolbox/alpha.html>

У вас должна быть возможность записи в каталог установки DB2.

Процедура

1. Остановите службу экземпляра текстового поиска DB2. Для остановки службы введите команду **db2ts "STOP FOR TEXT"**. Дополнительную информацию об этой команде смотрите в разделе “Остановка службы экземпляра текстового поиска DB2” на стр. 59.
2. Зарегистрируйтесь на сервере DB2 как пользователь с разрешениями, описанными в разделе Ограничения.
3. Запустите утилиту `richtextTool` для конфигурирования поддержки. Чтобы сконфигурировать поддержку:

- Для операционных систем Linux и UNIX:

```
$INSTHOME/sql/lib/db2tss/bin/richtextTool setup <DB2DIR> <каталог-файлов-zip OIT>
```

, где *INSTHOME* - это домашний каталог экземпляра, *<DB2DIR>* - положение копии установки DB2 Версии 9.7 и *<каталог-файлов-zip OIT>* содержит сжатые файлы библиотеки OIT (поиск и экспорт HTML).

Примечание: Если продукт DB2 установлен от имени пользователя `root`, запустить конфигурирование утилиты `richtextTool` надо от имени пользователя `root`. В противном случае нужно запустить конфигурирование утилиты `richtextTool`, используя ID пользователя владельца экземпляра.

- Для операционных систем Windows:

```
DB2PATH\db2tss\bin\richtextTool.bat setup <DB2PATH> <каталог-файлов-zip OIT>
```

, где *<DB2PATH>* - положение копии DB2 Версии 9.7 и *<каталог-файлов-zip OIT>* содержит сжатые файлы `zip` библиотеки OIT (поиск и экспорт html).

4. Включить поддержку расширенного текста. Инструкции о включении поддержки расширенного текста смотрите в теме “Включение поддержки документов в расширенном текстовом формате для DB2 Text Search” на стр. 58.

Результаты

Вы сконфигурировали поддержку расширенного текста для сервера DB2 Text Search.

Включение поддержки документов в расширенном текстовом формате для DB2 Text Search

Поддержка документов в расширенном текстовом формате может быть включена на правильно сконфигурированных серверах DB2 Text Search.

Прежде чем начать

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. Чтобы включить поддержку документов в расширенном текстовом формате для серверов DB2 Text Search, надо от имени владельца экземпляра запустить утилиту **richtextTool** с опцией **enable**.

Начиная с DB2 Версии 9.7 FP3 и для более ранних версий, перед включением поддержки документов в расширенном текстовом формате каждый сервер DB2 Text Search нужно сконфигурировать для поддержки документов в расширенном текстовом формате. Более подробную информацию смотрите в теме “Конфигурирование DB2 Text Search для поддержки документов в расширенном текстовом формате” на стр. 56.

Начиная с DB2 Версии 9.7 FP4 и для более новых версий, перед включением поддержки документов в расширенном текстовом формате каждый сервер DB2 Text Search надо подготовить для поддержки документов в расширенном текстовом формате. Дополнительную информацию смотрите в разделе “Установка DB2 Accessories Suite для DB2 Text Search Версии 9.7 FP4 или более новых версий” на стр. 54

Об этой задаче

Ограничения

Чтобы ввести команду **richtextTool enable**, необходимо зарегистрироваться как владелец экземпляра.

Процедура

1. Войдите как пользователь - владелец экземпляра.
2. Остановите службу экземпляра текстового поиска DB2. Для остановки службы введите команду **db2ts "STOP FOR TEXT"**. Дополнительную информацию об этой команде смотрите в разделе “Остановка службы экземпляра текстового поиска DB2” на стр. 59.
3. Запустите утилиту **richtextTool** для включения поддержки. Чтобы включить поддержку:

- Для операционных систем Linux и UNIX:

```
$INSTHOME/sql/lib/db2tss/bin/richtextTool enable <DB2DIR>
```

где *INSTHOME* - домашний каталог экземпляра, а *<DB2DIR>* - положение копии DB2 Версии 9.7.

- Для операционных систем Windows:

```
DB2PATH\db2tss\bin\richtextTool.bat enable <DB2PATH>
```

, где *DB2PATH* - это положение, где установлена ваша копия DB2 Версии 9.7.

4. Запустите службу экземпляра текстового поиска DB2. Для запуска службы введите команду **db2ts "START FOR TEXT"**. Дополнительную информацию об этой команде смотрите в разделе “Запуск службы экземпляра текстового поиска DB2” на стр. 51.

Результаты

Вы включили поддержку расширенного текста для сервера DB2 Text Search.

Остановка службы экземпляра текстового поиска DB2

При остановке службы экземпляра DB2 Text Search все текущие выполняемые команды **db2ts** будут завершены, но новые команды запустить нельзя.

Об этой задаче

Чтобы остановить службу экземпляра DB2 Text Search, введите следующую команду:
db2ts "STOP FOR TEXT"

Выключение базы данных для DB2 Text Search

Если вы больше не собираетесь выполнять текстовые поиски в базе данных, отключите текстовый поиск для нее.

Об этой задаче

При выключении базы данных для текстового поиска удаляются также следующие объекты из схемы SYSIBMTS:

- SYSIBMTS.TSDEFAULTS
- SYSIBMTS.TSLOCKS
- SYSIBMTS.TSINDEXES
- SYSIBMTS.TSCONFIGURATION
- SYSIBMTS.TSCOLLECTIONNAMES

Процедура

Чтобы выключить базу данных для текстового поиска, воспользуйтесь одним из следующих способов:

1. Отбросьте все определенные в базе данных индексы текстового поиска при помощи команды **DROP INDEX**.
2. Чтобы выключить базу данных для текстового поиска, воспользуйтесь одним из следующих способов:
 - Введите команду **DISABLE DATABASE FOR TEXT**:
db2ts "DISABLE DATABASE FOR TEXT CONNECT TO *имя-базы-данных*"
 - Вызовите процедуру SYSPROC.SYSTS_DISABLE:
 - CALL SYSPROC.SYSTS_DISABLE('en_US', ?)

Примечание: Индексы текстового поиска можно также отбросить при помощи опции **FORCE**. Однако при этом после выключения базы данных могут остаться некоторые данные, в частности, относящиеся к собранию текстового поиска. Это возможно, поскольку опция **FORCE** позволяет отбросить индексы текстового поиска даже при отключенном сервере текстового поиска DB2.

Удаление бесхозных собраний текстового поиска

Одна из задач администрирования, которую может потребоваться выполнить при использовании DB2 Text Search - это удаление бесхозных собраний. Бесхозные собрания можно удалить при помощи инструмента администрирования.

Об этой задаче

Собрание текстового поиска - это базовое представление индекса текстового поиска. Между собранием текстового поиска и индексом текстового поиска существует однозначная взаимосвязь. Команды **db2ts** работают с индексами текстового поиска, тогда как инструменты текстового поиска работают с собраниями текстового поиска. Существующее собрание текстового поиска, соответствующее индексу текстового поиска, который больше не существует, называется бесхозным собранием.

Кроме того, если не удастся связаться с сервером текстового поиска, можно создать бесхозное собрание, выполнив следующие операции:

- Отбросив таблицу или базу данных
Оператор DROP TABLE и команда **DROP DATABASE** не отбрасывают собрания текстового поиска автоматически.
- Применив опцию **FORCE** с операцией **DISABLE** и операцией **DROP index**.
Все эти команды или хранимые процедуры выполняются успешно, даже если сервер текстового поиска недоступен.

Чтобы определить, существуют ли бесхозные собрания:

1. Примените инструмент администрирования, чтобы получить отчет обо всех собраниях текстового поиска:
`adminTool status -configPath абсолютный-путь-к-папке-config`
2. Запросите управляющую производную таблицу SYSIBMTS.TSCOLLECTIONNAMES, чтобы получить отчет обо всех индексах текстового поиска для текущей базы данных:
`SELECT имя-собрания FROM SYSIBMTS.TSCOLLECTIONNAMES`

Передайте этот запрос на выполнение для всех баз данных с включенной поддержкой DB2 Text Search и объедините результаты в список.

Примечание: Инструмент администрирования возвратит список всех собраний текстового поиска, а запрос SYSIBMTS.TSCOLLECTIONNAMES - список индексов текстового поиска только для текущей базы данных.

3. Сравните списки, возвращенные инструментом администрирования и оператором SELECT. Любое собрание текстового поиска, возвращенное инструментом администрирования, но не возвращенное оператором SELECT, является бесхозным. Единственное исключение - это собрание по умолчанию "Default", создаваемое при запуске DB2 Text Search.

Чтобы удалить бесхозные собрания текстового поиска, введите следующую команду:

```
adminTool delete -configPath абсолютный-путь-к-папке-config  
-collectionName имя-собрания
```

Пример

В текущий момент у пользователя включена поддержка DB2 Text Search для базы данных DBCP1208, работающей в системе UNIX. Чтобы определить, существуют ли собрания текстового поиска, пользователь применяет инструмент администрирования и оператор SELECT:

```
adminTool.sh status -configPath $HOME/sql1lib/db2tss/config
```

CollectionName	IndexSize	NumOfDocuments	
Default	13,159B	0	
tigertail_DBCP1208_TS542717_0000		13,159B	11
tigertail_DBCP1208_TS012817_0000		13,159B	17
tigertail_DBCP1208_TS082817_0000		13,159B	16
tigertail_DBCP1208_TS152817_0000		13,159B	18
tigertail_DBCP1208_TS212817_0000		13,159B	16
tigertail_DBCP1208_TS302817_0000		13,159B	17
tigertail_DBCP1208_TS392817_0000		13,159B	10
tigertail_DBCP1208_TS462817_0000		13,159B	10
tigertail_DBCP1208_TS542817_0000		13,159B	12
tigertail_DBCP1208_TS022917_0000		13,159B	10
tigertail_DBCP1208_TS112917_0000		13,159B	16
tigertail_DBCP1208_TS192917_0000		13,159B	11
tigertail_DBCP1208_TS262917_0000		13,159B	12
tigertail_DBCP1208_TS867530_0000		13,159B	16

```
db2 select collectionname from sysibmts.tscollectionnames
```

```
COLLECTIONNAME
```

```
-----  
tigertail_DBCP1208_TS542717_0000  
tigertail_DBCP1208_TS012817_0000  
tigertail_DBCP1208_TS082817_0000  
tigertail_DBCP1208_TS152817_0000  
tigertail_DBCP1208_TS212817_0000  
tigertail_DBCP1208_TS302817_0000  
tigertail_DBCP1208_TS392817_0000  
tigertail_DBCP1208_TS462817_0000  
tigertail_DBCP1208_TS542817_0000  
tigertail_DBCP1208_TS022917_0000  
tigertail_DBCP1208_TS112917_0000  
tigertail_DBCP1208_TS192917_0000  
tigertail_DBCP1208_TS262917_0000
```

Выбрано 13 записей.

Сравнив два этих вывода, пользователь видит, что у собрания текстового поиска `tigertail_DBCP1208_TS867530_0000` нет соответствующего ему индекса текстового поиска. Тогда пользователь при помощи инструмента администрирования удаляет это бесхозное собрание:

```
adminTool.sh delete -configPath $HOME/sql1lib/db2tss/config  
-collectionName tigertail_DBCP1208_TS867530_0000
```

Выключение поддержки документов в расширенном текстовом формате для DB2 Text Search

Поддержка документов в расширенном текстовом формате для серверов DB2 Text Search может быть выключена в любое время.

Прежде чем начать

В Версии 9.7 у владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. Чтобы выключить поддержку документов в расширенном текстовом формате для серверов DB2 Text Search, надо от имени владельца экземпляра запустить утилиту **richtextTool** с опцией **disable**.

Об этой задаче

Ограничения

Чтобы запустить команду **richtextTool disable**, вы должны быть зарегистрированы как владелец экземпляра.

Процедура

1. Войдите как пользователь - владелец экземпляра.
2. Остановите службу экземпляра текстового поиска DB2. Для остановки службы введите команду **db2ts "STOP FOR TEXT"**. Дополнительную информацию об этой команде смотрите в разделе “Остановка службы экземпляра текстового поиска DB2” на стр. 59.
3. Запустите утилиту **richtextTool**, чтобы выключить поддержку. Для выключения поддержки:
 - Для операционных систем Linux и UNIX:

```
$INSTHOME/sqllib/db2tss/bin/richtextTool disable <DB2DIR>
```

где *INSTHOME* - домашний каталог экземпляра, а *<DB2DIR>* - положение копии DB2 Версии 9.7.
 - Для операционных систем Windows:

```
DB2PATH\db2tss\bin\richtextTool.bat disable
```



```
DB2PATH\db2tss\bin\richtextTool.bat disable DB2PATH
```

, где *DB2PATH* - это положение, где установлена ваша копия DB2 Версии 9.7.
4. Запустите службу экземпляра текстового поиска DB2. Для запуска службы введите команду **db2ts "START FOR TEXT"**. Дополнительную информацию об этой команде смотрите в разделе “Запуск службы экземпляра текстового поиска DB2” на стр. 51.

Результаты

Вы выключили поддержку расширенного текста для сервера DB2 Text Search.

Создание индексов текстового поиска

Индекс текстового поиска - это компиляция значимых терминов, извлеченных из текстовых документов. Для каждого термина сохраняется информация о том, из какого документа он извлечен.

Индекс тестового поиска создается один раз для каждого столбца, содержащего текст для поиска. При создании индекса текстового поиска одновременно создаются следующие объекты:

Промежуточная таблица

Сохраняет данные всех измененных строк в пользовательской таблице.

Таблица событий

Собирает информацию о состоянии команды обновления индексов или обо всех ошибках, встреченных при ее обработке. Если ошибки происходят во время индексирования, к таблице событий добавляются *события обновления индекса*.

Триггеры для пользовательской таблицы

Триггеры добавляют информацию к промежуточной таблице при всяком добавлении, удалении или изменении документа в столбце. Эта информация необходима для синхронизации индекса при следующем индексировании.

Примечание: Если вы используете команду **LOAD** для заполнения документов, триггеры не активируются и инкрементное индексирование загруженных документов не работает. Используйте вместо этого команду **IMPORT**, которая активирует триггеры.

После создания индекса текстового поиска он еще пуст (то есть непригоден для поиска), пока вы его не измените. При создании индекса текстового поиска вы можете указать, чтобы служба экземпляра текстового поиска периодически проверяла, не требуется ли обновление индекса текстового поиска, чтобы при необходимости запускать команду обновления.

Создание индекса текстового поиска

После включения DB2 Text Search в базе данных можно создать индексы текстового поиска для столбцов, содержащих текст, в котором нужно выполнять поиск.

Прежде чем начать

Для создания индекса текстового поиска требуется один из следующих уровней полномочий:

- Привилегия CONTROL для выбранной таблицы
- Привилегия INDEX для таблицы индекса плюс либо полномочия IMPLICIT_SCHEMA для базы данных, либо привилегия CREATEIN для схемы таблицы индекса
- Полномочия DBADM с DATAACCESS

Ограничения

- Для этой таблицы должен существовать первичный ключ. Если первичный ключ не существует, его надо создать до создания индекса.
- Текстовый столбец в индексе должен быть одного из следующих поддерживаемых типов:
 - CHAR
 - VARCHAR
 - LONG VARCHAR
 - CLOB
 - GRAPHIC
 - VARGRAPHIC
 - LONG VARGRAPHIC
 - DBCLOB
 - BLOB
 - XML
- Объекты, связанные с текстовым поиском, должны, помимо правил именования DB2, подчиняться следующему ограничению на используемые символы:
 - [A-Za-z][A-Za-z0-9@#\$_]* или

– "[A-Za-z][A-Za-z0-9@#\$_]*"

Это ограничение применяется к следующим сценариям:

- имена схем, содержащих индексы текстового поиска
 - имена таблиц, с которыми связаны индексы текстового поиска
 - имена текстовых столбцов
 - имена индексов текстового поиска
- Если первичный ключ составной, порядок столбцов в нем должен быть такой же, как в базовой таблице.

Об этой задаче

Процедура

Создайте индекс текстового поиска одним из следующих методов:

- Введите команду **CREATE INDEX**:

```
db2ts "CREATE INDEX имя-индекса FOR TEXT ON имя-таблицы (имя-столбца)"
```
- Вызовите хранимую процедуру SYSPROC.SYSTS_CREATE:

```
CALL SYSPROC.SYSTS_CREATE('схема-индекса', 'имя-индекса', 'имя-таблицы  
(имя-столбца)', 'опции', 'локаль', ?)
```

Пример

Например, пусть таблица PRODUCT в базе данных SAMPLE содержит столбцы для ID продукта, имени, цены, описания и так далее. Чтобы создать индекс текстового поиска с именем MYSCHEMA.MYTEXTINDEX для столбца NAME, введите следующую команду или вызовите хранимую процедуру, как описано ниже:

```
db2ts "CREATE INDEX MYSCHEMA.MYTEXTINDEX FOR TEXT ON PRODUCT(NAME)"  
CALL SYSPROC.SYSTS_CREATE('myschema', 'myTextIndex',  
'product (name)', '', 'en_US', ?)
```

Аналогичным образом, чтобы создать индекс текстового поиска с именем MYSCHEMA.MYXMLINDEX для столбца XML DESCRIPTION (описание), введите следующую команду:

```
db2ts "CREATE INDEX MYSCHEMA.MYXMLINDEX FOR TEXT ON PRODUCT(DESCRIPTION)"
```

или

```
CALL SYSPROC.SYSTS_CREATE('myschema', 'myXMLIndex',  
'product (description)', '', 'en_US', ?)
```

Задание частоты обновления

Если вы не хотите вручную управлять обновлением индекса текстового поиска по мере изменения таблицы, с которой он связан, можете добавить в команду **db2ts CREATE INDEX** параметры обновления. **UPDATE MINIMUM** задает минимальное число изменений в таблице, при котором требуется обновление, а **UPDATE FREQUENCY** задает, как часто службы обновления должны проверять, требуется ли обновление. Например, чтобы задать, что MYSCHEMA.MYTEXTINDEX нужно обновлять после минимум пяти изменений, и что службы обновления должны выполнять проверку каждый понедельник и среду в полночь и в полдень, введите следующую команду:

```
db2ts "CREATE INDEX MYSCHEMA.MYTEXTINDEX FOR TEXT ON PRODUCT(NAME)  
UPDATE FREQUENCY d(1,3) h(0,12) m(0) UPDATE MINIMUM 5"  
CALL SYSPROC.SYSTS_CREATE('myschema', 'myTextIndex', 'product (name)',  
'UPDATE FREQUENCY D(1,3) H(0,12) M(0)' 'UPDATE MINIMUM 5', 'en_US', ?)
```

Спецификация языка

Создавая индекс, вы можете задать так называемую локаль (то есть язык и территорию), используя опцию **LANGUAGE**. Чтобы выполнить автоматический просмотр документов для определения его локали, задайте для **LANGUAGE** значение **AUTO**. Если не задать **LANGUAGE**, будет использовано значение **LANGUAGE** по умолчанию. Это значение по умолчанию будет взято при помощи **DEFAULTVALUE** из **SYSIBMTS.TSDEFAULTS** при **DEFAULTNAME='LANGUAGE'**. (В этом случае **DEFAULTVALUE** задается тогда, когда для базы данных включается текстовый поиск. Это значение вычисляется по территории базы данных, если эту территорию удалось отобразить в одну из поддерживаемых локалей документов. Если определить поддерживаемую локаль документов по территории базы данных не удалось, для **DEFAULTVALUE** задается значение **AUTO**.)

Создание индекса текстового поиска для двоичных типов данных

При создании индекса текстового поиска можно задать кодовую страницу для двоичного столбца. Это помогает механизму текстового поиска DB2 идентифицировать кодировку символов.

Об этой задаче

Чтобы задать при создании индекса текстового поиска кодовую страницу, введите следующую команду:

```
db2ts "CREATE INDEX имя-индекса FOR TEXT ON имя-таблицы  
CODEPAGE кодировка-страница"
```

Данные, хранящиеся в столбце с двоичным типом данных, таким как **BLOB** или **FOR BIT DATA**, не преобразуются. Это значит, что для указанных документов сохраняются их исходные кодовые страницы, что может привести к ошибкам при создании индекса текстового поиска, поскольку у вас могут быть две разные кодовые страницы. Поэтому необходимо определить, используется ли кодовая страница базы данных или кодовая страница, заданная для команды **db2ts CREATE INDEX**. Если вы не знаете, какая кодовая страница была использована для создания индекса текстового поиска, это можно выяснить, введя следующий оператор:

```
db2 "SELECT CODEPAGE FROM SYSIBMTS.TSINDEXES where INDSHEMA='имя-схемы'  
and INDNAME='имя-индекса'"
```

Создание индекса текстового поиска для неподдерживаемых типов данных

Если документы хранятся в столбце неподдерживаемого типа данных, например, пользовательского типа данных, задайте функцию, которая принимает пользовательский тип на входе и преобразует его на выходе в один из поддерживаемых типов.

Об этой задаче

Текстовый столбец в индексе должен быть одного из следующих поддерживаемых типов:

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

- BLOB
- XML

Чтобы преобразовать тип данных столбца в один из допустимых типов, используйте один из следующих методов:

- Введите команду **db2ts CREATE INDEX** с именем функции преобразования.

```
db2ts "CREATE INDEX имя-индекса FOR TEXT ON
имя-таблицы имя-функции(имя-текстового-столбца)"
```
- Используйте пользовательскую функцию, указываемую как *имя-функции*, которая обращается к текстовым документам в столбце с неподдерживаемым текстовым поиском типом данных, выполняет преобразование типа для значения из этого столбца и возвращает значение в одном из поддерживаемых типов данных.

Пример

В следующем примере таблица UDTTABLE содержит столбец пользовательского типа (UDT) с именем "COMPRESSED_TEXT", который определен как CLOB(1M). Чтобы создать индекс для этого типа данных, создайте сначала пользовательскую функцию с именем UNCOMPRESS, которая получает значения типа COMPRESSED_TEXT. Затем создайте индекс текстового поиска так:

```
db2ts "CREATE INDEX UDTINDEX FOR TEXT ON
UDTTABLE UNCOMPRESS(text) ..."
```

Повышение производительности индексов текстового поиска

Есть ряд способов повысить производительность индексов текстового поиска.

Об этой задаче

Чтобы повысить производительность во время индексации, используйте следующие приемы:

- Для сохранения индекса текстового поиска и файлов баз данных используйте тип данных VARCHAR.
- Для столбцов первичного ключа используйте компактные типы данных, такие как TIMESTAMP или INTEGER, но не VARCHAR.
- Убедитесь, что в системе достаточный объем доступной реальной памяти для операций обновления индекса. Для обновлений индексов требуется память сверх того, что требуется для пулов буферов баз данных. Если памяти не хватает, операционная система вместо буферизации начинает использовать пространство подкачки страниц, что значительно снижает производительность поиска.
- Избегайте работы с большим числом малых документов - это увеличивает использование памяти на каждый документ.

Кроме того, имейте в виду, что производительность может снижаться при обновлении индекса, если в таблицу событий записываются сообщения об ошибках и предупреждения.

Обслуживание индексов текстового поиска

После создания индексов текстового поиска надо выполнить несколько задач по обслуживанию. Есть несколько способов выполнять эти задачи - с помощью различных команд администрирования, хранимых процедур и инструментов администрирования.

Существуют следующие стандартные задачи по обслуживанию индекса текстового поиска:

- Запуск периодических обновлений
Если не выполняется автоматическое обновление, необходимо обновлять индексы текстового поиска, чтобы они отражали изменения связанных с ними индексируемых текстовых столбцов.
- Мониторинг таблицы событий
Можно использовать таблицу событий для обнаружения ошибок в документах и определения, не следует ли изменить частоту обновлений.

Менее часто требуется выполнять изменение и отбрасывание индексов поиска.

Обновление индекса текстового поиска

Индекс текстового поиска можно обновлять автоматически или вручную. Автоматическое обновление выполняется на основе способа, которым был определен индекс текстового поиска. Обновить индексы вручную можно, введя команду или вызвав хранимую процедуру.

Прежде чем начать

Для обновления индекса текстового поиска требуется привилегия CONTROL или полномочия DATAACCESS для таблицы назначения.

Об этой задаче

После создания и первого обновления (заполнения) индекс текстового поиска нужно поддерживать в актуальном состоянии. Например, при добавлении текстового документа в базу данных или изменении существующего документа в базе данных его нужно проиндексировать для сохранения синхронизации содержимого индекса текстового поиска с содержимым базы данных. Кроме того, при удалении текстового документа из базы данных его термины нужно также удалить и из индекса текстового поиска.

Вы должны тщательно спланировать периодическое индексирование, поскольку индексация текстовых документов - это задача, занимающая много времени и ресурсов. Затрачиваемое время зависит от многих показателей, включая размер документов, число добавленных или измененных документов с момента предыдущего обновления индекса текстового поиска и мощность процессора.

Опция status инструмента администрирования позволяет получить информацию о ходе обновления документов во время выполнения команды **db2ts UPDATE INDEX**. Если новое обновление индекса начинается, когда предыдущее обновление еще не завершено, новое обновление завершается неудачно.

Автоматические обновления

Для автоматического обновления индексов текстового поиска используется одна из следующих команд:

- **db2ts CREATE INDEX**
- **db2ts ALTER INDEX**

Параметр **UPDATE FREQUENCY**, минимальное значение которого составляет пять минут, задает частоту обновления. Параметр **UPDATE MINIMUM** задает минимальное число изменений текста, которые надо помещать в очередь.

Если на заданные дату и время в промежуточной таблице окажется недостаточно изменений, индекс текстового поиска обновлен не будет.

Обновления вручную

Бывают также моменты, когда индекс текстового поиска надо обновить немедленно, например, после создания индекса текстового поиска, если индекс все еще пуст или если вы добавили в базу данных несколько текстовых документов и хотите выполнить поиск.

Для заполнения (обновления) индекса текстового поиска данными таблиц или его синхронизации с данными таблиц используется один из следующих способов:

- Введите команду **UPDATE INDEX**:
`db2ts "UPDATE INDEX имя-индекса FOR TEXT"`
- Вызовите управляющую подпрограмму SQL SYSPROC.SYSTS_UPDATE:

Пример

Предположим, для таблицы PRODUCT созданы два индекса текстового поиска: MYSCHEMA.MYTEXTINDEX по столбцу NAME и MYSCHEMA.MYXMLINDEX по столбцу DESCRIPTION. В таблицу PRODUCT добавляется новая запись:

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

Чтобы сделать информацию в новой записи доступной для поиска, введите следующую команду:

```
db2ts "UPDATE INDEX MYSCHEMA.MYTEXTINDEX FOR TEXT"
db2ts "UPDATE INDEX MYSCHEMA.MYXMLINDEX FOR TEXT"
```

Очистка событий индекса текстового поиска

Если сообщения в производной таблице событий индекса больше не требуются, их можно очистить (удалить).

Прежде чем начать

Очистка производной таблицы событий требует или привилегии CONTROL, или DBADM с полномочиями DATAACCESS.

Об этой задаче

Информация о событиях индексирования, такая как время начала и окончания обновления, число проиндексированных документов или ошибки документов, произошедшие во время обновления, хранится в производной таблице событий индекса текстового поиска. Эта информация может помочь при определении причин ошибок.

Процедура

Для очистки производной таблицы событий индекса текстового поиска используйте один из следующих способов:

- Введите команду **db2ts CLEAR EVENTS FOR INDEX** следующим образом:
`db2ts "CLEAR EVENTS FOR INDEX имя-индекса FOR TEXT"`

- Используйте административную подпрограмму SQL SYSPROC.SYSTS_CLEAR_EVENTS следующим образом:
`CALL SYSPROC.SYSTS_CLEAR_EVENTS('схема-индекса', 'имя-индекса', 'локаль')`

Изменение индекса текстового поиска

Вы можете изменить свойства обновления индекса текстового поиска.

Прежде чем начать

Команде **db2ts ALTER INDEX** требуется или привилегия CONTROL, или DBADM с полномочиями DATAACCESS для таблицы назначения.

Процедура

Для изменения индекса используйте один из следующих способов:

- Введите следующую команду:
`db2ts "ALTER INDEX имя-индекса FOR TEXT характеристики-изменения"`

Здесь *характеристики-изменения* - характеристика, такая как частота обновления индекса текстового поиска.

- Вызовите управляющую подпрограмму SQL SYSPROC.SYSTS_ALTER:
`CALL SYSPROC.SYSTS_ALTER('db2ts', 'myTextIndex', 'опция-изменения', 'en_US', ?)`

Здесь *опция-изменения* - характеристика, такая как частота обновления индекса текстового поиска.

Результаты

Если обновление индекса уже выполняется, появится сообщение об ошибке, информирующее о том, что индекс текстового поиска в настоящее время заблокирован и изменения невозможны.

Пример

Вы можете использовать любой из способов, чтобы изменить и частоту обновления индекса текстового поиска, и минимальное количество изменений данных, требуемых для инициирования обновления. (Если никакие параметры не заданы, текущие установки останутся неизменными). Например, чтобы изменить частоту обновления для индекса текстового поиска MYTEXTINDEX таким образом, чтобы он обновлялся с понедельника по пятницу в 12 и 15 часов при условии, что в индексированном столбце произведено по крайней мере 100 изменений, введите следующую команду:

```
db2ts "ALTER INDEX MYTEXTINDEX FOR TEXT
UPDATE FREQUENCY d(1,2,3,4,5) h(12,15) m(00) UPDATE MINIMUM 100"
```

Чтобы отменить периодическое обновление MYTEXTINDEX, введите следующую команду:

```
db2ts "ALTER INDEX MYTEXTINDEX FOR TEXT UPDATE FREQUENCY NONE"
```

Просмотр состояния индексов текстового поиска

Для получения информации о текущих индексах текстового поиска в базе данных ее можно запросить из административной производной таблицы или же использовать инструмент администрирования.

Об этой задаче

Для получения информации о доступных текущих индексах текстового поиска используйте следующую команду:

```
db2 "select имя-столбца from SYSIBMTS.TSINDEXES"
```

Для проверки при помощи инструмента администрирования всех собраний текстового поиска и их свойств используйте следующую команду:

```
adminTool status  
-configPath абсолютный-путь-к-папке-config
```

Отбрасывание индекса текстового поиска

Если вы больше не собираетесь выполнять текстовые поиски в базе данных, отключите текстовый поиск для нее.

Прежде чем начать

Когда вы вводите команду **DROP INDEX**, у ID авторизации оператора должны быть по крайней мере одна из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен индекс
- Полномочия DBADM

Об этой задаче

Когда отбрасывается индекс текстового поиска, отбрасываются также и следующие объекты:

- Промежуточные таблицы индекса и таблицы событий индекса
- Триггеры для пользовательской таблицы

Всегда отбрасывайте индексы текстового поиска для таблицы до отбрасывания самой таблицы. Если вы сначала отбросите таблицу, может образоваться так называемое *бесхозное собрание*. При создании индекса текстового поиска создается собрание (представление индекса в файловой системе) с автоматически сгенерированным именем. Если такое собрание остается в файловой системе после отбрасывания индекса, это может привести к проблемам с будущими запросами в следующих условиях:

- используется то же соединение с базой данных,
- создается таблица с таким же именем,
- для этой таблицы создается текстовый индекс с тем же, что и прежде, именем и
- повторно запускается тот же запрос, что и до этого.

В этом случае может повторно использоваться кэшированный план запроса, что приведет к неправильному результату запроса.

Процедура

Для отбрасывания индекса текстового поиска используйте один из следующих способов:

- Введите команду **DROP INDEX**:

```
db2ts "DROP INDEX имя-индекса FOR TEXT"
```
- Вызовите хранимую процедуру SYSPROC.SYSTS_DROP:

```
CALL SYSPROC.SYSTS_DROP('схема-индекса', 'имя-индекса', 'локаль', ?)
```

Здесь *локаль* - это пятисимвольный код локали, например, en_US, задающий, на каком языке будут записываться сообщения в файл журнала.

Дальнейшие действия

Примечание: Если после отбрасывания индексов текстового поиска остались какие-то бесхозные собрания, их можно удалить при помощи инструмента администрирования.

Если после отбрасывания индекса текстового поиска вы планируете создать новый индекс для того же текстового столбца, необходимо сначала отсоединиться от базы данных, а затем соединиться с ней снова.

Резервное копирование и восстановление индексов текстового поиска

Об этой задаче

Процедура

- Выполните следующие действия для резервного копирования базы данных DB2 с индексами Text Search:
 1. Получите текущий список положений индексов DB2 Text Search

```
db2 "select indschema, indname, collectiondirectory, collectionnameprefix  
from sysibmts.tsindexes"
```
 2. Убедитесь, что никакая административная команда DB2 Text Search не запущена
 3. Остановите службы DB2 Text Search

```
db2ts stop for text
```
 4. Сделайте резервную копию базы данных, введя следующие команды:

```
db2 backup database  
<имя_базы_данных>
```
 5. Сделайте резервные копии каталогов и подкаталогов индекса текстового поиска.
 6. Повторно запустите службы DB2 Text Search
- Выполните следующие действия для восстановления базы данных DB2 с индексами Text Search:
 1. Убедитесь, что никакая административная команда DB2 Text Search не запущена
 2. Остановите службы DB2 Text Search

```
db2ts stop for text
```
 3. Восстановите базу данных, введя следующие команды:

```
db2 restore database <имя_базы_данных>
```
 4. Восстановите из резервной копии положения индекса текстового поиска на прежнем месте.
 5. Повторно запустите службы DB2 Text Search

```
db2ts start for text
```

Словари синонимов для DB2 Text Search

Словарь синонимов содержит слова - синонимы друг друга. Словарь синонимов можно использовать для поиска синонимов терминов вашего запроса в индексе текстового поиска, что повышает результативность запросов поиска.

Используя словарь синонимов, можно искать специфичные для вашей организации слова, например, сокращения и технические жаргонизмы.

По умолчанию словарь синонимов для поиска не используется. Чтобы включить его использование, надо непосредственно добавить этот словарь в конкретный индекс текстового поиска. До использования словаря синонимов индекс текстового поиска должен быть по крайней мере однажды изменен. После добавления словаря синонимов его можно изменять сколь угодно часто по необходимости.

Словарь синонимов состоит из групп синонимов, определяемых вами в файле XML, как это показано в следующем примере:

```
<?xml version="1.0" encoding="UTF-8"?>
<synonymgroups version="1.0">
  <synonymgroup>
    <synonym>ball</synonym>
    <synonym>globe</synonym>
    <synonym>sphere</synonym>
    <synonym>orb</synonym>
  </synonymgroup>
  <synonymgroup>
    <synonym>worldwide patent tracking system</synonym>
    <synonym>wpts</synonym>
  </synonymgroup>
</synonymgroups>
```

Добавление словаря синонимов для DB2 Text Search

При помощи инструмента синонимов можно легко добавить словарь синонимов в индекс текстового поиска.

Прежде чем начать

- Для возможности добавления словаря синонимов в индекс текстового поиска сначала нужно активировать службу экземпляра DB2 Text Search.
- Индекс текстового поиска уже должен быть обновлен хотя бы один раз.
- Должен также существовать файл XML синонимов, в котором задаются группы синонимов.

Процедура

Чтобы добавить словарь синонимов:

1. Скопируйте указанный файл XML в произвольный каталог на сервере DB2 Text Search.
2. Определите имя собрания текстового поиска, связанного с индексом текстового поиска, куда вы хотите добавить словарь синонимов. При помощи инструмента администрирования можно получить отчет обо всех собраниях текстового поиска:
`adminTool status -configPath абсолютный-путь-к-папке-config`
3. С помощью инструмента синонимов добавьте словарь синонимов в конкретный индекс текстового поиска. Синонимы можно добавлять в режиме **append** или **replace**, то есть добавлять их к существующим синонимам, определенным для данного индекса текстового поиска, или же заменять их.
`synonymTool importSynonym -synonymFile абсолютный-путь-к-файлу-синонимов
-collectionName имя-собрания -replace true или false
-configPath абсолютный-путь-к-папке-config`

Примечание: Если формат XML недопустим или файл XML пуст, будет возвращена ошибка.

Пример

Например, чтобы добавить файл синонимов `synfile.xml` в режиме добавления (`append`), введите следующую команду:

```
synonymTool importSynonym
-synonymFile $HOME/sql1lib/misx/xmlsynfile.xml
-collectionName tigertail_DBCP1208_TS867530_0000
-replace false
-configPath $HOME/sql1lib/db2tss/config
```

Удаление словаря синонимов для DB2 Text Search

Словарь синонимов надо удалять последовательно, удаляя его для каждого собрания по отдельности, поэтому инструмент синонимов нужно использовать для каждого индекса текстового поиска.

Об этой задаче

Чтобы удалить словарь синонимов, введите следующую команду:

```
synonymTool removeSynonym -collectionName имя-собрания
-configPath абсолютный-путь-к-папке-config
```

Где *имя-собрания* задает собрание текстового поиска, а *абсолютный-путь-к-папке-config* - абсолютный путь к папке конфигурации текстового поиска.

Глава 6. Поиск с использованием индексов текстового поиска

После заполнения индекса текстового поиска данными можно использовать этот индекс для поиска. DB2 Text Search поддерживает поиск в SQL, XQuery и SQL/XML.

Можно использовать следующие функции поиска:

- Функция SQL CONTAINS и функция XML xmlcolumn-contains - для создания запросов определенных слов или словосочетаний
- Функция SQL SCORE - для получения оценки релевантности найденного текстового документа

Поиск по индексу текстового поиска может быть как простым, например при поиске вхождений одного слова в названии, так и сложным, например при запросах с использованием логических операторов или заданием весов для термов. Помимо операторов, позволяющих уточнять сложность запросов, есть функции, дающие возможность улучшить поиск по индексам текстового поиска, например словари синонимов и лингвистическая поддержка.

Функции поиска для DB2 Text Search

После обновления индекса текстового поиска можно выполнять поиск с помощью скалярной функции поиска CONTAINS или SCORE SQL или с помощью функции xmlcolumn-contains.

Поиск по индексу текстового поиска может быть как простым, например при поиске вхождений одного слова в названии, так и сложным, например при запросах с использованием логических операторов или заданием весов для термов. Помимо операторов, позволяющих уточнять сложность запросов, есть функции, дающие возможность улучшить поиск по индексам текстового поиска, например словари синонимов и лингвистическая поддержка.

Можно использовать следующие функции поиска:

- Функция SQL CONTAINS и функция XML xmlcolumn-contains - для создания запросов определенных слов или словосочетаний
- Функция SQL SCORE - для получения оценки релевантности найденного текстового документа

Скалярные функции текстового поиска CONTAINS и SCORE гладко интегрируются в SQL. Можно использовать эти функции поиска в тех же местах, где бы вы использовали стандартные выражения SQL в запросах SQL. Скалярная функция SQL SCORE возвращает показатель соответствия текстовых документов заданному условию текстового поиска. Выражение SELECT запроса SQL определяет, какая информация возвращается.

Функция CONTAINS выполняет поиск совпадений слова или словосочетания, и ее можно использовать с символами подстановки для поиска совпадений подстроки аналогично предикату SQL LIKE; кроме того, ее можно использовать для поиска точных совпадений строки аналогично оператору SQL =. Однако есть важные различия между использованием функции CONTAINS и предиката SQL LIKE или оператора =. Предикат LIKE и оператор = выполняют поиск шаблонов в документе, тогда как CONTAINS использует лингвистическую обработку, то есть выполняет

поиск различных словоформ термина поиска. Например, даже если символы подстановки не используются, поиск термина work также возвращает документы со словами working и worked. Кроме того, можно добавить словарь синонимов в индекс текстового поиска, расширяя тем самым область поиска. Например, можно сгруппировать слова laptop и ThinkPad, чтобы они возвращались как результаты поиска для notebook computers. Для документов XML синтаксис аргументов поиска XML позволяет искать текст внутри тегов и атрибутов. Кроме того, поиск XQuery учитывает регистр.

Обратите внимание, что оптимизатор DB2 оценивает, сколько текстовых документов предположительно может соответствовать предикату CONTAINS и насколько различны по затратам будут различные планы доступа. Оптимизатор выбирает наименее затратный план доступа.

Функция xmlcolumn-contains - это встроенная функция DB2, возвращающая документы XML из столбца данных XML DB2 на основе текстового поиска, выполняемого механизмом DB2 Text Search. Можно использовать xmlcolumn-contains в выражениях XQuery для извлечения документов на основе поиска определенных элементов документа. Например, если документ XML содержит описание продаваемых игрушек и цены, можно использовать xmlcolumn-contains в выражении XQuery для поиска по элементам описания и цен и возврата только тех документов, которые содержат терм outdoors, не содержат pool и стоят меньше 25,00 долларов.

Есть важные различия между использованием функции xmlcolumn-contains и функции XQuery contains. Функция XQuery contains выполняет поиск подстроки внутри строки; она ищет точное совпадение с термином поиска или словосочетанием. Функция XQuery xmlcolumn-contains похожа на функцию CONTAINS, но она работает только со столбцами XML. Кроме того, она возвращает документы XML, содержащие терм поиска или словосочетание, тогда как contains возвращает только значение (1, 0 или NULL), указывающее, найден ли терм поиска.

Задание локали

При выполнении поиска по индексу текстового поиска рекомендуется всегда использовать в поисковом запросе опцию **QUERYLANGUAGE** для указания, какую локаль (сочетание языка и информации о территории) использовать для интерпретации термина поиска. Например, если есть поисковый терм "bald", можно указать рассматривать его как английское слово ("лысый"), задав **QUERYLANGUAGE=en_US** в поисковом запросе. Аналогично, если требуется интерпретировать это слово как немецкое ("скоро"), можно задать для **QUERYLANGUAGE** значение de_DE. Однако необходимо иметь в виду, что возвращаемые результаты в значительной мере зависят от значения LANGUAGE, используемого для индексирования, независимо от **QUERYLANGUAGE**, заданного в запросе.

Если в поисковом запросе значение **QUERYLANGUAGE** не задано, используется следующая логика:

- Терм поиска рассматривается как принадлежащий локали, заданной для соответствующего текстового индекса при создании индекса.
- Если при создании индекса для него была задана локаль AUTO, по умолчанию используется английская локаль (en_US) и поисковый терм рассматривается как английское слово.

Ограничения:

- Если локаль, заданная в поисковых запросах, недопустима (например, **QUERYLANGUAGE=Mongolian**), запрос будет считаться недопустимым и возникнет исключительная ситуация.

- Значение **QUERYLANGUAGE=AUTO** в поисковом запросе не поддерживается, результаты такого запроса не определены.

Обратите внимание на то, что локаль, заданная опцией **QUERYLANGUAGE**, не влияет на локаль сообщений об ошибках, создаваемых для запросов. Локаль сообщений об ошибках зависит от того, запущены ли службы экземпляра текстового поиска. Если службы не запущены, сообщения создаются с использованием локали **en_US**; если службы запущены, сообщения создаются с использованием локали среды, в которой запущена команда **db2ts START FOR TEXT**.

Ограничение на применение текстового поиска к таблицам, полученным как LEFT OUTER JOIN и FULL OUTER JOIN

Предикаты текстового поиска используют объединение первичного ключа для интеграции результатов, получаемых с помощью индекса текстового поиска, с базовой таблицей. Функции **CONTAINS** и **SCORE** нельзя использовать с **LEFT OUTER JOIN** или **FULL OUTER JOIN**, так как первичные ключи могут быть недоступны для интеграции во время выполнения плана запроса.

В средах MPP индексы текстового поиска размещаются совместно с базовыми таблицами. Применяя эти функции, убедитесь в том, что строки из таблицы с текстовым индексом не перемещены в другие узлы.

1. Первичный ключ при проектировании может попасть вне объединяемой таблицы, и объединение не удастся установить.
2. Что более важно, в среде MPP текстовый индекс размещается совместно с базовой таблицей. Чтобы функция поиска вернула правильные результаты, при применении этих функций проверьте, что строки из таблицы с текстовым индексом не были перемещены в другие узлы. Наиболее надежный способ - потребовать, чтобы функция текстового поиска выполнялась непосредственно после доступа к базовой таблице.

Планирование запросов DB2 Text Search

При планировании запроса текстового поиска необходимо принять во внимание несколько вопросов.

Аргументы текстового поиска DB2 (DB2 Text Search)

Использование символов подстановки и предел их раскрытия, регистрозависимость аргументов или свойства опций аргумента влияют на поведение запроса и часто - на производительность.

Символ подстановки

Использование символа подстановки в начале термина поиска замедляет обработку запроса. Старайтесь по возможности не прибегать к поиску вида **термин-поиска* или *?термин-поиска*.

Примечание: Если символ **?** используется, когда термин не заключен в двойные кавычки, этот термин интерпретируется как необязательный. Необязательный маркер не влияет на производительность.

Предел раскрытия символов подстановки

Если термин запроса содержит символ подстановки, то для получения соответствующих документов для термина запроса выполняется раскрытие. Если число различных соответствующих терминов в собрании текстовых индексов превышает заданный предел раскрытия символов подстановки, возвращаются только

те документы, которые соответствуют уже раскрытым терминам. Это ограничение особенно важно для символа подстановки *. Чтобы уменьшить число возможных различных соответствий, включите в термин поиска больше символов.

По умолчанию может быть возвращено 1024 термина. Чтобы изменить этот предел, задайте параметр **queryExpansionLimit** со значением в файле <ECMTS_HOME>\config\config.xml. Например, для задания предела 4096 добавьте в этот файл такую строку: <queryExpansionLimit>4096</queryExpansionLimit>

Учет регистра

Аргументы текстового поиска регистронезависимы. Это верно даже в том случае, когда вы задаете точный термин или фразу при помощи двойных кавычек. Например, при поиске термина "Орел" может быть найден и город Орел, и птица орел.

Опции аргумента поиска

Опции аргумента поиска задаются как свойства аргумента поиска. Например, в следующем запросе поиска слово bank ищется дважды, на английском и на немецком, и эти два вызова поиска выполняются с разными аргументами:

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

Множественные предикаты DB2 Text Search

Если запрос содержит несколько предикатов, учитывайте следующие ограничения, зависящие от организации предикатов.

Операции UNION и OR

Можно повысить производительность запроса, используя для объединения нескольких предикатов UNION, а не OR.

Ограничение при использовании JOIN

Функции CONTAINS и SCORE не поддерживаются как предикаты в LEFT OUTER JOIN или FULL OUTER JOIN. Использование функций поиска для стороны внешнего объединения, не сохраняющей порядок, приведет к ошибке.

Функция SCORE DB2 Text Search

Оценка документа - динамическая.

При изменениях в документе, а также при добавлении и удалении других документов в индексе текстового поиска оценка документа для термина поиска может изменяться.

Пусть есть набор документов о транспорте и загрязнении. Если вы хотите найти документы, содержащие ссылки на оба эти термина, но при этом термин загрязнение важнее для вас, чем термин транспорт, можно использовать следующую команду:

```
SELECT id_документа  
FROM библиотека_документов  
WHERE SCORE(содержимое_документа, 'загрязнение') >  
SCORE(содержимое_документа, 'транспорт')  
and CONTAINS(document_content, 'транспорт загрязнение') = 1
```

Чтобы повысить производительность, запрос можно сформулировать с использованием модификатора значимости (^) так, что функция поиска будет выполнена только один раз:

```
SELECT id_документа
FROM библиотека_документов
WHERE SCORE(содержимое_документа, 'загрязнение^10 транспорт') > 0
```

При низкой оценке загрязнение первый запрос не возвращает никаких результатов. Во втором запросе более высокая важность придается термину загрязнение, но тем не менее он возвращает результаты, даже если оценка термина загрязнение во всех документах низка.

Функция RESULTLIMIT DB2 Text Search

Чтобы несколько экземпляров **RESULTLIMIT** в одном запросе давали предсказуемые результаты, необходимо, чтобы у них был один и тот же аргумент поиска.

Описание

Если вы включаете в один запрос несколько текстовых поисков с функцией **RESULTLIMIT**, используйте общий аргумент поиска. Если использовать разные аргументы текстового поиска, результаты могут оказаться непредвиденными.

Например, в следующем запросе **RESULTLIMIT** отнюдь не гарантирует, что будет возвращено именно 10 документов:

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

Вместо этого используйте **RESULTLIMIT** так:

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

Заметим, что этот метод работает только тогда, когда обе функции **CONTAINS** применяются к одному и тому же столбцу таблицы. Если они применяются к разным столбцам, попробуйте для повышения производительности использовать **FETCH FIRST n ROWS**.

Поиск в индексах текстового поиска

Для поиска в индексах текстового поиска можно использовать оператор SQL или XQuery.

Процедура

Чтобы найти в индексе текстового поиска конкретное слово или словосочетание, используйте один из следующих способов:

- Поиск при помощи SQL.

Для поиска в индексе текстового поиска конкретного слова или словосочетания при помощи оператора SQL используется функция **CONTAINS**:

```
db2 "SELECT имя-столбца FROM имя-таблицы WHERE CONTAINS (...)"
```

Например, следующий запрос выполняет поиск в таблице **PRODUCT** наименований и цен различных снеговых лопат (snow shovel):

```
db2 "SELECT NAME, PRICE FROM PRODUCT WHERE CONTAINS
(NAME, 'snow shovel') = 1"
```

- Поиск при помощи XQuery.

Для поиска в индексе текстового поиска конкретного слова или словосочетания при помощи XQuery используется функция `db2-fn:xmlcolumn-contains()`.

Например, следующий запрос выполняет поиск в таблице `PRODUCT` наименований и цен различных снеговых лопат (`snow shovel`):

```
db2 "xquery for \${info} in db2-fn:xmlcolumn-contains
('PRODUCT.DESCRPTION','snow shovel')
return <result> {\${info}/description/name, \${info}/description/price} </result>"
```

Примечание: В зависимости от используемой вами оболочки операционной системы перед знаком доллара в информации о переменных может потребоваться другой эскейп-символ. В предыдущем примере использовалась обратная дробная черта (`\`), применяемая в качестве эскейп-символа в операционных системах UNIX.

Поиск в индексах текстового поиска при помощи SCORE

Функция `SCORE` позволяет определить, насколько документ соответствует аргументу поиска.

Об этой задаче

Функция `SCORE` возвращает число с плавающей запятой двойной точности от 0 до 1, указывающее степень соответствия документа критериям поиска. Чем точнее документ соответствует запросу, тем больше релевантность оценки и значение результата.

Оценка вычисляется динамически на основе содержимого собрания текстовых индексов на момент запроса, и имеет смысл только для однораздельного текстового индекса.

Пример

Предположим, есть набор документов, содержащих слова `транспорт` и `загрязнение`. Если вы хотите найти документы, содержащие ссылки на оба эти термина, но при этом термин `загрязнение` важнее для вас, чем термин `транспорт`, можно использовать следующую команду:

```
SELECT id_документа
FROM библиотека_документов
WHERE SCORE(содержимое_документа, 'загрязнение') >
SCORE(содержимое_документа, 'транспорт')
and CONTAINS(document_content, 'транспорт загрязнение') = 1
```

Однако для повышения производительности запрос можно переписать, применив модификатор значимости (`^`), чтобы функция поиска выполнялась только один раз:

```
SELECT id_документа
FROM библиотека_документов
WHERE SCORE(содержимое_документа, 'загрязнение^10 транспорт') > 0
ORDER BY SCORE(содержимое_документа, 'загрязнение^10 транспорт') DESC
```

При низкой оценке `загрязнение` первый запрос не возвращает никаких результатов. Во втором запросе более высокая важность придается термину `загрязнение`, но тем не менее он возвращает результаты, даже если оценка термина `загрязнение` во всех документах низка.

Аналогично, чтобы найти в базе данных SAMPLE число сотрудников, указавших в своих резюме, что они знают, как программировать на языке Java или COBOL, можно ввести следующий запрос:

```
SELECT EMPNO, INTEGER(SCORE(RESUME, 'программист AND (java OR cobol)') * 100)
AS RELEVANCE FROM EMP_RESUME WHERE RESUME_FORMAT = 'ascii'
ORDER BY RELEVANCE DESC
```

Однако лучше использовать следующий запрос с функцией CONTAINS. Оптимизатор DB2 сначала оценивает предикат CONTAINS в условии WHERE и тем самым избегает оценки функции SCORE в списке SELECT для каждой строки таблицы. Заметим, что это возможно, только если аргументы SCORE и CONTAINS в запросе идентичны.

```
SELECT EMPNO, INTEGER(SCORE(RESUME, 'программист AND (java OR cobol)') * 100)
AS RELEVANCE FROM EMP_RESUME WHERE RESUME_FORMAT = 'ascii'
AND CONTAINS(RESUME, 'программист AND (java OR cobol)') = 1
ORDER BY RELEVANCE DESC
```

Повышение производительности для полнотекстовых запросов

Для повышения производительности при поиске рекомендуются следующие подходы:

Процедура

- Избегайте использования функции SCORE без функции CONTAINS. Кроме того, во избежание дублирования обработки убедитесь, что строка (аргумент и все опции поиска), задаваемая вами для функции CONTAINS, в точности совпадает (включая пробелы) со строкой, используемой вами для функции SCORE.
- Убедитесь, что у компилятора DB2 есть верная статистика таблиц. Для обновления статистики используйте команду **RUNSTATS**.

Синтаксис аргументов текстового поиска

Аргумент поиска состоит из одного или нескольких терминов и необязательных параметров поиска через пробел, заданных для поиска в текстовых документах.

Для простого поиска достаточно ввести один или несколько терминов. Механизм поиска возвращает документы, которые содержат все термины, а также, по умолчанию, варианты этих терминов. Например, если выполнить поиск по термину king (король), будут возвращены документы, содержащие слова king (король) и kings (короли). При поиске по двум терминам поисковый механизм возвращает только документы, содержащие оба термина. Чтобы выполнить поиск терминов как точного словосочетания, добавьте кавычки.

Примечание:

- Поиск выполняется без учета регистра, поэтому при поиске в русском тексте точного термина "ГНИ" может вернуть ГНИ или гни.
- Запросы текстового поиска не должны превышать пределы запросов SQL DB2.

Чем точнее указан поисковый термин, тем более точными будут результаты. Однако уточнить поиск можно и с помощью следующих опций:

- Логические операции. Используйте операции AND и OR для поиска по обоим терминам или как минимум по одному термину.
 - AND - это операция конъюнкции, она подразумевается по умолчанию, то есть если между двумя терминами не указана логическая операция, используется AND.

- Если термины связаны операцией OR, ищется документ, в котором есть любой из них.
- Модификаторы встречаемости. Используйте +, NOT или -, чтобы указать, что термины обязательны или запрещены.
 - Модификатор + отличается от операции AND, так как модификатор + указывает, что второй термин должен быть точным соответствием – то есть не может быть синонимом.
-
- Задание значимости. Используйте ^, чтобы повысить важность встречаемости указанного термина (имеет значение для функции SCORE).
 - ^ увеличивает значимость термина или словосочетания, указанного перед ним, если указанное число больше 1. Чтобы уменьшить ранг термина или словосочетания в возвращаемом списке, укажите число больше 0, но меньше 1.
 - Для правильной работы модификатора задания значимости используйте его с условием ORDER BY и функцией SCORE.
- Символы подстановки. Используйте ? в качестве замены одного символа или * в качестве замены любого числа символов, добавляемых к терминам поиска, чтобы поддерживались варианты написания и увеличилась область поиска.
 - Символ * указывает, что к термину можно прибавить любое число символов.
 - Символ подстановки * в начале термина поиска отрицательно влияет на производительность поискового запроса.
 - Символ ? указывает, что к термину можно прибавить один символ.
 - Чтобы символ ? воспринимался как символ подстановки, используйте двойные кавычки. Иначе он будет восприниматься как модификатор *необязательный_термин*, описанный в следующем пункте.
- Необязательные термины. Используйте ? в начале термина или фразы, чтобы указать на их необязательность
- Эскейп-символ: Используйте символ \, чтобы специальные символы, а именно:

<	>	&&		!
()	!	=	"
{	}	~	*	?
[]	:	\	-

анализировались как часть запроса.

- Точный поиск. Чтобы возвращались только точные соответствия, заключите термин или словосочетание поиска в двойные кавычки " ".
- Скобки. Используйте (), чтобы термины поиска (и их взаимосвязь) трактовались как единый элемент.

Используйте для запросов поиска XML, направляемых в анализатор XML, подмножество языка XPath с использованием неанализируемых терминов. Анализатор запросов распознает неанализируемый терм по синтаксису, который используется в запросе.

Если при поиске нужна обработка с учетом языка, для параметра аргумента поиска подразумевается локаль. Язык этого запроса соответствует локали индекса текстового поиска, используемого при работе функции поиска.

Для аргументов поиска используется следующий синтаксис:

Аргумент поиска

Модифицированное_условие ((Операция) (Модифицированное_условие))

Операция

AND | OR

Модифицированное_условие

(Модификатор **Условие** (^число))

Модификатор

+ | - | NOT

Условие

неспецифицированный термин | неанализируемый термин.

Примечание:

- Неспецифицированный термин - это термин или словосочетание. Термин может быть словом, например, king; точным словом, например, "king"; или же словом с символом подстановки, например, king* или king?. Аналогично, словосочетание может быть группой слов, например, cabbages and kings; точным словосочетанием, например, "The King and I"; или словосочетанием, содержащим символ подстановки, например, "all the king's ho?ses" или "all the king's *".
- Неанализируемые термины поиска не анализируются лингвистическим анализатором запросов; неанализируемые запросы идентифицируются по их синтаксису. Неанализируемый термин, используемый для запросов текстового поиска - @xpath: например, @xpath: '/TagA/TagB[. contains("king")]'

Примеры

Таблица 2. Логические операции для поисковых запросов

Оператор	Пример	Результаты запросов
AND	King AND Lear King Lear	Возвращает все документы, содержащие термины King и Lear. Если включен словарь синонимов, могут быть также возвращены такие слова, как monarch.
OR	Hamlet OR Othello	Возвращает документы, содержащие Hamlet или Othello.

Таблица 3. Модификаторы встречаемости

Модификатор	Пример	Результат запроса
NOT -	Hamlet NOT Othello Hamlet -Othello	Возвращает документы, которые содержат Hamlet, но не содержат Othello.
+	Lear + King	Возвращает все документы, содержащие термины Lear и King. Документы, содержащие Lear и monarch, не возвращаются.

Таблица 4. Другие модификаторы

Модификатор	Пример	Результат запроса
<i>термин1</i> или <i>словосочетание1</i> ^ <i>число</i> <i>термин2</i> или <i>словосочетание2</i>	Hamlet^2 Othello Hamlet Othello^5	Возвращает документы, содержащие Hamlet и Othello, но более важным считается термин Hamlet. В обоих примерах запросов каждому найденному соответствию термина Hamlet назначается в два раза большая значимость, чем каждому найденному соответствию термина Othello.
*	king* k*ng *ing	Возвращает документы, содержащие возможные варианты поискового термина, содержащего символ подстановки. Этот пример запроса может вернуть такие результаты, как king и kingdom в первом примере, king и kissing во втором примере и king и skiing в третьем примере.
?	mea? be?n "?ean"	Возвращает документы, содержащие возможные варианты поискового термина, содержащего символ подстановки. Будут возвращены такие результаты, как meal и mean в первом примере, bean и been во втором примере и mean и bean в третьем примере.
? в начале	King James ?Edition	Возвращает документы, содержащие и king, и james, но edition - необязательный термин.
"словосочетание" "точный термин" "словосочетание с символом подстановки"	"King Lear" "king" "John * Kennedy" "John ? Kennedy"	Возвращает документы, содержащие в точности слово или словосочетание King Lear в первом примере, и слово king без таких вариантов, как kings или kingly, во втором примере. С символами подстановки можно использовать кавычки, чтобы возвращать такие соответствия, как John Kennedy со средним именем или инициалами или без них в третьем примере, и John <i>инициалы</i> Kennedy в четвертом примере.
()	(Hamlet OR Othello) AND plays	Возвращает документы, содержащие следующие термины: <ul style="list-style-type: none"> • Термин Hamlet или Othello • Термин plays
\	\(1+1\):2	Возвращает документы, содержащие (1+1):2. Символ \ служит для того, чтобы следующие за ним символы не считались специальными символами синтаксиса запросов.

Поиск документов XML при помощи DB2 Text Search

DB2 Text Search поддерживает использование поиска XML для поиска документов XML.

Используя подмножество языка XPath с расширениями для текстового поиска, поиск XML позволяет индексировать документы XML и проводить в них поиск. Вы можете использовать в запросах структурные элементы (имена тегов, имена атрибутов и значения атрибутов) отдельно или в сочетании со свободным текстом.

Основные возможности поиска XML:

Структурный поиск XML

Используя в запросах текстового поиска синтаксис поиска XML, вы можете искать в документах XML структурные элементы (теги, имена атрибутов и значения атрибутов) и текст, находящийся в области действия этих элементов. Обратите внимание на то, что простой поиск не обрабатывает поля атрибутов в документе XML.

Лексический анализ запроса XML

Текст, используемый в выражении предиката поиска XML в качестве термов запроса XML, при лексическом анализе обрабатывается так же, как текст в термах других запросов, за тем исключением, что не поддерживаются исправления ввода, термы в поле (fielded terms) и вложенные термы поиска XML. Поддерживаются синонимы, символы подстановки, фразы и лемматизация.

Игнорирование пространств имен XML

Префиксы пространства имен не сохраняются при индексации имен тегов и атрибутов XML. Документы XML, в которых объявляются и используются пространства имен, можно индексировать, и в них можно будет проводить поиск, но префиксы пространств имен игнорируются при индексации и удаляются из запросов поиска XML.

Числовые значения

Поддерживаются предикаты, в которых значения атрибутов сравниваются с числами.

Полное соответствие

Операция = (знак равенства) со строчным аргументом в предикате требует полного соответствия всех маркеров в строке всем маркерам в указанной области текста, при этом порядок существен.

Отсутствие доступа UIMA

Архитектура управления неструктурированной информацией (Unstructured Information Management Architecture - UIMA) используется для лексического анализа в поиске XML, но написанные пользователями аннотации не поддерживаются.

Подмножество XPath, реализованное в поиске XML, отличается от стандартного XPath следующим:

- Оно не поддерживает итераций и диапазонов в выражениях для пути.
- Исключается фильтрация выражений, то есть фильтрация допускается только в выражениях предикатов, но не в выражении для пути.
- Он запрещает имена с абсолютным путем в выражениях предикатов
- Реализуется только один тег axis, распространение разрешено только в направлении вперед.

В следующей таблице приведен список некоторых допустимых запросов поиска XML.

Таблица 5. Допустимые запросы поиска XML

Запрос	Описание
/	Корневой узел; любой документ
/sentences	Все документы с тегом верхнего уровня sentences
//sentences	Все документы с тегом любого уровня sentences
sentences	Все документы с тегом любого уровня sentences
/sentence/paragraph	Все документы с тегом верхнего уровня sentences и тегом - непосредственным потомком paragraph.
/sentence/paragraph/	Все документы с тегом верхнего уровня sentences и тегом - непосредственным потомком paragraph.
/book/@author	Любой документ с тегом верхнего уровня book, у которого есть атрибут author.
/book//@author	Любой документ с тегом верхнего уровня book, у которого есть тег-потомок любого уровня с атрибутом author.
/book[@author contains("barnes") and @title contains("lemon")]	Выводится любой документ с тегом верхнего уровня book, в котором заданы атрибуты author и title со значениями, содержащими нормализованные строки
/book[@author contains("barnes") and (@title contains("lemon") or @title contains("flaubert"))]	Любой документ с тегом верхнего уровня book, в котором задан атрибут author и один из двух указанных атрибутов title.
/program[. contains("hello, world.")]	Любой документ с тегом верхнего уровня program, который содержит по крайней мере маркеры hello и world
/book[paragraph contains("flaubert")]//sentence	Любой документ с тегом верхнего уровня book, у которого, во-первых, есть тег - непосредственный потомок paragraph, содержащий "flaubert", и, во-вторых, тег-потомок sentence на любом уровне со ссылкой на тег book
/auto[@price <30000]	Любой документ с тегом верхнего уровня auto, у которого есть атрибут price с числовым значением меньше 30000.
//microbe[@size < 3.0e-06]	Любой документ, содержащий тег microbe на любом уровне с атрибутом size, значение которого меньше 3.0e-06

Примечание: В синтаксисе поиска XML не поддерживается:

- /*
- //*
- /@*
- //@*

Простой поиск не обрабатывает поля атрибутов в документе XML.

Конфигурирование поиска XML для DB2 Text Search

Можно сконфигурировать некоторые параметры, используемые для поиска XML.

Все параметры конфигурации анализатора расположены в файле `parser_config.xml` в элементе XML, определяющем анализатор, `com.ibm.es.nuvo.parser.xml.XMLParser`. Каждый параметр задан элементом `Parameter` в следующей форме:

```
<Parameter Name="параметр">значение</Parameter>
```

Если элементов `parameter` нет, используются параметры по умолчанию. В следующем списке приведены допустимые имена и значения параметров:

UnresolvableEntityResponse

Этот элемент позволяет DB2 Text Search обрабатывать неразрешимые внешние объекты. Даже в том случае, если анализатор XML DB2 Text Search не использует DTD для проверки, по крайней мере одно DTD может потребоваться для разрешения объектов. Используйте следующие параметры, чтобы задать, как должен действовать анализатор в случае отсутствия DTD или обнаружения неразрешимого объекта:

ignore Документ принимается для анализа, но внешне определенные элементы, которые нельзя разрешить, во время анализа игнорируются без вывода сообщений, а не индексируются в неразрешенной форме. Это значение по умолчанию.

reject Документ отклоняется с кодом состояния 404.

handleExternalFiles

Документы XML могут ссылаться на другие файлы, содержащие DTD, схему, определение пространства имен и другое содержание. Такие файлы обычно располагаются в сети, и их скачивание увеличивает время анализа для документов XML и нагрузку ввода-вывода на компьютере, где запущена система поиска. Большая часть информации в таких внешних файлах - это метаданные, задающие грамматические свойства элементов разметки и ничего не добавляющие в индекс поиска. Параметр **handleExternalFiles** задает, скачивать ли такие файлы, или рассматривать их как пустые без чтения. Можно задать следующие параметры:

ignore Внешние файлы не скачиваются; они рассматриваются как пустые. Анализ ускоряется, но внешне определенные объекты нельзя индексировать. Это значение по умолчанию.

read Если внешние файлы доступны, они скачиваются, из-за чего время анализа растет. Если файл недоступен, анализатор выдает исключительную ситуацию ввода-вывода, а URI обрабатываемого документа индексируется с кодом ошибки.

handleSkippedEntities

Документы XML могут ссылаться на определенные где-либо еще объекты. Если анализатор не может разрешить какой-либо из этих объектов, параметр конфигурации **handleSkippedEntities** задает, отклонять ли документ (не индексировать его), или игнорировать объект и индексировать остаток документа.

ignore При анализе неразрешенные ссылки игнорируются без вывода сообщений. Это значение по умолчанию.

reject Содержащий неразрешенный объект документ отклоняется, а в индексе для его URI сохраняется запись об ошибке.

titleTagNameList

Если документ XML содержит тег title (регистр символов неважен), текстовое содержимое этого тега, если оно есть, индексируется как содержимое поля title для этого документа и таким образом доступно для поиска. Если документ XML не содержит тега title, его URL сохраняется как доступный для поиска по полям заголовков. Чтобы использовать другой тег, любой из нескольких тегов или вообще не использовать теги для идентификации заголовков документов в целях поиска по полю, можно определить параметр **titleTagNameList** со следующими опциями. В случае нескольких совпадений только содержимое первого из совпадений индексируется как заголовок документа.

отсутствует

Если этого параметра нет в файле конфигурации, для идентификации содержания заголовка используется только тег <title>. Это значение по умолчанию.

пусто Если этот параметр присутствует в файле конфигурации, но не задает ни одного тега, теги не используются для идентификации содержания заголовка.

имя-тега

Для идентификации заголовков документов используется тег, совпадающий с заданным именем (регистр неважен).

имя-тега1, имя-тега2, ...

Для каждого индексируемого документа первый тег, совпадающий с любым именем в списке (без учета регистра), используется для идентификации заголовка документа.

Синтаксис поиска для документов XML

Используя выражения поиска XML, можно при помощи механизма DB2 Text Search выполнять поиск в заданных секциях документа XML в столбце XML в DB2.

Синтаксис

►►@xpath: ' | Запрос поиска XML | ' ◀◀

Запрос поиска XML:

|—путь_положения—| [—предикат_поиска—]

@xpath:

Ключевое слово, с которого начинается запрос текстового поиска в документе XML.

Запрос поиска XML

Запрос текстового поиска, используемый DB2 Text Search для поиска в документах XML. Этот запрос указывается в одинарных кавычках. Поисковый запрос XML - это выражение поиска XML, состоящее из пути размещения, указывающего участок документа XML, где надо выполнять поиск, и необязательный предикат, указывающий критерии поиска.

путь_положения

Выражение поиска XML, использующее подмножество сокращенного синтаксиса

XPath для указания узла или атрибута в документе XML. Дополнительную информацию смотрите в разделе "Путь положения".

предикат_поиска

Необязательные критерии поиска, используемые DB2 Text Search при поиске в документе XML. Дополнительную информацию смотрите в разделе "Предикат поиска".

Механизм DB2 Text Search возвращает документ XML, если находит текст, указанный в параметре *предикат_поиска*, в указанных узлах или атрибутах документа XML.

Путь положения

При выполнении текстового поиска в документе XML DB2 Text Search использует локальные имена узлов и атрибутов, а также подмножество синтаксиса XPath для указания узлов и атрибутов в документе XML. DB2 Text Search поддерживает следующие элементы поиска XML:

- Локальные имена узлов или атрибутов
- . (точка) как узел текущего контекста
- / или // как символ-разделитель
- @ как сокращенный символ для атрибута

Нормализация имен: имена узлов и атрибутов XML не нормализуются при индексации для использования механизмом DB2 Text Search : они не переводятся в нижний регистр, не подвергаются лексическому анализу, и вообще никак не изменяются. В именах узлов и атрибутов XML регистр учитывается, так что для успешного поиска строки, используемые в качестве этих имен в запросах, должны в точности совпадать с именами в документах.

Обработка пространств имен: При создании индекса текстового поиска можно использовать документы XML, содержащие спецификаторы пространств имен XML, но спецификаторы пространств имен не сохраняются в индексе. Например, тег `<nsdoc:heading>` индексируется как просто `heading`, а термин запроса `@xpath:'/nsdoc:heading'` при анализе заменяется на `@xpath:'/heading'`. При анализе запросов префиксы пространств имен XML отбрасываются.

Примеры

Ниже приведен пример допустимого запроса текстового поиска, использующего поиск XML для нахождения термина `snow shovel` (снеговая лопата) в узле описания информации о продукте:

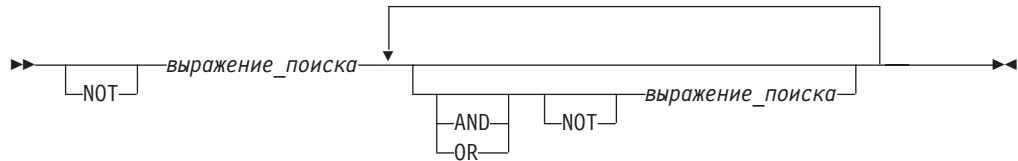
```
@xpath:'/info/product/description[. contains("snow shovel")]'
```

Следующий пример не является допустимым запросом текстового поиска, использующего поиск XML, так как в нем используется `".."`, сокращение поиска XML для `parent::node()`:

```
@xpath:'/info/product/description/..@ID[. contains("A2")]'
```

Предикат поиска

Синтаксис



выражение_поиска

Запрос поиска XML DB2 Text Search. DB2 Text Search использует выражение поиска для поиска значений узлов или атрибутов в документе XML.

Для создания выражений поиска можно использовать следующие операции:

- Логические операции: AND, OR и NOT
- Операции содержания: contains и excludes
- Операции сравнения: =, >, <, >=, <= и !=

Примечание:

Операции сравнения применимы только к значениям атрибутов; они неприменимы к значениям узлов.

Таким образом, для `<root><aaa id="10">100</aaa><aaa id="11">101</aaa></root>` следующий запрос недопустим:

```
select id from testtable where contains(item,'@xpath: '/root/aaa[. > 20]')>0
```

Пример допустимого запроса:

```
select id from testtable where contains(item,'@xpath: '/root/aaa/@id[. > 20]')>0
```

Можно создавать сложные выражения поиска, сочетая операции сравнения и содержания с помощью логических операторов AND, OR и NOT. Кроме того, для группировки выражений можно использовать скобки.

Строки задаются в одинарных или двойных кавычках. Строку, содержащую кавычки, нельзя заключать в кавычки того же типа. Например, строка, заключенная в одинарные кавычки, не может содержать символ одинарной кавычки.

Приоритет операций: В предикатах поиска XML операции сравнения предшествуют логическим операциям, а все логические операции имеют равный приоритет. Чтобы задать нужный порядок вычисления операций, используйте скобки.

Нормализация значений атрибутов и значений узлов: Свободный текст в документах XML (расположенный не внутри тега, а между тегами) и значения атрибутов нормализуются перед индексированием. Свободный текст в запросах XML (в операторах содержания) нормализуется таким же образом, как и в других запросах (не XML).

Пример

Ниже приведен пример использования запроса поиска XML для поиска продуктов, в описании которых содержится термин snow shovel (снеговая лопата) и цена которых ниже 29,99 долларов.

```
@xpath: '/info/product [(description contains("snow shovel")) and (@price < 29.99)]'
```

Выражения сравнения

Выражения сравнения сравнивают значение атрибута с указанным значением.

Синтаксис

►—*выражение_пути*—*операция*—*литерал*—►

выражение_пути

Выражение пути, использующее подмножество сокращенного синтаксиса поиска XML для указания узла или атрибута.

операция

Тип выполняемого сравнения. Возможные типы операций:

- = *выражение_пути* равно *литерал*.
- > *выражение_пути* больше, чем *литерал*.
- < *выражение_пути* меньше, чем *литерал*.
- >= *выражение_пути* больше либо равно *литерал*.
- <= *выражение_пути* меньше либо равно *литерал*.
- != *выражение_пути* не равно *литерал*.

литерал

Строка или число, сравниваемое со значением узла или атрибута *выражение_пути*.

Строки нужно заключать в одинарные или двойные кавычки. Строку, содержащую кавычки, нельзя заключать в кавычки того же типа. Например, строка, заключенная в одинарные кавычки, не может содержать символ одинарной кавычки. Используйте символ обратной дробной черты (\) как эскейп-символ для двойных кавычек (").

Если строка содержит двойные кавычки, ее можно заключить в одинарные кавычки. В следующем примере показана строка в одинарных кавычках, содержащая двойные кавычки:

```
'he said "Hello, World"'
```

Если строка содержит одинарные кавычки, ее можно заключить в двойные кавычки, заданные при помощи эскейп-символов. В следующем примере показана строка в двойных кавычках, содержащая одинарную кавычку:

```
"the cat's toy"
```

Такие возможности DB2 Text Search, как словосочетания, символы подстановки и синонимы, не поддерживаются в запросах поиска XML.

Пример

В следующем примере используется операция = для поиска ID продуктов, совпадающих со строкой 100-200-101:

```
@xpath:'/info/product/@pid[. = "100-200-101" ]'
```

Примечание:

Строковые аргументы поддерживаются только операциями сравнения = и !=, так что использовать <, <=, >, >= нельзя. С числовыми аргументами работают все шесть операций. Числовые аргументы можно сравнивать со значениями атрибутов, но нельзя сравнивать с содержимым тегов (узлов)

Выражения содержания

Выражения содержания определяют, содержится ли указанное значение в значении узла или атрибута.

Синтаксис

► *выражение_пути* { *contains* | *excludes* } (*литерал*)

выражение_пути

Выражение поиска XML, задающее узел или атрибут XML.

contains

Выражение, указывающее, что значение *выражение_пути* содержит *литерал*.

excludes

Выражение, указывающее, что значение *выражение_пути* не содержит *литерал*.

литерал

Строка, сравниваемая со значением узла или атрибута *выражение_пути*.

Строки задаются в одинарных или двойных кавычках. Строка не может содержать тех кавычек, в которые она заключается: например, строка, заключенная в одинарные кавычки, не может содержать символ одинарной кавычки. Используйте символ обратной дробной черты (\) как эскейп-символ для двойных кавычек (").

Если строка содержит двойные кавычки, ее можно заключить в одинарные кавычки.

В следующем примере показана строка в одинарных кавычках, содержащая двойные кавычки:

```
'he said "Hello, World"'
```

Если строка содержит одинарные кавычки, ее можно заключить в двойные кавычки, заданные при помощи эскейп-символов. В следующем примере показана строка в двойных кавычках, содержащая одинарную кавычку:

```
"the cat's toy"
```

Такие возможности DB2 Text Search, как словосочетания, символы подстановки и синонимы, не поддерживаются в запросах поиска XML.

Пример

В следующем примере используется сокращенный синтаксис XQuery для выражений пути, чтобы указать, что узел `description` не содержит термина `ice scraper`:

```
@xpath:'/info/product/description[. excludes('ice scraper')]'
```

Глава 7. Встроенные функции поиска SQL и XML

Можно использовать следующие встроенные функции поиска DB2 в DB2 Text Search. Схема этих функций - SYSIBM.

CONTAINS

Возвращает NULL или значение INTEGER, равное 0 или 1, в зависимости от того, соответствует ли входной текстовый документ условию текстового поиска

SCORE

Возвращает NULL или значение DOUBLE от 0 до 1, указывающее, насколько текстовый документ отвечает критериям поиска.

xmlcolumn-contains

Возвращает NULL или значение INTEGER, равное 0 или 1, в зависимости от того, соответствует ли входной текстовый документ типа данных XML условию текстового поиска

функция CONTAINS

Функция CONTAINS выполняет поиск по индексу текстового поиска при помощи заданных вами в аргументе поиска критериев и возвращает значение, которое показывает, найдено ли совпадение.

Синтаксис функции

►► CONTAINS (—имя-столбца—, —аргумент-поиска—)

► [(1)]

► [, —строчная-константа—]

Примечания:

- 1 *строчная-константа* должна соответствовать правилам для опций-аргумента-поиска.

опции-аргументов-поиска:

(1)

QUERYLANGUAGE	==	—локаль—
RESULTLIMIT	==	—значение—
SYNONYM	==	—OFF— —ON—

Примечания:

- 1 Нельзя задавать несколько раз одно и то же условие.

Схема - SYSIBM.

Параметры функций

имя-столбца

Специфицированное или неспецифицированное имя столбца индекса текстового поиска, в котором будет выполняться поиск. Этот столбец должен существовать в таблице или производной таблице, указанной в операторе в условии FROM, а со столбцом таблицы или базовой таблицы этой производной таблицы должен быть связан индекс текстового поиска (SQLSTATE 38N12). Базовое выражение столбца должно представлять собой простую ссылку на столбец базовой таблицы: либо непосредственную, либо через вложенную производную таблицу.

аргумент-поиска

Выражение, возвращающее строчное значение (кроме типа большого объекта), которое содержит поисковые термины; не может быть строкой из пробелов или пустой строкой (SQLSTATE 42815). Строчное значение - результат выражения не может содержать больше 4096 байт (SQLSTATE 42815). Это значение должно быть преобразовано в Unicode перед его использованием для поиска по индексу текстового поиска. Максимальное число терминов в одном запросе не может превышать 1024 (SQLSTATE 38N10).

строчная-константа

Строчная константа, задающая опции аргумента поиска, которые действуют для этой функции.

В составе *опций-аргумента-поиска* можно задать следующие опции:

QUERYLANGUAGE=*локаль*

Задает локаль, которую механизм DB2 Text Search использует при выполнении текстового поиска в текстовом столбце DB2. Значение - любая поддерживаемая локаль. Если не задать **QUERYLANGUAGE**, значением по умолчанию будет локаль индекса текстового поиска. Если в индексе текстового поиска значение параметра **LANGUAGE** - AUTO, для **QUERYLANGUAGE** используется значение en_US.

RESULTLIMIT=*значение*

Если оптимизатор выбирает план, при котором, чтобы получить оценку, механизм поиска вызывается для каждой строки набора результатов, то опция **RESULTLIMIT** на производительность не влияет. Если же механизм поиска вызывается один раз для всего набора результатов, то **RESULTLIMIT** действует как условие FETCH FIRST.

При использовании нескольких текстовых поисков, в которых **RESULTLIMIT** задается в одном и том же запросе, использовать один и тот же *аргумент-поиска*. Если вы используете разные значения *аргумента-поиска*, можно не получить ожидаемых результатов.

SYNONYM = OFF | ON

Задает, следует ли использовать словарь синонимов, связанный с индексом текстового поиска. Значение по умолчанию - OFF. Чтобы использовать синонимы, добавьте словарь синонимов в индекс текстового поиска при помощи Инструментов синонимов.

OFF Не использовать словарь синонимов

ON Использовать словарь синонимов, связанный с индексом текстового поиска.

Результат выполнения этой функции - длинное целое (large integer). Если второй аргумент допускает пустые значения, результат также допускает пустые значения; если второй аргумент пуст, результатом будет пустое значение. Если третий аргумент пуст, результат будет тем же, как и в случае, когда третий аргумент не задан. CONTAINS возвращает целое число 1, если в документе есть совпадение с критериями, заданными в аргументе поиска. Иначе возвращается 0.

Функция CONTAINS - недетерминированная.

Примечание: Используя маркеры параметров как аргумент поиска в функциях текстового поиска, надо выполнить дополнительные действия. У маркеров параметров нет типа во время прекомпиляции в программах JDBC и ODBC, но аргумент поиска в функциях текстового поиска должен при анализе разрешаться в строчное значение. Поскольку неизвестный тип маркера параметра не может разрешаться в строчное значение (SQLCODE -418), надо явно отобразить маркер параметра в тип данных VARCHAR.

Примеры

- Следующий запрос служит для поиска всех сотрудников, в резюме которых упоминается COBOL. В аргументе текстового поиска регистр символов не учитывается.

```
SELECT EMPNO
FROM EMP_RESUME
WHERE RESUME_FORMAT = 'ascii'
AND CONTAINS(RESUME, 'COBOL') = 1
```

- В следующей программе С ищется точный термин ate в столбце COMMENT:

```
char search_arg[100]; /* входная переменная хоста */
...
EXEC SQL DECLARE C3 CURSOR FOR
SELECT CUSTKEY
FROM CUSTOMERS
WHERE CONTAINS(COMMENT, :search_arg)= 1
ORDER BY CUSTKEY;
strcpy(search_arg, "ate");
EXEC SQL OPEN C3;
...
```

- Следующий запрос служит для поиска любых 10 студентов, написавших интерактивные рефераты, которые содержат словосочетание горючие полезные ископаемые на испанском языке, то есть combustible fósil. Для соответствующего индекса текстового поиска был создан словарь синонимов. Поскольку требуется только 10 студентов, запрос оптимизируется при помощи опции **RESULTLIMIT**, чтобы ограничить число результатов с базового сервера текстового поиска.

```
SELECT FIRSTNAME, LASTNAME
FROM STUDENT ESSAYS
WHERE CONTAINS(TERM_PAPER, 'combustible fósil',
'QUERYLANGUAGE= es_ES RESULTLIMIT = 10 SYNONYM=ON') = 1
```

функция SCORE

Функция SCORE выполняет поиск по индексу текстового поиска при помощи критериев, заданных вами в аргументе поиска, и возвращает оценку релевантности, показывающую степень соответствия документа запросу по сравнению с другими документами в столбце.

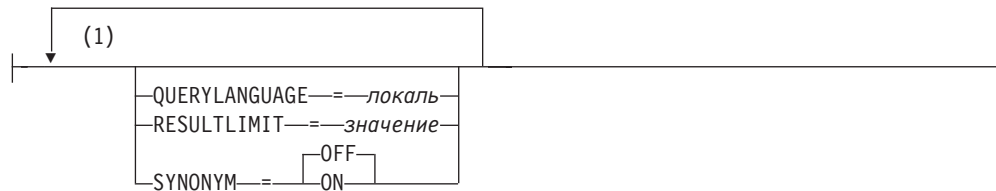
Синтаксис функции



Примечания:

- 1 *строочная-константа* должна соответствовать правилам для опций-аргумента-поиска.

опции-аргументов-поиска:



Примечания:

- 1 Нельзя задавать несколько раз одно и то же условие.

Схема - SYSIBM.

Параметры функций

имя-столбца

Специфицированное или неспецифицированное имя столбца индекса текстового поиска, в котором будет выполняться поиск. Этот столбец должен существовать в таблице или производной таблице, указанной в операторе в условии FROM, а со столбцом таблицы или базовой таблицы этой производной таблицы должен быть связан индекс текстового поиска (SQLSTATE 38N12). Базовое выражение столбца должно представлять собой простую ссылку на столбец базовой таблицы: либо непосредственную, либо через вложенную производную таблицу.

аргумент-поиска

Выражение, возвращающее строчное значение (кроме типа большого объекта), которое содержит поисковые термины; не может быть строкой из пробелов или пустой строкой (SQLSTATE 42815). Строчное значение - результат выражения не может содержать больше 4096 байт (SQLSTATE 42815). Это значение должно быть преобразовано в Unicode перед его использованием для поиска по индексу текстового поиска. Максимальное число терминов в одном запросе не может превышать 1024 (SQLSTATE 38N10).

строочная-константа

Строчная константа, задающая опции аргумента поиска, которые действуют для этой функции.

В составе *опций-аргумента-поиска* можно задать следующие опции:

QUERYLANGUAGE=*локаль*

Задает локаль, которую механизм DB2 Text Search использует при

выполнении текстового поиска в текстовом столбце DB2. Значение - любая поддерживаемая локаль. Если не задать **QUERYLANGUAGE**, значением по умолчанию будет локаль индекса текстового поиска. Если в индексе текстового поиска значение параметра **LANGUAGE** - **AUTO**, для **QUERYLANGUAGE** используется значение en_US.

RESULTLIMIT=значение

Если оптимизатор выбирает план, при котором, чтобы получить оценку, механизм поиска вызывается для каждой строки набора результатов, то опция **RESULTLIMIT** на производительность не влияет. Если же механизм поиска вызывается один раз для всего набора результатов, то **RESULTLIMIT** действует как условие **FETCH FIRST**.

При использовании нескольких текстовых поисков, в которых **RESULTLIMIT** задается в одном и том же запросе, использовать один и тот же *аргумент-поиска*. Если вы используете разные значения *аргумента-поиска*, можно не получить ожидаемых результатов.

Примечание: Если важно снизить число результатов, для этого лучше уточнить термины поиска, а не использовать **RESULTLIMIT**. **RESULTLIMIT** ограничивает число результатов независимо от их оценок, так что документы высокого ранга могут не попасть в набор результатов.

SYNONYM = OFF | ON

Задаёт, следует ли использовать словарь синонимов, связанный с индексом текстового поиска. Значение по умолчанию - **OFF**. Чтобы использовать синонимы, добавьте словарь синонимов в индекс текстового поиска при помощи Инструментов синонимов.

OFF Не использовать словарь синонимов

ON Использовать словарь синонимов, связанный с индексом текстового поиска.

Результат выполнения этой функции - число с плавающей точкой двойной точности. Если второй аргумент допускает пустые значения, результат также допускает пустые значения; если второй аргумент пуст, результатом будет пустое значение. Если третий аргумент пуст, результат будет тем же, как и в случае, когда третий аргумент не задан.

Если столбец содержит совпадение с критериями, задаваемыми аргументом поиска, результатом будет значение от 0 до 1. Чем чаще встречалось соответствие, тем больше значение результата. Если столбец не содержит соответствий, результат - 0.

Функция **SCORE** - недетерминированная.

Примечание: Используя маркеры параметров как аргумент поиска в функциях текстового поиска, надо выполнить дополнительные действия. У маркеров параметров нет типа во время прекомпиляции в программах JDBC и ODBC, но аргумент поиска в функциях текстового поиска должен при анализе разрешаться в строчное значение. Поскольку неизвестный тип маркера параметра не может разрешаться в строчное значение (SQLCODE -418), надо явно отобразить маркер параметра в тип данных **VARCHAR**.

Пример

- Следующий запрос служит для генерирования списка сотрудников в порядке соответствия их резюме запросу "программист AND (java OR cobol)" и нормализованного значения релевантности от 0 до 100:

```
SELECT EMPNO,  
       INTEGER(SCORE(RESUME,  
                  'programmer AND (java OR cobol)') * 100) AS RELEVANCE  
FROM EMP_RESUME  
WHERE RESUME_FORMAT = 'ascii'  
      AND CONTAINS(RESUME, 'программист AND (java OR cobol)') = 1  
ORDER BY RELEVANCE DESC
```

Функция xmlcolumn-contains

Функция db2-fn:xmlcolumn-contains возвращает последовательность документов XML из столбца данных XML на основе текстового поиска, выполненного механизмом DB2 Text Search для заданных терминов поиска.

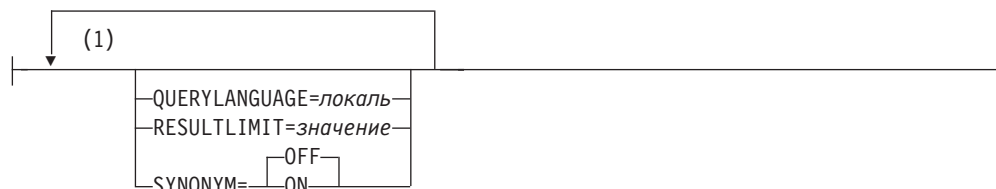
Синтаксис

►► db2-fn:xmlcolumn-contains (строковый-литерал, аргумент-поиска (1) , строковый-литерал-опций)

Примечания:

- 1 строковый-литерал-опций должен соответствовать правилам для опций-аргумента-поиска.

опции-аргументов-поиска:



Примечания:

- 1 Вы можете задать каждую опцию только однажды.

строковый-литерал

Задаёт имя столбца типа данных XML для поиска с помощью функции db2-fn:xmlcolumn-contains. Значение строкового литерала *строковый-литерал* регистрозависимо; регистр должен совпадать с регистром имени таблицы и столбца. Вы должны специфицировать имя столбца с помощью имени таблицы или имени производной таблицы. Имя схемы SQL не обязательно. Если вы не задаёте имя схемы SQL, используется значение CURRENT SCHEMA.

У столбца должен быть индекс текстового поиска.

аргумент-поиска

Выражение, возвращающее атомарное строковое значение или пустую последовательность. Строка не может полностью состоять из пробелов и не может быть пустой строкой. Строка должна быть преобразуема к типу данных VARCHAR максимальной длины 4096 байт в соответствии с правилами XMLCAST.

строковый-литерал-опций

Задаёт опции аргументов поиска, действительные для функции.

В составе *опций-аргумента-поиска* можно задать следующие опции:

QUERYLANGUAGE=локаль

Задаёт локаль, которую механизм DB2 Text Search использует при выполнении текстового поиска в текстовом столбце DB2. Значение - любая поддерживаемая локаль. Если не задать **QUERYLANGUAGE**, значением по умолчанию будет локаль индекса текстового поиска. Если в индексе текстового поиска значение параметра **LANGUAGE** - AUTO, для **QUERYLANGUAGE** используется значение en_US.

RESULTLIMIT=значение

Если оптимизатор выбирает план, при котором, чтобы получить оценку, механизм поиска вызывается для каждой строки набора результатов, то опция **RESULTLIMIT** на производительность не влияет. Если же механизм поиска вызывается один раз для всего набора результатов, то **RESULTLIMIT** действует как условие FETCH FIRST.

При использовании нескольких текстовых поисков, в которых **RESULTLIMIT** задается в одном и том же запросе, использовать один и тот же *аргумент-поиска*. Если вы используете разные значения *аргумента-поиска*, можно не получить ожидаемых результатов. Пример того, что может произойти при использовании нескольких текстовых поисков, а также возможное решение смотрите в последнем примере в теме “Примеры” на стр. 100.

SYNONYM = OFF | ON

Задаёт, следует ли использовать словарь синонимов, связанный с индексом текстового поиска. Значение по умолчанию - OFF. Чтобы использовать синонимы, добавьте словарь синонимов в индекс текстового поиска при помощи Инструментов синонимов.

OFF Не использовать словарь синонимов

ON Использовать словарь синонимов, связанный с индексом текстового поиска.

Возвращаемые значения

Возвращаемое значение - это конкатенация непустых значений XML из столбца, который задан строковым литералом *строковой-литерал*. Непустые значения XML возвращаются в недетерминированном порядке. Такие значения XML - это документы XML, где функция SQL CONTAINS, использующая *аргумент-поиска* для заданного строковым литералом *строковый-литерал* столбца, может вернуть 1. Если таких значений XML нет, возвращается пустая последовательность.

Если *аргумент-поиска* - это пустая последовательность, возвращается также пустая последовательность. Если *аргумент-поиска* - это пустая строка или строка, полностью состоящая из пробелов, возвращается ошибка. Если третий аргумент пуст, результат будет тем же, как и в случае, когда третий аргумент не задан.

Если у столбца, для определения которого вы использовали *строковый-литерал*, нет индекса текстового поиска, возвращается ошибка.

Функция db2-fn:xmlcolumn-contains связана с функцией db2-fn:sqlquery, и обе функции могут дать один и тот же результат. Однако аргументы двух функций отличаются по

зависимости от регистра символов. Первый аргумент, *строковой-литерал*, в функции `db2-fn:xmlcolumn-contains` обрабатывается XQuery; он регистрозависим. Поскольку имена столбцов и таблиц в базе данных DB2 по умолчанию состоят из заглавных букв, обычно первый аргумент функции `db2-fn:xmlcolumn-contains` также состоит из заглавных букв. Первый аргумент функции `db2-fn:sqlquery` обрабатывается SQL, который автоматически преобразует идентификаторы в символы верхнего регистра.

Следующие вызовы функций эквивалентны и возвращают одинаковые результаты, в предположении, что таблица `PRODUCT` принадлежит схеме, в настоящее время назначенной `CURRENT SCHEMA`:

```
db2-fn:xmlcolumn-contains("PRODUCT.DESCRPTION", "snow shovel")
```

```
db2-fn:sqlquery("select description from product  
where contains(description, 'snow shovel') = 1")
```

Примеры

В следующих примерах для выполнения поиска использован механизм DB2 Text Search. Столбцы, по которым проводится поиск - это столбцы XML, у которых есть индекс текстового поиска.

Первая функция ищет документы XML, хранимые в столбце `PRODUCT.DESCRPTION` и содержащие слова `snow` и `shovel` (снеговая лопата). Эта функция задает для максимального количества возвращаемых документов значение 2. Если текстовый поиск возвращает больше документов, можно оптимизировать поиск при помощи опции **RESULTLIMIT**, ограничивающей максимальное число возвращаемых документов.

```
db2-fn:xmlcolumn-contains('PRODUCT.DESCRPTION', 'snow shovel', 'RESULTLIMIT=2')
```

Функция возвращает документы XML, соответствующие критериям поиска. Документы могут содержать не только описание одного продукта. Например, следующий фрагмент XML состоит из описаний двух продуктов из столбца XML. Каждый документ содержит описание продукта и такую информацию, как имя продукта, цена, вес и ID продукта.

```
<product xmlns="http://posample.org" pid="100-100-01">  
  <description>  
    <name>Snow Shovel, Basic 22 inch</name>  
    <details>Basic Snow Shovel, 22 inches wide, straight handle with  
      D-Grip</details>  
    <price>9.99</price>  
    <weight>1 kg</weight>  
  </description>  
</product>  
<product xmlns="http://posample.org" pid="100-101-01">  
  <description>  
    <name>Snow Shovel, Deluxe 24 inch</name>  
    <details>A Deluxe Snow Shovel, 24 inches wide, ergonomic curved handle  
      with D-Grip</details>  
    <price>19.99</price>  
    <weight>2 kg</weight>  
  </description>  
</product>
```

Следующая функция ищет в столбце `XML STUDENT_ESSAYS.ABSTRACTS` десять студенческих рефератов, содержащих фразу `горючие полезные ископаемые на испанском`, то есть `combustible fossil`. В качестве языка текстового поиска функция задает `es_ES` (испанский в Испании) и использует словарь синонимов, который был создан для связанного индекса текстового поиска. Эта функция оптимизирует поиск, используя параметр **RESULTLIMIT**, который ограничивает количество результатов.

```
db2-fn:xmlcolumn-contains('STUDENT_ESSAYS.ABSTRACTS', 'combustible fossil',
'QUERYLANGUAGE=es_ES RESULTLIMIT=10 SYNONYM=ON')
```

В следующем примере функция db2-fn:xmlcolumn-contains используется для поиска документов XML, хранящихся в столбце PRODUCT.DESCRPTION и содержащих слово ergonomic. Выражение возвращает имя продукта, цена которого меньше 20.

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

Предыдущее выражение возвращает только элементы имен из возвращаемых документов XML. Например, если термин ergonomic содержится в описании продукта Snow Shovel, Deluxe 24 inch, это выражение возвращает элемент имени аналогично следующему:

```
<name xmlns="http://posample.org" >Snow Shovel, Deluxe 24 inch</name>
```

В следующем выражении функция db2-fn:xmlcolumn-contains используется для поиска документов XML из столбца PRODUCT.DESCRPTION, содержащих слова ice и scraper. Это выражение использует ID продуктов из описаний продуктов, чтобы найти заказы на покупку в таблице PURCHASEORDER, содержащей ID продуктов. Это выражение возвращает ID покупателей из заказов на покупку, содержащих ID продуктов из соответствующих документов описаний XML.

```
xquery
declare default element namespace "http://posample.org";
for $po in db2-fn:sqlquery('
select XMLElement(Name "po", XMLElement(Name "custid", purchaseorder.custid),
XMLElement(Name "porder", purchaseorder.porder))
from purchaseorder')
let $product := db2-fn:xmlcolumn-contains('PRODUCT.DESCRPTION',
'ice scraper')/product
where $product/@pid = $po/porder/PurchaseOrder/item/partid
order by $po/custid
return $po/custid
```

Это выражение возвращает элементы custid, содержащие ID покупателей. Эти элементы расположены в возрастающем порядке. Например, если запросу соответствуют три заказа на покупку, а ID покупателей равны 1001, 1002 и 1003, выражение возвращает следующие элементы:

```
<custid xmlns="http://posample.org">1001</custid>
<custid xmlns="http://posample.org">1002</custid>
<custid xmlns="http://posample.org">1003</custid>
```

Если в одном запросе выполняется несколько текстовых поисков, механизм DB2 Text Search объединяет несколько результатов текстового поиска и возвращает их. Например, следующий оператор SELECT ищет резюме сотрудников, содержащие точные фразы ruby on rails и ajax web. Условие WHERE содержит два текстовых поиска. Каждый текстовый поиск возвращает не более десяти результатов, и каждый текстовый поиск использует различные аргументы поиска для искомых резюме сотрудников. Оператор может вернуть меньше десяти ID, даже если существует более десяти резюме, содержащих обе фразы.

```
SELECT EMPNO FROM EMP_RESUME
WHERE XMLEXISTS('db2-fn:xmlcolumn-contains(''EMP_RESUME.XML_FORMAT'',
''ruby on rails'', ''RESULTLIMIT=10'')')
AND XMLEXISTS('db2-fn:xmlcolumn-contains(''EMP_RESUME.XML_FORMAT'',
''ajax web'', ''RESULTLIMIT=10'')')
```

Для предыдущего оператора DB2 Text Search возвращает не более десяти строк для каждого текстового поиска. Однако если резюме в возвращаемых строках содержат только одну фразу, а не обе, никакие ID работников не возвращаются.

Один из способов изменить оператор SELECT - скомбинировать эти два текстовых поиска в условии WHERE в единый текстовый поиск. Следующий оператор использует единый текстовый поиск и возвращает ID работников, в чьих резюме содержатся обе фразы, ruby on rails и ajax web:

```
SELECT EMPNO FROM EMP_RESUME
WHERE XMLEXISTS('db2-fn:xmlcolumn-contains(''EMP_RESUME.XML_FORMAT'',
      '"ruby on rails" AND "ajax web"', 'RESULTLIMIT=10')')
```

Глава 8. Команды управления для DB2 Text Search

Есть несколько команд, позволяющих управлять DB2 Text Search на уровнях экземпляра, базы данных, таблицы и текстового индекса. Все эти команды можно запускать через db2ts.

Команды управления уровня экземпляра служат для запуска и остановки служб экземпляра DB2 Text Search и для очистки более не нужных индексов текстового поиска:

db2ts START FOR TEXT

Запускает службы экземпляра DB2 Text Search

db2ts STOP FOR TEXT

Останавливает службы экземпляра DB2 Text Search

db2ts CLEANUP FOR TEXT

Очищает все ненужные собрания текстового поиска

Команды управления уровня базы данных задают или отключают базы данных для DB2 Text Search и очищают блокировки команд:

db2ts ENABLE DATABASE FOR TEXT

Включает для текущей базы данных создание, использование индексов текстового поиска и управление ими.

db2ts DISABLE DATABASE FOR TEXT

Отключает DB2 Text Search для базы данных и отбрасывает несколько таблиц и производных таблиц каталога текстового поиска.

db2ts CLEAR COMMAND LOCKS

Удаляет все блокировки команд для определенного индекса текстового поиска или для всех индексов в базе данных.

Команды уровня таблицы и индекса служат для создания и администрирования индексов текстового поиска по столбцам таблицы:

db2ts CREATE INDEX

Создает индекс текстового поиска

db2ts DROP INDEX

Отбрасывает индекс текстового поиска, связанный с текстовым столбцом

db2ts ALTER INDEX

Изменяет характеристики индекса текстового поиска

db2ts UPDATE INDEX

Составляет или обновляет индекс текстового поиска на основе текущего содержимого текстового столбца

db2ts CLEAR EVENTS FOR TEXT

Удаляет события из производной таблицы событий SYSIBMTS.TSEVENT, содержащей информацию о состоянии и ошибках индексирования

db2ts HELP

Выводит список опций команд **db2ts** и информацию о конкретных сообщениях об ошибках

db2ts START FOR TEXT,

Эта команда:

- Запускает службы экземпляра DB2 Text Search, поддерживающие другие команды управления для DB2 Text Search и возможность указывать индексы текстового поиска в запросах SQL.
- Включает службы (демоны для UNIX) на компьютере хоста, запуская сервер баз данных DB2. Эти службы ответственны за планирование обновлений индексов текстового поиска на сервере баз данных DB2 и за работу механизма текстового поиска. Службы экземпляра запускаются под именем авторизации владельца экземпляра DB2. Если службы экземпляра уже запущены, эта команда не оказывает никакого действия.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS для текущего экземпляра DB2.

Обязательное соединение

Эту команду надо запускать с сервера баз данных DB2.

Синтаксис команды

▶▶—START FOR TEXT—▶▶

Параметры команды

Нет

db2ts STOP FOR TEXT

Эта команда останавливает службы экземпляра текстового поиска DB2.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS для текущего экземпляра DB2.

Обязательное соединение

Эту команду надо запускать с сервера баз данных DB2.

Синтаксис команды

▶▶—STOP FOR TEXT—▶▶

Параметры команды

Нет

Замечания по использованию

Заметим, что нельзя одновременно вызывать для индекса текстового поиска несколько команд, если они могут конфликтовать. Если эта команда введена во время работы конфликтующей команды, произойдет ошибка и команда завершится неудачно; после этого можно попытаться запустить команду еще раз. Для исключения прерывания выполнения одновременно запущенных команд перед вводом команды остановки убедитесь, что не активны никакие другие команды управления, например, обновления индекса. Некоторые из конфликтующих команд:

- **DROP INDEX**
- **UPDATE INDEX**
- **CLEAR EVENTS FOR INDEX**
- **ALTER INDEX**
- **DISABLE DATABASE FOR TEXT**
- **STOP FOR TEXT**

db2ts CLEANUP FOR TEXT,

Эта команда очищает все устаревшие собрания текстового поиска DB2 в экземпляре. Собрание может оказаться устаревшим, если:

- База данных с индексами текстового поиска отброшена раньше, чем для этой базы данных отключен текстовый поиск DB2.
- Таблица отброшена раньше, чем отброшены связанные с ней индексы текстового поиска.

Примечание: Собрание текстового поиска - это базовое представление индекса текстового поиска. Между собранием текстового поиска и индексом текстового поиска существует однозначная взаимосвязь. С индексами текстового поиска работают команды, тогда как с собраниями текстового поиска работают инструменты. Запросите таблицу каталога SYSIBMTS.TSCOLLECTIONNAMES, чтобы определить собрание текстового поиска для индекса текстового поиска. Дополнительную информацию смотрите в теме “Инструмент управления для текстового поиска DB2”.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Для успешного запуска команды пользователь должен быть владельцем экземпляра DB2 с полномочиями DBADM и DATAACCESS.

Обязательное соединение

Эту команду надо запускать с сервера баз данных DB2.

Синтаксис команды

►►—CLEANUP FOR TEXT—◄◄

AUTOGRANT

Когда задана эта опция, делается попытка предоставить владельцу экземпляра привилегии DBADM с DATAACCESS, если у владельца экземпляра не оказалось таких привилегий для этой базы данных. Для успешного предоставления переменных у пользователя должна быть привилегия SECADM для этой базы данных, и он не может быть владельцем экземпляра (владелец не может предоставлять привилегии сам себе).

Замечания по использованию

После успешного выполнения эта команда:

- Включает возможность текстового поиска DB2 для базы данных.
- Устанавливает значения по умолчанию конфигурации базы данных DB2 Text Search в производной таблице SYSIBMTS.TSDEFAULTS.
- Создает следующие управляющие производные таблицы текстового поиска DB2 в схеме SYSIBMTS:
 - SYSIBMTS.TSDEFAULTS
 - SYSIBMTS.TSLOCKS
 - SYSIBMTS.TSINDEXES
 - SYSIBMTS.TSCONFIGURATION
 - SYSIBMTS.TSCOLLECTIONNAMES

Эти таблицы создаются в табличном пространстве базы данных по умолчанию (IBMDEFAULTGROUP).

Изменения в файловой системе на сервере DB2: Нет.

db2ts DISABLE DATABASE FOR TEXT

Эта команда в некотором смысле обращает изменения, внесенные командой **ENABLE DATABASE FOR TEXT** (например, отбрасывает относящиеся к текстовому поиску таблицы и производную таблицу).

Эта команда при запуске:

- Отключает возможность текстового поиска DB2 для базы данных.
- Удаляет следующие таблицы и производные таблицы:
 - SYSIBMTS.TSDEFAULTS
 - SYSIBMTS.TSLOCKS
 - SYSIBMTS.TSINDEXES
 - SYSIBMTS.TSCONFIGURATION
 - SYSIBMTS.TSCOLLECTIONNAMES

Эти таблицы удаляются из табличного пространства базы данных по умолчанию (IBMDEFAULTGROUP).

- Если задана опция **FORCE**, удаляется вся информация текстового индекса из базы данных и все связанные собрания. Кроме того, обновляются текстовые службы, чтобы удалить всю оставшуюся информацию изменения расписаний. Справку смотрите в разделе “Команда db2ts DROP INDEX”.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Привилегии, предоставленные ID авторизации данной процедуры, должны включать полномочия DBADM.

Предварительное требование

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. SYSADM более не имеет привилегий SECADM и DBADM в Версии 9.7. Привилегия SECADM должна быть явно предоставлена DBADM с полномочиями DATAACCESS владельцу экземпляра перед запуском команды **DISABLE DATABASE**.

Обязательное соединение

База данных

Синтаксис команды

►►—DISABLE DATABASE FOR TEXT—┐
└FORCE┘ | Возможности соединения | ───►

Возможности соединения:

┌──CONNECT TO—имя_базы_данных──┐
└USER— имя_пользователя—USING— пароль┘

Параметры команды

FORCE

Задает, что все индексы текстового поиска будут отброшены из базы данных принудительно.

Если эта опция не задана, а для этой базы данных определены индексы текстового поиска, команда завершится неудачно.

Если эта опция задана и служба DB2 Text Search еще не запущена (не введена команда db2ts **START FOR TEXT**), индексы (собрания) текстового поиска не отбрасываются.

CONNECT TO имя_базы_данных

Это условие задает базу данных, с которой надо установить соединение. База данных должна находиться в локальной системе. Если условие задано, оно имеет приоритет перед переменной среды DB2DBDFT. Это условие можно опустить, если выполнены следующие условия:

- Для переменной среды DB2DBDFT задано допустимое имя базы данных.
- Пользователь, запускающий эту команду, обладает необходимой авторизацией для соединения с сервером баз данных.

USER имя-пользователя USING пароль

Это условие задает имя авторизации и пароль, используемые для установки соединения.

Замечания по использованию

Эта команда не влияет на состояние включения DB2 Net Search Extender для базы данных. Она удаляет таблицы и производные таблицы каталога DB2 Text Search, созданные командой **ENABLE FOR TEXT**.

Перед отбрасыванием базы данных DB2 с определениями индексов текстового поиска запустите эту команду и убедитесь, что все текстовые индексы и собрания удалены успешно.

Если какие-либо индексы не удастся удалить при помощи опции **FORCE**, имена таких собраний записываются в файлы журнала **db2diag**. Если команда текстового поиска **DISABLE DATABASE FOR TEXT** не выполнена до команды процессора командной строки **DROP DATABASE**, надо выполнить также очистку службы индексов текстового поиска при помощи команды **CLEANUP FOR TEXT**. Дополнительную информацию об отбрасывании индексов смотрите в описании команды **DROP INDEX**, а информацию о собраниях текстового поиска и их связи с индексами текстового поиска - в описании команды **CLEANUP FOR TEXT**.

Примечание: Не рекомендуется выполнять действия, которые приводят к возникновению бесхозных собраний, то есть собраний, остающихся определенными на сервере текстового поиска, но не используемых DB2. Вот некоторые ситуации, которые приводят к появлению бесхозных собраний:

- Когда команда процессора командной строки **DROP DATABASE** или оператор **DROP TABLE** выполняются без предварительного запуска команды **DISABLE DATABASE FOR TEXT**.
- Когда команда **DISABLE DATABASE FOR TEXT** выполняется с использованием опции **FORCE**.
- Некоторые другие ситуации ошибок. В некоторых сценариях можно использовать команду **CLEANUP FOR TEXT**.

Заметим, что нельзя одновременно вызывать для индекса текстового поиска несколько команд, если они могут конфликтовать. Если эта команда введена во время работы конфликтующей команды, произойдет ошибка и команда завершится неудачно; после этого можно попытаться запустить команду еще раз. Некоторые из конфликтующих команд:

- **DROP INDEX**
- **UPDATE INDEX**
- **CLEAR EVENTS FOR INDEX**
- **ALTER INDEX**
- **DISABLE DATABASE FOR TEXT**
- **STOP FOR TEXT**

db2ts CLEAR COMMAND LOCKS

Удаляет все блокировки команд для определенного индекса текстового поиска или для всех индексов текстового поиска в базе данных. Блокировка команды создается в начале выполнения команды текстового поиска и уничтожается при завершении этой команды. Она предотвращает конфликты между различными командами.

Для всех блокировок, связанных с более не выполняемыми процессами, очистка выполняется автоматически. Это делается, чтобы индекс текстового поиска был доступен для нового требования поиска.

Использование этой команды требуется в том редком случае, когда блокировка остается неудаленной в результате неожиданного поведения системы, и ее требуется очистить явно.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Если имя индекса не задано, у пользователя *имя_пользователя* соединения с базой данных должны быть полномочия DBADM. Для очистки блокировки команд конкретного индекса у пользователя *имя_пользователя* соединения с базой данных должна быть привилегия CONTROL для таблицы, на которой создан этот индекс текстового поиска.

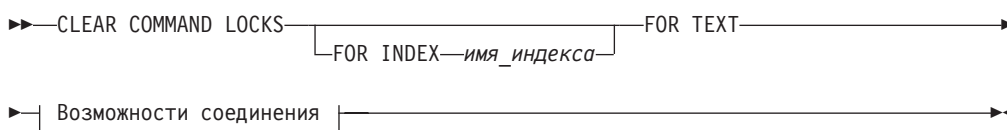
Предварительное требование

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. SYSADM более не имеет привилегий SECADM и DBADM в Версии 9.7. Привилегия SECADM должна быть явно предоставлена DBADM с полномочиями DATAACCESS владельцу экземпляра перед запуском команды **CLEAR COMMAND LOCKS**.

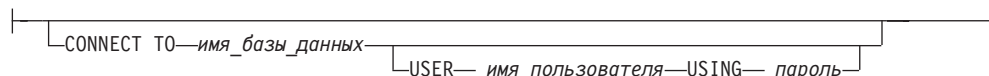
Обязательное соединение

База данных

Синтаксис команды



Возможности соединения:



Параметры команды

FOR INDEX *имя_индекса*

Имя индекса, как оно было задано в команде **CREATE INDEX**.

CONNECT TO *имя_базы_данных*

Это условие задает базу данных, с которой надо установить соединение. База данных должна находиться в локальной системе. Если условие задано, оно имеет приоритет перед переменной среды DB2DBDFT. Это условие можно опустить, если выполнены следующие условия:

- Для переменной среды DB2DBDFT задано допустимое имя базы данных.
- Пользователь, запускающий эту команду, обладает необходимой авторизацией для соединения с сервером баз данных.

USER *имя-пользователя* USING *пароль*

Это условие задает имя авторизации и пароль, используемые для установки соединения.

Замечания по использованию

Вы вызываете эту процедуру, потому что процесс, которому принадлежит блокировка команды, не работает. В таком случае эта команда (владелец блокировки) не может завершиться, и индекс не может стать доступным. Вам надо предпринять действия по исправлению ситуации. Предположим, процесс, выполняющий команду **DROP INDEX**, неожиданно завис. Он удалил некоторые данные индексов, но не всю информацию каталога и собраний. Блокировка команды остается нетронутой. После очистки блокировки команды **DROP INDEX** вы можете захотеть повторить команду **DROP INDEX**. В другом примере процесс, выполняющий команду **CREATE INDEX**, неожиданно завис. Он создал часть информации каталога индексов и собраний, но не всю. Блокировка команды остается нетронутой. После очистки блокировки команды **CREATE INDEX** вы можете выполнить команды **DROP INDEX** и **CREATE INDEX**.

Когда вводится эта команда, изменяется содержимое производной таблицы текстового поиска DB2 SYSIBMTS.TSLOCKS.

db2ts CREATE INDEX,

Создает индекс текстового поиска для столбца с текстовыми данными, что позволяет выполнять поиск данных в столбце при помощи функций текстового поиска.

Индекс не содержит никаких данных, пока команда текстового поиска **UPDATE INDEX** не будет в явном виде вызвана пользователем или в неявном виде - службой текстового поиска в соответствии с заданной частотой обновления этого индекса.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

Один из следующих серверов:

- Привилегия CONTROL для таблицы, на которой определен индекс
- Привилегия INDEX для таблицы, на которой определен индекс
и одно из следующих:
 - Полномочия IMPLICIT_SCHEMA для базы данных, если нет явного или неявного имени схемы индекса
 - Привилегия CREATEIN для схемы, если имя схемы индекса ссылается на существующую схему.
- Полномочия DBADM

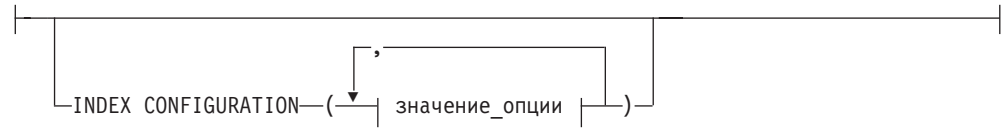
Предварительное требование

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. SYSADM более не имеет привилегий SECADM и DBADM в Версии 9.7. Привилегия SECADM должна быть явно предоставлена DBADM с полномочиями DATAACCESS владельцу экземпляра перед запуском команды **CREATE INDEX**.

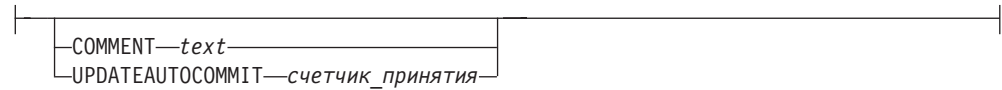
Обязательное соединение

База данных

опции конфигурации индекса:



значение_опции:



Возможности соединения:



Параметры команды

CREATE INDEX *имя_индекса*

Задаёт имя индекса (необязательно - с указанием схемы), который будет создан и однозначно идентифицирует индекс текстового поиска в базе данных. Это имя должно подчиняться ограничениям именования для индексов DB2.

ON *имя_таблицы*

Имя таблицы, содержащей текстовый столбец. Индексы текстового поиска нельзя создать для следующих таблиц:

- таблиц с разделением по диапазонам
- таблиц объединения
- материализованных таблиц запросов
- производные таблицы

имя_текстового_столбца

Имя столбца для индексирования. Столбец должен быть одного из следующих типов данных: CHAR, VARCHAR, LONG VARCHAR, CLOB, DBCLOB, BLOB, GRAPHIC, VARGRAPHIC, LONG VARGRAPHIC или XML. Если у столбца другой тип данных, воспользуйтесь функцией преобразования (указывается как *схема_функции.имя_функции*), чтобы преобразовать тип данных этого столбца в один из допустимых типов. Другой вариант - указать внешнюю пользовательскую функцию, обращающуюся к текстовым документам для индексации. Для одного столбца можно создать только один индекс текстового поиска,

имя_функции(имя_текстового_столбца)

Задаёт имя с указанием схемы, соответствующее соглашениям DB2 об именовании и указывающее внешнюю скалярную функцию, обращающуюся к текстовым документам в столбце, тип данных которого не поддерживается текстовым поиском. Функция выполняет преобразование типа данных значения и возвращает это значение преобразованным в один из типов данных, поддерживаемых текстовым поиском. Ее задача - выполнение преобразования типа столбца. Эта функция должна принимать ровно один параметр и возвращать ровно одно значение.

CODEPAGE *кодовая_страница*

Задаёт кодовую страницу DB2 (CODEPAGE), которую нужно использовать при индексировании текстовых документов. Значение по умолчанию задается значением в производной таблице SYSIBMTS.TSDEFAULTS в строке с DEFAULTNAME='CODEPAGE' (то есть используется кодовая страница по умолчанию для базы данных). Этот аргумент применяется только к двоичным типам данных, то есть тип столбца или тип возврата из функции преобразования должен быть типом BLOB или символьным типом FOR BIT DATA.

LANGUAGE *локаль*

Задаёт язык, используемый DB2 для зависящей от языка обработки документа при индексировании. Если вы не указали локаль, для задания значения по умолчанию для параметра **LANGUAGE** будет использована территория базы данных. Если вы хотите выполнять автоматический просмотр документов для определения локали, задайте для переменной *locale* значение AUTO.

FORMAT *формат*

Задаёт формат текстовых документов в столбце. Поддерживаемые форматы включают в себя TEXT, XML, HTML и INSO. Эта информация нужна DB2 Text Search при индексировании документов. Если формат не указан, используется значение по умолчанию. Значение по умолчанию находится в производной таблице SYSIBMTS.TSDEFAULTS, в строке с DEFAULTNAME='FORMAT'. Для столбцов типа данных XML используется формат по умолчанию 'XML' независимо от значения DEFAULTNAME. Для использования формата INSO должна быть установлена поддержка расширенного текста.

UPDATE FREQUENCY

Задаёт частоту выполнения обновлений индекса. Индекс будет обновлен, если число изменений не меньше значения, заданного для **UPDATE MINIMUM**. Частота обновления **NONE** указывает, что индекс не будет обновляться. Это может пригодиться для текстового столбца в таблице, данные в которой никогда не изменяются. Это полезно также, если пользователь хочет обновлять индекс вручную (с помощью команды **UPDATE INDEX**). Автоматические обновления могут выполняться, только если выполнена команда **START FOR TEXT** и запущены службы экземпляра DB2 Text Search.

Значение частоты по умолчанию берётся из производной таблицы SYSIBMTS.TSDEFAULTS, из строки с DEFAULTNAME='UPDATEFREQUENCY'.

NONE

Индекс не будет обновляться. Обновление должно запускаться вручную.

D День недели для обновления индекса.

* Каждый день недели.

целое1

Определённые дни недели, от воскресенья (0) до субботы (6)

H Час указанного дня для обновления индекса.

* Каждый час дня.

целое2

Определённые часы дня, от полуночи до 11 вечера: от 0 до 23

M Минута указанного часа для обновления индекса.*целое3*

Указывается как начало часа (0) или как кратное 5 число минут с начала часа: 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 или 55

UPDATE MINIMUM *минимальное_число_изменений*

Задаёт минимальное число изменений в текстовых документах перед инкрементным обновлением индекса в момент, заданный в **UPDATE FREQUENCY**. Допускаются только положительные целые значения. Значение по умолчанию берётся из производной таблицы SYSIBMTS.TSDEFAULTS, из строки с DEFAULTNAME='UPDATEMINIMUM'.

Примечание: Это значение игнорируется командой **UPDATE INDEX** (если в ней не указана опция **USING UPDATE MINIMUM**). Меньшее значение улучшает соответствие между столбцом таблицы и индексом текстового поиска. Однако оно также повышает использование ресурсов.

COLLECTION DIRECTORY *каталог*

Каталог, в котором хранится индекс текстового поиска. По умолчанию данные собрания будут находиться в DBPATH/db2collections, где значение DBPATH - путь файла, использовавшийся при создании базы данных. Следует указывать абсолютный путь. Максимальная длина абсолютного пути - 215 символов. Разные индексы распределяются по подкаталогам с именем *идентификатор индекса* внутри COLLECTION DIRECTORY, где идентификатор индекса - идентификатор, генерируемый системой.

ADMINISTRATION TABLES IN *имя_табличного_пространства*

Задаёт имя существующего обычного табличного пространства для управляющих таблиц, создаваемых для индекса. Если не указано, используется табличное пространство базовой таблицы, для которой создается индекс.

INDEX CONFIGURATION (*значение_опции*)

Задаёт дополнительные значения, связанные с индексом, в виде пар строк опция/значение. Поддерживаются следующие значения:

Таблица 6. Спецификации для опции - значения

Опция	Значение	Тип данных	Описание
COMMENT	<i>текст</i>	Строка длиной до 512 байтов	Добавляет строковый комментарий в столбец REMARKS в производной таблице каталога DB2 Text Search TSINDEXES. Добавляет также значение строкового комментария в качестве описания собрания.

Таблица 6. Спецификации для опции - значения (продолжение)

Опция	Значение	Тип данных	Описание
UPDATEAUTOCOMMIT	<i>счетчик_принятия</i>	Целое	<p>Начиная с DB2 Версии 9.7 Fix Pack 3 и в более новых выпусках Fix Pack, задает число изменений индекса, после которого автоматически выполняется принятие для сохранения сделанной работы, для первого обновления индекса или для инкрементных изменений.</p> <ul style="list-style-type: none"> Первоначальное обновление индекса обрабатывает пакеты документов из базовой таблицы после того, как будет активирован триггер для захвата изменений данных. Когда число измененных документов достигает COMMITCOUNT, сервер выполняет промежуточное принятие. Записи журнала, сгенерированные необработанными документами, будут удалены из таблицы кэша. Использование опции UPDATEAUTOCOMMIT для первоначального обновления текстового индекса ведет к значительному увеличению времени выполнения. Для инкрементных обновлений обработанные записи журнала удаляются из таблицы кэша с каждым промежуточным принятием. COMMITCOUNT подсчитывает число измененных документов, а не число записей таблицы кэша.

Напоминание: Нечисловые значения должны указываться в одинарных кавычках. Знак одинарной кавычки внутри строкового значения указывается как две одинарных кавычки подряд.

Пример:

```
INDEX CONFIGURATION (COMMENT 'Index on User''s Guide column')
```

CONNECT TO *имя_базы_данных*

Это условие задает базу данных, с которой надо установить соединение. База данных должна находиться в локальной системе. Если условие задано, оно имеет приоритет перед переменной среды DB2DBDFT. Это условие можно опустить, если выполнены следующие условия:

- Для переменной среды DB2DBDFT задано допустимое имя базы данных.
- Пользователь, запускающий эту команду, обладает необходимой авторизацией для соединения с сервером баз данных.

USER *имя-пользователя* **USING** *пароль*

Это условие задает имя авторизации и пароль, используемые для установки соединения.

Замечания по использованию

Все ограничения и соглашения об именовании, применимые к объектам баз данных и запросам DB2, применяются также к возможностям и запросам текстового поиска DB2. Идентификаторы, относящиеся к текстовому поиску DB2, должны отвечать соглашениям именованию DB2. Есть также некоторые дополнительные ограничения. Например, идентификаторы должны иметь вид:

```
[A-Za-z][A-Za-z0-9@#$_]*
```

или

```
"[A-Za-z ] [A-Za-z0-9@#$_ ]*"
```

В случае успешного завершения команды **CREATE INDEX**:

- Обновляются данные сервера DB2 Text Search. Создается собрание с именем *экземпляр_имя_базы_данных_идентификатор_индекса_номер*, как в следующем примере:

```
tigertail_MYTSDB_TS250517_0000
```

Имя собрания можно получить из столбца COLLECTIONNAME производной таблицы SYSIBMTS.TSCOLLECTIONNAMES.

- Обновляется информация в каталоге DB2 Text Search. В указанном табличном пространстве создается таблица кэша индексов с соответствующими индексами DB2. Кроме того, в указанном табличном пространстве создается таблица событий индексирования.
- Если DB2 Text Search используется одновременно с DB2 Net Search Extender и уже существует активный индекс Net Search Extender для данного столбца таблицы, новый текстовый индекс деактивируется.
- Свежесозданный индекс текстового поиска не заполняется автоматически. Для заполнения индекса текстового поиска нужно запустить команду **UPDATE INDEX** вручную или автоматически (в результате определения расписания обновлений для этого индекса с помощью опции **UPDATE FREQUENCY**).
- Обновляется файл данных индекса текстового поиска на сервере баз данных DB2. Для каждого индекса в экземпляра записывается информация о запланированных обновлениях.

Ограничения использования:

- Для таблицы должен быть задан первичный ключ. В DB2 Text Search можно без каких либо ограничений по типу данных использовать многостолбцовые первичные ключи DB2. Число столбцов в первичном ключе должно быть не более числа столбцов в первичном ключе, разрешенного DB2, минус 2.
- Общая длина всех столбцов первичного ключа для таблицы с индексами DB2 Text Search должна быть не больше максимальной общей длины первичного ключа, разрешенной DB2, минус 15 байтов. Посмотрите описание ограничений оператора DB2 CREATE INDEX.

Заметим, что нельзя одновременно вызывать для индекса текстового поиска несколько команд, если они могут конфликтовать. Если эта команда введена во время работы конфликтующей команды, произойдет ошибка и команда завершится неудачно; после этого можно попытаться запустить команду еще раз. Некоторые из конфликтующих команд:

- **DROP INDEX**
- **UPDATE INDEX**
- **CLEAR EVENTS FOR INDEX**
- **ALTER INDEX**
- **DISABLE DATABASE FOR TEXT**
- **STOP FOR TEXT**

db2ts DROP INDEX,

Отбрасывает существующий индекс текстового поиска.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен индекс
- Полномочия DBADM

Предварительное требование

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. SYSADM более не имеет привилегий SECADM и DBADM в Версии 9.7. Привилегия SECADM должна быть явно предоставлена DBADM с полномочиями DATAACCESS владельцу экземпляра перед запуском команды **DROP INDEX**.

Обязательное соединение

База данных

Синтаксис команды

►►—DROP INDEX—*имя_индекса*—FOR TEXT— | Возможности соединения | —————►►

Возможности соединения:

|—————|
|—CONNECT TO—*имя_базы_данных*— |
|—USER— *имя_пользователя*—USING— *пароль*— |

Параметры команды

DROP INDEX *имя_индекса*

Схема и имя индекса, как они были заданы в команде **CREATE INDEX**. Они уникальным образом задают индекс текстового поиска в базе данных.

CONNECT TO *имя_базы_данных*

Это условие задает базу данных, с которой надо установить соединение. База данных должна находиться в локальной системе. Если условие задано, оно имеет приоритет перед переменной среды DB2DBDFT. Это условие можно опустить, если выполнены следующие условия:

- Для переменной среды DB2DBDFT задано допустимое имя базы данных.
- Пользователь, запускающий эту команду, обладает необходимой авторизацией для соединения с сервером баз данных.

USER *имя-пользователя* USING *пароль*

Это условие задает имя авторизации и пароль, используемые для установки соединения.

Замечания по использованию

Заметим, что нельзя одновременно вызывать для индекса текстового поиска несколько команд, если они могут конфликтовать. Если эта команда введена во время работы конфликтующей команды, произойдет ошибка и команда завершится неудачно; после этого можно попытаться запустить команду еще раз. Некоторые из конфликтующих команд:

- **DROP INDEX**

- UPDATE INDEX
- CLEAR EVENTS FOR INDEX
- ALTER INDEX
- DISABLE DATABASE FOR TEXT
- STOP FOR TEXT

Отбрасывание пользовательской таблицы в DB2 не приводит к отбрасыванию индексов текстового поиска. Их надо отбросить вручную перед отбрасыванием таблицы (это предпочтительно) или после него. После отбрасывания индекса текстового поиска текстовый поиск по соответствующему столбцу становится невозможным. Если вы планируете создать новый индекс для того же текстового столбца, перед созданием нового индекса текстового поиска надо отсоединиться от базы данных, а затем соединиться с ней снова.

Изменения в базе данных:

- Обновляется информация в каталоге DB2 Text Search.
- Отбрасываются таблицы кэша индекса и таблицы событий.
- Удаляются триггеры для пользовательской текстовой таблицы.
- Собрание, связанное с этим определением индекса текстового поиска DB2 уничтожается.

db2ts ALTER INDEX,

Эта команда изменяет характеристики обновления индекса

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен текстовый индекс
- Полномочия DBADM

Предварительное требование

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. SYSADM более не имеет привилегий SECADM и DBADM в Версии 9.7. Привилегия SECADM должна быть явно предоставлена DBADM с полномочиями DATAACCESS владельцу экземпляра перед запуском команды **ALTER INDEX**.

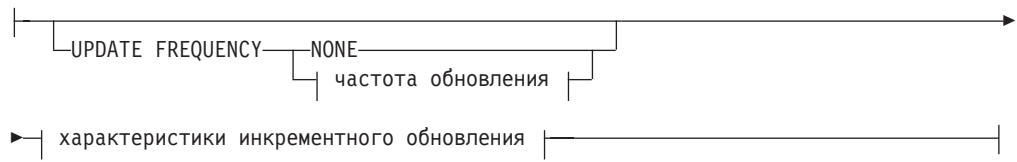
Обязательное соединение

База данных

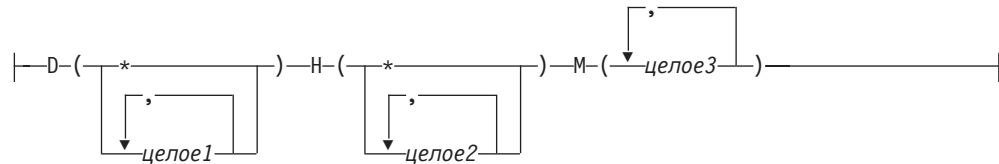
Синтаксис команды

►► ALTER INDEX—*имя_индекса*—FOR TEXT—| изменить характеристики |—————►
 ►| варианты |—| Возможности соединения |—————►►

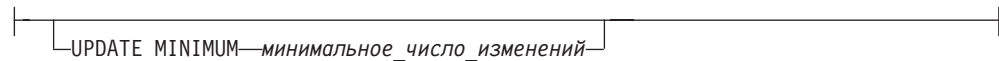
Изменить характеристики:



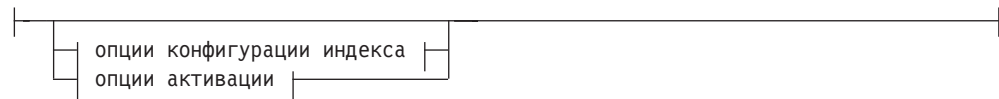
частота обновления:



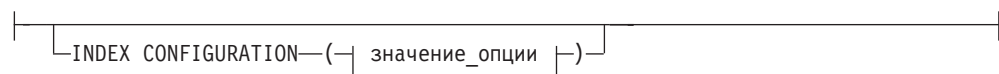
характеристики инкрементного обновления:



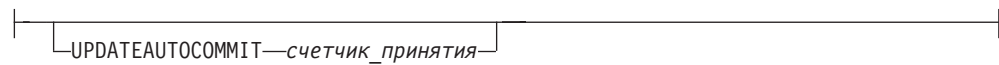
варианты:



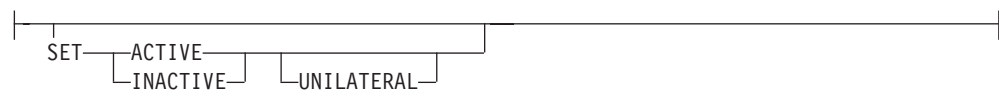
опции конфигурации индекса:



значение_опции:



опции активации:



Возможности соединения:



Параметры команды

ALTER INDEX *имя_индекса*

Схема и имя индекса, как они были заданы в команде **CREATE INDEX**. Они уникальным образом задают индекс текстового поиска в базе данных.

UPDATE FREQUENCY

Задаёт частоту выполнения обновлений индекса. Индекс будет обновлен, если число изменений не меньше значения, заданного для **UPDATE MINIMUM**. Частота обновления **NONE** указывает, что индекс не будет обновляться. Это может пригодиться для текстового столбца в таблице, данные в которой никогда не изменяются. Это полезно также, если пользователь хочет обновлять индекс вручную (с помощью команды **UPDATE INDEX**). Автоматические обновления могут выполняться, только если выполнена команда **START FOR TEXT** и запущены службы экземпляра DB2 Text Search.

Значение частоты по умолчанию берётся из производной таблицы SYSIBMTS.TSDEFAULTS, из строки с DEFAULTNAME='UPDATEFREQUENCY'.

NONE

К этому текстовому индексу не будет применяться автоматическое обновление. Все последующие обновления индекса должны запускаться вручную.

D День недели для обновления индекса.

* Каждый день недели.

целое1

Определённые дни недели, от воскресенья (0) до субботы (6)

H Час указанного дня для обновления индекса.

* Каждый час дня.

целое2

Определённые часы дня, от полуночи до 11 вечера: от 0 до 23

M Минута указанного часа для обновления индекса.

целое3

Указывается как начало часа (0) или как кратное 5 число минут с начала часа: 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 или 55

Если вы не задали опцию **UPDATE FREQUENCY**, параметры частоты обновления не изменяются.

UPDATE MINIMUM *минимальное_число_изменений*

Задаёт минимальное число изменений в текстовых документах перед инкрементным обновлением индекса. Несколько изменений одного и того же текстового документа рассматриваются как отдельные изменения. Если вы не задали опцию **UPDATE MINIMUM**, этот параметр не изменяется.

INDEX CONFIGURATION (*значение_опции*)

Начиная с версии 9.7 с пакетом Fix Pack 3 и более новыми пакетами Fix Pack, этот необязательный входной аргумент типа VARCHAR(32K) позволяет изменять параметры конфигурации текстового индекса. Поддерживаются следующие опции:

Таблица 7. Спецификации для опции - значения

Опция	Значение	Тип данных	Описание
UPDATEAUTOCOMMIT	<i>счетчик_принятия</i>	Целое	<p>Задаёт число изменений индекса, после которого автоматически выполняется принятие для сохранения сделанной работы, для первого обновления индекса или для инкрементных обновлений.</p> <ul style="list-style-type: none"> Первоначальное обновление индекса обрабатывает пакеты документов из базовой таблицы после того, как будет активирован триггер для захвата изменений данных. Когда число измененных документов достигает COMMITCOUNT, сервер выполняет промежуточное принятие. Записи журнала, сгенерированные необработанными документами, будут удалены из таблицы кэша. Использование опции UPDATEAUTOCOMMIT для первоначального обновления текстового индекса ведет к значительному увеличению времени выполнения. Для инкрементных обновлений обработанные записи журнала удаляются из таблицы кэша с каждым промежуточным принятием. COMMITCOUNT подсчитывает число измененных документов, а не число записей таблицы кэша.

ОПЦИИ АКТИВАЦИИ

Начиная с версии 9.7 с пакетом Fix Pack 3 и более новыми пакетами Fix Pack, этот входной аргумент типа integer задает состояние текстового индекса.

ACTIVE

Задаёт активное состояние текстового индекса

INACTIVE

Задаёт неактивное состояние текстового индекса

UNILATERAL

Задаёт одностороннее изменение, которое действует только на состояние индексов текстового поиска DB2. Если указан этот аргумент, меняется на активное или неактивное состояние только индекса текстового поиска DB2. Без аргумента UNILATERAL состояния активации индексов DB2 Text Search и DB2 Net Search Extender меняются синхронно, так что активным будет только один из этих текстовых индексов.

Примечание: После изменения состояния индекса текстового поиска с активного на неактивное или наоборот введите команду **FLUSH PACKAGE CACHE DYNAMIC**. Она удаляет все кэшированные динамические операторы SQL из кэша пакетов и указывает DB2 неявным образом скомпилировать тот же оператор SQL при следующем требовании.

CONNECT TO имя_базы_данных

Это условие задает базу данных, с которой надо установить соединение. База данных должна находиться в локальной системе. Если условие задано, оно имеет приоритет перед переменной среды DB2DBDFT. Это условие можно опустить, если выполнены следующие условия:

- Для переменной среды DB2DBDFT задано допустимое имя базы данных.
- Пользователь, запускающий эту команду, обладает необходимой авторизацией для соединения с сервером баз данных.

USER *имя-пользователя* **USING** *пароль*

Это условие задает имя пользователя и пароль, используемые для установки соединения.

Замечания по использованию

Все ограничения и соглашения об именовании, применимые к объектам баз данных и запросам DB2, применяются также к возможностям и запросам текстового поиска DB2. Идентификаторы, относящиеся к текстовому поиску DB2, должны отвечать соглашениям именованя DB2. Есть также некоторые дополнительные ограничения. Например, идентификаторы должны иметь вид:

```
[A-Za-z][A-Za-z0-9@#$_]*
```

или

```
"[A-Za-z ][A-Za-z0-9@#$_ ]*"
```

Заметим, что нельзя одновременно вызывать для индекса текстового поиска несколько команд, если они могут конфликтовать. Если эта команда введена во время работы конфликтующей команды, произойдет ошибка и команда завершится неудачно; после этого можно попытаться запустить команду еще раз. Некоторые из конфликтующих команд:

- **ALTER INDEX**
- **CLEAR EVENTS FOR INDEX**
- **DROP INDEX**
- **UPDATE INDEX**
- **DISABLE DATABASE FOR TEXT**
- **STOP FOR TEXT**

Изменения в базе данных: Обновляется информация в каталоге DB2 Text Search.

db2ts UPDATE INDEX,

Эта команда обновляет индекс текстового поиска (собрание в текстовом поиске DB2 Text Search), чтобы отразить текущее содержимое текстового столбца, с которым связан индекс. Во время выполнения изменения возможен поиск. До завершения изменения поиск работает с частично обновленным индексом.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен текстовый индекс
- Полномочия DATAACCESS

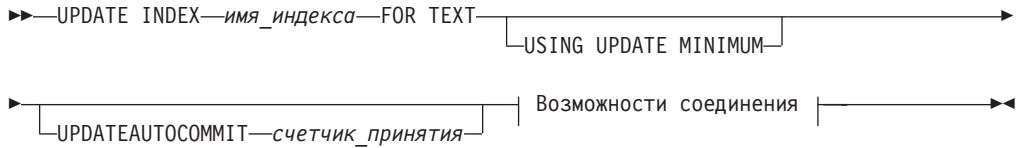
Предварительное требование

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. SYSADM более не имеет привилегий SECADM и DBADM в Версии 9.7. Привилегия SECADM должна быть явно предоставлена DBADM с полномочиями DATAACCESS владельцу экземпляра перед запуском команды **UPDATE INDEX**.

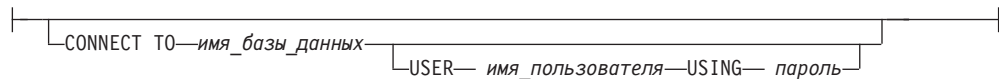
Обязательное соединение

База данных

Синтаксис команды



Возможности соединения:



Параметры команды

UPDATE INDEX *имя_индекса*

Задаёт имя индекса текстового поиска для обновления. Это имя должно подчиняться ограничениям именованности для индексов DB2.

USING UPDATE MINIMUM

Указывает значение *минимального_числа_изменений* **UPDATE MINIMUM**, задаваемое в команде **CREATE INDEX**, которое надо использовать, и запускает инкрементное обновление, только если внесено указанное минимальное число изменений. По умолчанию обновление запускается безусловно.

INDEX CONFIGURATION (*значение_опции*)

Начиная с версии 9.7 с пакетом Fix Pack 3 и более новыми пакетами Fix Pack, этот необязательный входной аргумент типа VARCHAR(32K) позволяет изменять параметры конфигурации текстового индекса. Поддерживаются следующие опции:

Таблица 8. Спецификации для опции - значения

Опция	Значение	Тип данных	Описание
UPDATEAUTOCOMMIT	<i>счетчик_принятия</i>	Целое	<p>Задает число изменений индекса, после которого автоматически выполняется принятие для сохранения сделанной работы, для первого обновления индекса или для инкрементных обновлений.</p> <ul style="list-style-type: none"> Первоначальное обновление индекса обрабатывает пакеты документов из базовой таблицы после того, как будет активирован триггер для захвата изменений данных. Когда число измененных документов достигает COMMITCOUNT, сервер выполняет промежуточное принятие. Записи журнала, сгенерированные необработанными документами, будут удалены из таблицы кэша. Использование опции UPDATEAUTOCOMMIT для первоначального обновления текстового индекса ведет к значительному увеличению времени выполнения. Для инкрементных обновлений обработанные записи журнала удаляются из таблицы кэша с каждым промежуточным принятием. COMMITCOUNT подсчитывает число измененных документов, а не число записей таблицы кэша.

CONNECT TO *имя_базы_данных*

Это условие задает базу данных, с которой надо установить соединение. База данных должна находиться в локальной системе. Если условие задано, оно имеет приоритет перед переменной среды DB2DBDFT. Это условие можно опустить, если выполнены следующие условия:

- Для переменной среды DB2DBDFT задано допустимое имя базы данных.
- Пользователь, запускающий эту команду, обладает необходимой авторизацией для соединения с сервером баз данных.

USER имя-пользователя **USING** пароль

Это условие задает имя авторизации и пароль, используемые для установки соединения.

Замечания по использованию

Все ограничения и соглашения об именовании, применимые к объектам баз данных и запросам DB2, применяются также к возможностям и запросам текстового поиска DB2. Идентификаторы, относящиеся к текстовому поиску DB2, должны отвечать соглашениям именованию DB2. Есть также некоторые дополнительные ограничения. Например, идентификаторы должны иметь вид:

```
[A-Za-z][A-Za-z0-9@#$_]*
```

или

```
"[A-Za-z ][A-Za-z0-9@#$_ ]*"
```

Возврат из этой команды не происходит, пока не будет полностью завершена обработка обновления индекса. Продолжительность зависит от числа индексируемых документов и от числа уже проиндексированных документов. Имя собрания для индекса можно получить из столбца COLLECTIONNAME производной таблицы SYSIBMTS.TSCOLLECTIONNAMES.

Заметим, что нельзя одновременно вызывать для индекса текстового поиска несколько команд, если они могут конфликтовать. Если эта команда введена во время работы конфликтующей команды, произойдет ошибка и команда завершится неудачно; после этого можно попытаться запустить команду еще раз. Некоторые из конфликтующих команд:

- **UPDATE INDEX**
- **CLEAR EVENTS FOR INDEX**
- **ALTER INDEX**
- **DROP INDEX**
- **DISABLE DATABASE FOR TEXT**
- **STOP FOR TEXT**

Примечание: Если в отдельных документах есть ошибки, эти документы надо исправить. Первичные ключи документов с ошибками можно найти в таблице событий для индекса. Следующая команда **UPDATE INDEX** переиндексирует эти документы, если соответствующие строки в пользовательской таблице изменены.

Изменения в базе данных:

- В таблицу событий вставляются строки (с информацией об ошибках анализа текстового поиска DB2).
- В случае инкрементного обновления удаляется информация из таблицы кэша индекса.
- Перед первым обновлением создаются триггеры для пользовательской текстовой таблицы.
- Обновляется собрание.
- Новые или измененные документы анализируются и индексируются.
- Удаленные документы удаляются из индекса.

db2ts CLEAR EVENTS FOR INDEX,

Эта команда удаляет события индексации из таблицы событий индекса, используемой для управления. Имя этой таблицы можно найти в столбце EVENTVIEWNAME производной таблицы SYSIBMTS.TSINDEXES.

Каждая операция обновления индекса, обработавшая хотя бы один документ, заносит информационные записи и, в некоторых случаях, записи об ошибках в таблицу событий. Для автоматических обновлений эту таблицу следует регулярно проверять. Ошибки конкретных документов надо исправлять (корректируя содержимое документов). После исправления этих ошибок события могут быть (и должны быть ради экономии места) очищены.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен индекс
- Полномочия DBADM

Предварительное требование

У владельца экземпляра должна быть авторизация DBADM с полномочиями DATAACCESS. SYSADM более не имеет привилегий SECADM и DBADM в Версии 9.7. Привилегия SECADM должна быть явно предоставлена DBADM с полномочиями DATAACCESS владельцу экземпляра перед запуском команды **CLEAR EVENTS**.

Обязательное соединение

База данных

Синтаксис команды

►►—CLEAR EVENTS FOR INDEX—*имя_индекса*—FOR TEXT—| Возможности соединения |—►►

Возможности соединения:

|—CONNECT TO—*имя_базы_данных*—|—USER— *имя_пользователя*—USING— *пароль*—|

Параметры команды

имя_индекса

Имя индекса, как оно было задано в команде **CREATE INDEX**. Это имя должно подчиняться ограничениям именованности для индексов DB2.

CONNECT TO *имя_базы_данных*

Это условие задает базу данных, с которой надо установить соединение. База данных должна находиться в локальной системе. Если условие задано, оно имеет приоритет перед переменной среды DB2DBDFT. Это условие можно опустить, если выполнены следующие условия:

- Для переменной среды DB2DBDFT задано допустимое имя базы данных.

- Пользователь, запускающий эту команду, обладает необходимой авторизацией для соединения с сервером баз данных.

USER *имя-пользователя* **USING** *пароль*

Это условие задает имя авторизации и пароль, используемые для установки соединения.

Замечания по использованию

Все ограничения и соглашения об именовании, применимые к объектам баз данных и запросам DB2, применяются также к возможностям и запросам текстового поиска DB2. Идентификаторы, относящиеся к текстовому поиску DB2, должны отвечать соглашениям именованию DB2. Есть также некоторые дополнительные ограничения. Например, идентификаторы должны иметь вид:

```
[A-Za-z][A-Za-z0-9@#$_]*
```

или

```
"[A-Za-z ][A-Za-z0-9@#$_ ]*"
```

Если планируются регулярные обновления (смотрите опции **UPDATE FREQUENCY** в командах **CREATE INDEX** или **ALTER INDEX**), таблицу событий надо проверять регулярно. Чтобы очистить таблицу событий текстового поиска DB2 для индекса текстового поиска, после проверки причины события и устранения источника ошибки используйте команду **CLEAR EVENTS FOR INDEX**.

Убедитесь, что во все строки, упомянутые в таблице событий, внесены изменения. Изменяя строки в пользовательской таблице, вы гарантируете, что следующая попытка **UPDATE INDEX** успешно переиндексирует ранее содержавшие ошибки документы.

Заметим, что нельзя одновременно вызывать для индекса текстового поиска несколько команд, если они могут конфликтовать. Если эта команда введена во время работы конфликтующей команды, произойдет ошибка и команда завершится неудачно; после этого можно попытаться запустить команду еще раз. Некоторые из конфликтующих команд:

- **CLEAR EVENTS FOR INDEX**
- **UPDATE INDEX**
- **ALTER INDEX**
- **DROP INDEX**
- **DISABLE DATABASE FOR TEXT**
- **STOP FOR TEXT**

Изменения в базе данных: Очищается таблица событий.

db2ts HELP

Команда **HELP** выводит список доступных команд текстового поиска DB2 или синтаксис отдельной команды.

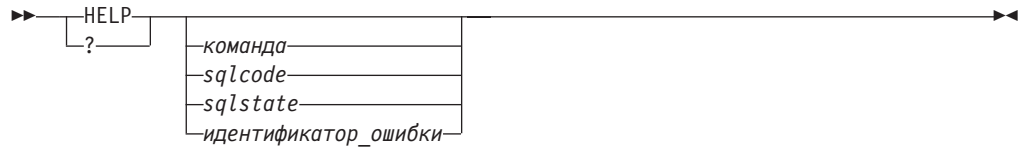
Команда **db2ts HELP** служит также для получения справки по конкретным сообщениям об ошибках.

Для выполнения необходимо перед командой ввести **db2ts** в командной строке.

Авторизация

Нет.

Синтаксис команды



Параметры команды

HELP | ?

Выводит информацию справки для команды или кода ошибки.

команда

Первые ключевые слова указывают команду текстового поиска DB2 Text Search:

- ENABLE
- DISABLE
- CREATE
- DROP
- ALTER
- UPDATE
- CLEAR (и для CLEAR COMMAND LOCKS, и для CLEAR EVENTS FOR INDEX)
- CLEANUP
- START
- STOP

sqlcode

SQLCODE для сообщения, возвращаемого командой db2ts (в хранимой процедуре управления или вне ее) или запросом текстового поиска.

sqlstate

SQLSTATE, возвращаемое командой, хранимой процедурой управления или запросом текстового поиска.

идентификатор_ошибки

Идентификатор составляет часть *сообщения_об_ошибке_текстового_поиска*, встроенную в сообщения об ошибках. Этот идентификатор начинается с 'CIE' и имеет вид CIE#####, где ##### - число. Этот идентификатор представляет конкретную ошибку, возвращаемую при текстовом поиске. Он может также быть возвращен в информационном сообщении при завершении команды текстового поиска или в сообщении по завершении процедуры управления текстовым поиском. Если идентификатор не начинается с 'CIE', **db2ts help** не сможет вывести информацию о таком *идентификаторе_ошибки*. Например, db2ts не может дать справку о сообщении с *идентификатором_ошибки* IQQR0012E.

Замечания по использованию

При использовании оболочки UNIX может оказаться необходимым задать аргументы для **db2ts** в двойных кавычках, как в следующем примере:

```
db2ts "? CIE00323"
```

Без этих кавычек оболочка попытается заменить символ подстановки содержимым рабочего каталога, что может привести к непредсказуемым результатам.

Если в качестве первого ключевого слова задана какая-либо из команд **db2ts**, выводится синтаксис указанной команды. Если две команды **db2ts** начинаются с одного и того же ключевого слова (**CLEAR COMMAND LOCKS** и **CLEAR EVENTS FOR INDEX**), в результате ввода **db2ts help clear** выводится синтаксис обеих команд; можно вывести справку по каждой команде конкретно, добавив второе ключевое слово, чтобы различить их, например, **db2ts help clear events**. Если после **?** или **HELP** не задан параметр, **db2ts** выводит список всех доступных команд **db2ts**.

При задании *sqlcode*, *sqlstate* или *CIEидентификатор_ошибки* выводится информация о **SQLCODE**, **SQLSTATE** или идентификаторе ошибки. Например,

```
db2ts help SQL20423
```

или

```
db2ts ? 38H10
```

или

```
db2ts ? CIE00323
```

Примечание: Для следующих команд текстового поиска указанных ниже опции на данный момент недопустимы:

```
db2ts DISABLE DATABASE FOR TEXT  
[SERVER]
```

```
db2ts ENABLE DATABASE FOR TEXT  
[SERVER]
```

```
db2ts UPDATE INDEX  
[REORGANIZE] и [PARSE ONLY]
```

Глава 9. Подпрограммы DB2 Text Search

DB2 Text Search содержит несколько управляющих подпрограмм SQL для выполнения команд и для возврата сообщения о результатах выполняемых команд и кодов возврата для них.

При помощи управляющих подпрограмм SQL можно запускать следующие команды **db2ts**:

- **ENABLE DATABASE**
- **DISABLE DATABASE**
- **CREATE INDEX**
- **DROP INDEX**
- **ALTER INDEX**
- **UPDATE INDEX**
- **CLEAR EVENTS**
- **CLEAR COMMAND LOCKS**

Хранимая процедура SYSTS_ADMIN_CMD - Запуск команд управления текстовым поиском

Процедура SYSTS_ADMIN_CMD используется программами для запуска команд управления текстовым при помощи оператора SQL CALL.

Синтаксис

►►—SYSTS_ADMIN_CMD—(—строка_команды—,—локаль_сообщений—,—сообщение—)————►►

Схема - SYSPROC.

Параметр процедуры

строка-команды

Входной аргумент типа VARCHAR(32K), задающий одну команду управления текстовым поиском для выполнения. Синтаксис команды совпадает с синтаксисом в DB2 Text Search, за исключением опций соединения. В этой процедуре опции соединения не поддерживаются. Команды, запускаемые через эту процедуру, используют текущее соединение.

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий требуемую локаль для всех возвращаемых текстовых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Выходной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершенной операции.

Авторизация

Привилегия EXECUTE для процедуры SYSTS_ADMIN_CMD.

Эта процедура на настоящий момент поддерживает следующие команды текстового поиска DB2:

- ALTER INDEX
- CLEAR COMMAND LOCKS
- CLEAR EVENTS
- CREATE INDEX
- DISABLE DATABASE
- DROP INDEX
- ENABLE DATABASE
- UPDATE INDEX

Пример

Обновление индекса текстового поиска MYTEXTINDEX в схеме DB2TS с выводом сообщений на русском языке.

```
CALL SYSPROC.SYSTS_ADMIN_CMD  
('UPDATE INDEX DB2TS.MYTEXTINDEX FOR TEXT', 'ru_RU', ?);
```

Ниже приведен пример вывода этого запроса.

Значение выходных параметров

Имя параметра : MESSAGE

Значение параметра: CIE00001 Операция завершена успешно.

Состояние возврата = 0

Замечания по использованию

- Если выполнение этой команды завершается неудачно, возвращаются SQLCODE -20427 и SQLSTATE 38H14 с текстовым сообщением о конкретной ошибке. Например, если индекс MYTEXTINDEX уже существует, и введен следующий оператор:

```
CALL SYSPROC.SYSTS_ADMIN_CMD ('CREATE INDEX MYTEXTINDEX FOR TEXT  
ON DB2TS.TEXTBOOKS (STORY)', 'ru_RU', ?)
```

создание индекса завершится неудачно со следующим сообщением об ошибке:

```
SQL20427N Ошибка при выполнении процедуры или команды  
управления текстовым поиском. Сообщение об ошибке "CIE00201 Индекс текстового поиска  
"DB2TS"."MYTEXTINDEX" уже существует. ". SQLSTATE=38H14
```

- Если процедура возвращает SQLCODE, сообщение может быть усечено. Полную информацию сообщения можно найти в файлах журналов **db2diag**.

Процедура SYSTS_ENABLE - Включение текущей базы данных для текстового поиска

Эта процедура включает текстовый поиск DB2 для текущей базы данных.

Эта процедура должна быть успешно выполнена прежде, чем можно будет создавать индексы текстового поиска по столбцам таблиц в базе данных.

Эта процедура передает на сервер базы данных команду управления текстовым поиском **ENABLE DATABASE FOR TEXT**.

Синтаксис

►—SYSTS_ENABLE—(—локаль_сообщений—,—сообщение—)————►

Схема - SYSPROC.

Параметры процедуры

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий локаль, используемую для всех возвращаемых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Выходной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершенной операции.

Авторизация

Для выполнения команды ENABLE DATABASE у пользователя должна быть привилегия DBADM.

Примеры

Пример 1: Включение базы данных для текстового поиска с выводом сообщений на русском языке.

```
CALL SYSPROC.SYSTS_ENABLE('ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
Значение выходных параметров
-----
Имя параметра      : MESSAGE
Значение параметра: Операция выполнена успешно.
Состояние возврата = 0
```

Пример 2: В следующем примере SYSTS_ENABLE вызывается для базы данных, для которой уже включен текстовый поиск. В результате вызывающему возвращается сообщение об ошибке.

```
CALL SYSPROC.SYSTS_ENABLE('ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
SQL20427N Ошибка при выполнении процедуры или команды
управления текстовым поиском. Сообщение об ошибке продукта текстового поиска
"C1E00322 В указанной базе данных или базе данных по умолчанию уже разрешен
текстовый поиск. ". SQLSTATE 38H14
```

Замечания по использованию

- Процедуры управления текстовыми индексами используют уже существующее соединение с базой данных. В зависимости от завершения этих процедур для текущей транзакции выполняется принятие или откат. Поэтому рекомендуется перед выполнением этой процедуры управления текстовым поиском выполнить принятие всех изменений, внесенных транзакциями, чтобы избежать нежелательного влияния принятия или отката этой процедуры. Один из способов достигнуть этого - включить AUTOCOMMIT.

Примечание: Если для транзакции не выполнено принятие до вызова операции управления текстовым поиском, может возникнуть тупиковая ситуация, так как могут быть затронуты одни и те же объекты базы данных и операция будет ожидать завершения предыдущего шага. Это может случиться, например, если при выключенной опции AUTOCOMMIT создать таблицу, а затем создать текстовый индекс, не вызвав принятие в явном виде между этими двумя транзакциями.

- При выполнении этой процедуры
 - в схеме SYSIBMTS создаются объекты базы данных, такие как управляющие таблицы и производные таблицы каталога текстового поиска. Эти объекты помещаются в табличное пространство по умолчанию базы данных (IBMDEFAULTGROUP).
 - задаваемые значения по умолчанию индексов текстового поиска для базы данных можно посмотреть в производной таблице SYSIBMTS.TSDEFAULTS.
 - при успешном выполнении этой команды создаются и становятся доступными таблицы и производные таблицы каталога текстового поиска.

Процедура SYSTS_DISABLE - Выключение текущей базы данных для текстового поиска

Эта процедура отключает текстовый поиск DB2 для текущей базы данных.

Когда возможность текстового поиска отключена, индексы и команды текстового поиска для этой базы данных становятся недоступными.

Эта процедура передает на сервер базы данных команду управления текстовым поиском **DISABLE DATABASE FOR TEXT**.

Синтаксис

►►—SYSTS_DISABLE—(—опции—, —локаль_сообщений—, —сообщение—)—————►►

Схема - SYSPROC.

Параметры процедуры

опции

Входной аргумент типа VARCHAR(128), задающий используемые опции отключения базы данных. Можно задать аргумент FORCE. Если задано это значение, все индексы отбрасываются и возможность текстового поиска отключается принудительно. Никакие индексы текстового поиска не сохраняются, и никаких сообщений об ошибках и предупреждений не возвращается. Если аргумент является пустым значением или пустой строкой, выполняется попытка отключения возможности текстового поиска для базы данных.

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий локаль, используемую для всех возвращаемых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Выходной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершенной операции.

Авторизация

Привилегии, предоставленные ID авторизации данной процедуры, должны включать полномочия DBADM.

Примеры

Пример 1: В следующем примере текстовый поиск отключается для базы данных при помощи процедуры SYSTS_DISABLE. Задана опция FORCE, чтобы отключить возможность, даже если для таблиц базы данных все еще существуют индексы текстового поиска. Сообщения об ошибках должны возвращаться по-русски. Для выходного параметра *сообщение* задана строка информационного сообщения.

```
CALL SYSPROC.SYSTS_DISABLE('FORCE', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
Значение выходных параметров
-----
Имя параметра      : MESSAGE
Значение параметра: Операция выполнена успешно.
Состояние возврата = 0
```

Пример 2: В следующем примере текстовый поиск отключается для базы данных с существующими индексами текстового поиска при помощи процедуры SYSTS_DISABLE без задания опции FORCE. В результате вызывающему возвращается сообщение об ошибке. Желательно отбросить все существующие индексы текстового поиска перед отключением возможности тестового поиска, или же можно задать опцию FORCE как значение входного параметра *опции*.

```
CALL SYSPROC.SYSTS_DISABLE(' ', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
SQL20427N Ошибка при выполнении процедуры или команды
управления текстовым поиском. Сообщение об ошибке "CIE00326 В указанной базе данных или
базе данных по умолчанию активен индекс текстового поиска.". SQLSTATE 38N14
```

Замечания по использованию

- Процедуры управления текстовыми индексами используют уже существующее соединение с базой данных. В зависимости от завершения этих процедур для текущей транзакции выполняется принятие или откат. Поэтому вам может быть полезно выполнять принятие всех изменений транзакции во избежание неожиданных последствий такого принятия или отката. Один из способов достигнуть этого - включить AUTOCOMMIT.

Примечание: Если для транзакции не выполнено принятие до вызова операции управления текстовым поиском, может возникнуть тупиковая ситуация, так как могут быть затронуты одни и те же объекты базы данных и операция будет ожидать завершения предыдущего шага. Это может случиться, например, если при выключенной опции AUTOCOMMIT создать таблицу, а затем создать текстовый индекс, не вызвав принятие в явном виде между этими двумя транзакциями.

- Нельзя одновременно вызывать для индекса текстового поиска несколько процедур или команд, если они могут конфликтовать. Некоторые из конфликтующих процедур и команд:
 - Процедура SYSTS_ALTER или команда db2ts ALTER INDEX
 - Процедура SYSTS_CLEAR_EVENTS или команда db2ts CLEAR EVENTS FOR INDEX

- Процедура SYSTS_DISABLE или команда db2ts DISABLE DATABASE FOR TEXT
- Команда db2ts STOP FOR TEXT
- Процедура SYSTS_UPDATE или команда db2ts UPDATE INDEX

В случае конфликта процедура возвращает SQLCODE -20426 и SQLSTATE 38H13.

- При выполнении этой процедуры
 - обновляется информация в каталоге DB2 Text Search. Журнал индекса и таблицы событий отбрасываются. Триггеры для пользовательской текстовой таблицы удаляются.
 - если задана опция **FORCE**, вся информация текстового индекса удаляется из базы данных, и все связанные собрания удаляются. Кроме того, обновляются текстовые службы, чтобы удалить всю оставшуюся информацию изменения расписаний. Смотрите справку в темах "Команда db2ts DROP INDEX" и "Процедура SYSTS_DROP".
- Эта процедура не влияет на состояние включения DB2 Net Search Extender для базы данных. Она удаляет таблицы и производные таблицы каталога текстового поиска DB2, созданные процедурой SYSTS_ENABLE или командой **ENABLE FOR TEXT**.
- Перед отбрасывание базы данных DB2 с определениями индексов текстового поиска запустите эту процедуру и убедитесь, что все текстовые индексы и собрания удалены успешно.
- Если какие-либо индексы не удается удалить при помощи опции **FORCE**, имена таких собраний записываются в файлы журнала **db2diag**. Если процедура индексов текстового поиска SYSTS_DISABLE или команда **DISABLE DATABASE FOR TEXT** не выполнена до команды процессора командной строки **DROP DATABASE**, надо выполнить очистку службы индексов текстового поиска при помощи команды **CLEANUP FOR TEXT**. Дополнительную информацию об отбрасывании индексов смотрите в описании процедуры SYSTS_DROP или команды **DROP INDEX**, а информацию о собраниях текстового поиска и их связи с индексами текстового поиска - в описании команды **CLEANUP FOR TEXT**.

Примечание: Не рекомендуется выполнять действия, которые приводят к возникновению бесхозных собраний, то есть собраний, остающихся определенными на сервере текстового поиска, но не используемых DB2. Вот некоторые ситуации, которые могут привести к появлению бесхозных собраний:

- Когда команда процессора командной строки **DROP DATABASE** или оператор **DROP TABLE** выполняется без предварительного запуска процедуры SYSTS_DISABLE или команды **DISABLE DATABASE FOR TEXT**.
- Когда процедура SYSTS_DISABLE или команда **DISABLE DATABASE FOR TEXT** выполняется с использованием опции **FORCE**.
- Некоторые другие ситуации ошибок. В некоторых сценариях можно использовать команду **CLEANUP FOR TEXT**.

Процедура SYSTS_CREATE - Создание индекса текстового поиска для столбца

Эта процедура создает индекс текстового поиска для столбца с текстовыми данными, что позволяет выполнять поиск данных в столбце при помощи функций текстового поиска.

После создания индекса текстового поиска запросы могут использовать функции текстового поиска для поиска данных в столбце. Индекс не содержит никаких данных, пока команда текстового поиска **UPDATE INDEX** или процедура SYSTS_UPDATE не

будут в явном виде вызваны пользователем или в неявном виде - службами текстового поиска уровня экземпляра в соответствии с заданной частотой обновления этого индекса.

Эта процедура вызывает на сервере баз данных команду администрирования текстового поиска **CREATE INDEX**.

Синтаксис

►► SYSTS_CREATE (—схема_индекса—, —имя_индекса—, | источник текста |, — варианты |, —локаль_сообщений—, —сообщение—)

источник текста:

|—имя_таблицы— (— имя текстового столбца |) ,

имя текстового столбца:

|—имя_столбца—
|—имя_функции (имя_столбца)—

опции:

информация о тексте по умолчанию
характеристики обновления
опции хранения
опции конфигурации индекса

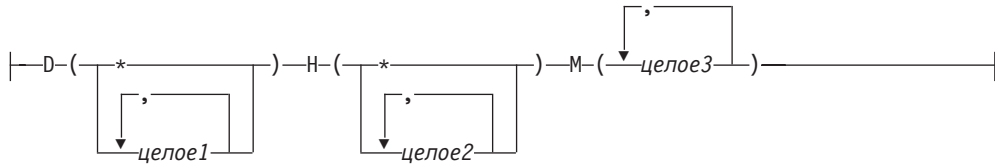
информация о тексте по умолчанию:

|—CODEPAGE—кодовая_страница— |—LANGUAGE—язык— |—FORMAT—формат—

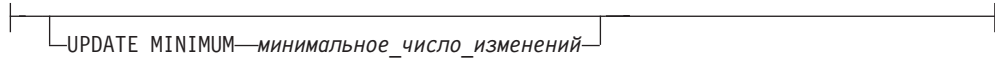
Изменить характеристики:

|—UPDATE FREQUENCY—NONE— частота обновления |
► |— характеристики инкрементного обновления |

частота обновления:



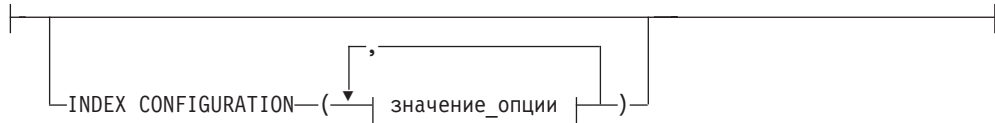
характеристики инкрементного обновления:



опции хранения:



опции конфигурации индекса:



значение_опции:

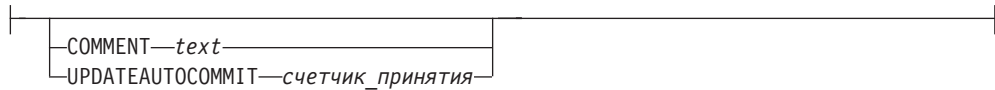


Схема - SYSPROC.

Параметры процедуры

схема_индекса

Входной аргумент типа VARCHAR(128), задающий схему индекса текстового поиска. Значение *схема_индекса* должно соответствовать ограничениям именованя для имен схем DB2. Если аргумент является пустым значением или пустой строкой, используется значение CURRENT SCHEMA Аргумент *схема_индекса* регистрозависим.

имя_индекса

Входной аргумент типа VARCHAR(128), задающий имя индекса. Вместе с аргументом *схема_индекса* уникальным образом задает индекс текстового поиска в базе данных. Аргумент *имя_индекса* регистрозависим.

источник текста

Входной аргумент типа VARCHAR(1024), задающий имя столбца для индексирования. Возможные значения:

имя-таблицы

Имя таблицы, содержащей текстовый столбец. Индексы текстового поиска нельзя создать для следующих таблиц:

- таблиц с разделением по диапазонам
- таблиц объединения
- материализованных таблиц запросов
- производных таблиц

Аргумент *имя_таблицы* регистрозависим.

имя текстового столбца

Имя столбца для индексирования.

имя-столбца

Столбец должен быть одного из следующих типов данных: CHAR, VARCHAR, LONG VARCHAR, CLOB, DBCLOB, BLOB, GRAPHIC, VARGRAPHIC, LONG VARGRAPHIC или XML. Если у столбца другой тип данных, воспользуйтесь функцией преобразования (указывается как *схема_функции.имя_функции*), чтобы преобразовать значения из этого столбца в один из подходящих типов. Смотрите синтаксис и подробную информацию в описании *имя_функции (имя_столбца)*. Другой вариант - указать внешнюю пользовательскую функцию, обращающуюся к текстовым документам для индексации. Для одного столбца можно создать только один индекс текстового поиска, Аргумент *имя_столбца* регистрозависим.

имя_функции (имя_столбца)

Задаёт имя с указанием схемы, соответствующее соглашениям DB2 об именовании, и указывающее внешнюю скалярную функцию, обращающуюся к текстовым документам в столбце, тип данных которого не поддерживается текстовым поиском. Функция выполняет преобразование типа данных значения и возвращает это значение преобразованным в один из типов данных, поддерживаемых текстовым поиском. Её задача - выполнение преобразования типа столбца. Эта функция должна принимать ровно один параметр и возвращать ровно одно значение. Аргумент *имя_функции (имя_столбца)* регистрозависим.

варианты

Входной аргумент типа VARCHAR(32K), задающий используемые опции. Если никаких опций не требуется, в качестве аргумента можно указать пустое значение или пустую строку. Доступные опции:

CODEPAGE *кодовая_страница*

Задаёт кодовую страницу DB2 (CODEPAGE), которую нужно использовать при индексировании текстовых документов. Значение по умолчанию задается значением в производной таблице SYSIBM.TSDEFAULTS в строке с DEFAULTNAME='CODEPAGE' (то есть используется кодовая страница по умолчанию для базы данных). Этот аргумент применяется к двоичным типам данных, то есть тип столбца или тип возврата из функции преобразования должен быть типом BLOB или символьным типом FOR BIT DATA.

LANGUAGE *язык*

Задаёт язык, используемый DB2 для зависящей от языка обработки документа при индексировании. Если вы не указали локаль, для задания значения по умолчанию для параметра LANGUAGE будет использована

территория базы данных. Если вы хотите выполнять автоматический просмотр документов для определения локали, задайте для локали значение AUTO.

FORMAT *формат*

Задает формат текстовых документов в столбце. Поддерживаемые форматы включают в себя TEXT, XML, HTML и INSO. Эта информация нужна DB2 Text Search при индексировании документов. Если формат не указан, используется значение по умолчанию. Значение по умолчанию находится в производной таблице SYSIBMTS.TSDEFAULTS, в строке с DEFAULTNAME='FORMAT'. Для столбцов типа данных XML используется формат по умолчанию 'XML' независимо от значения DEFAULTNAME. Для использования формата INSO должна быть установлена поддержка расширенного текста.

UPDATE FREQUENCY

Задает частоту выполнения обновлений индекса. Индекс будет обновлен, если число изменений не меньше значения, заданного для **UPDATE MINIMUM**. Частота обновления **NONE** указывает, что индекс не будет обновляться. Это может пригодиться для текстового столбца в таблице, данные в которой никогда не изменяются. Это полезно также, если пользователь хочет обновлять индекс вручную (с помощью команды **UPDATE INDEX**). Автоматические обновления могут выполняться, только если выполнена команда **START FOR TEXT** и запущены службы экземпляра DB2 Text Search.

Значение частоты по умолчанию берется из производной таблицы SYSIBMTS.TSDEFAULTS, из строки с DEFAULTNAME='UPDATEFREQUENCY'.

NONE

Индекс не будет обновляться. Обновление должно запускаться вручную.

D День недели для обновления индекса.

* Каждый день недели.

целое1

Определенные дни недели, от воскресенья (0) до субботы (6)

H Час указанного дня для обновления индекса.

* Каждый час дня.

целое2

Определенные часы дня, от полуночи до 11 вечера: от 0 до 23

M Минута указанного часа для обновления индекса.

целое3

Указывается как начало часа (0) или как кратное 5 число минут с начала часа: 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 или 55

UPDATE MINIMUM *минимальное_число_изменений*

Задает минимальное число изменений в текстовых документах перед инкрементным обновлением индекса в момент, заданный в **UPDATE FREQUENCY**. Допускаются только положительные целые значения. Значение по умолчанию берется из производной таблицы SYSIBMTS.TSDEFAULTS, из строки с DEFAULTNAME='UPDATEMINIMUM'.

Примечание: Это значение игнорируется командой **UPDATE INDEX** (если в ней не указана опция **USING UPDATE MINIMUM**). Меньшее значение улучшает

соответствие между столбцом таблицы и индексом текстового поиска. Однако оно также повышает использование ресурсов.

COLLECTION DIRECTORY *каталог*

Каталог, в котором хранится индекс текстового поиска. По умолчанию данные собрания будут находиться в DBPATH/db2collections, где значение DBPATH - путь файла, использовавшийся при создании базы данных. Следует указывать абсолютный путь. Максимальная длина абсолютного пути - 215 символов. Разные индексы распределяются по подкаталогам с именем *идентификатор индекса* внутри COLLECTION DIRECTORY, где идентификатор индекса - идентификатор, генерируемый системой.

ADMINISTRATION TABLES IN *имя_табличного_пространства*

Задаёт имя существующего обычного табличного пространства для управляющих таблиц, создаваемых для индекса. Если не указано, используется табличное пространство базовой таблицы, для которой создается индекс.

INDEX CONFIGURATION (*значение_опции*)

Задаёт дополнительные значения, связанные с индексом, в виде пар строк опция/значение. Поддерживаются следующие значения:

Таблица 9. Спецификации для опции - значения

Опция	Значение	Тип данных	Описание
COMMENT	<i>текст</i>	Строка длиной до 512 байтов	Добавляет строковый комментарий в столбец REMARKS в производной таблице каталога DB2 Text Search TSINDEXES. Добавляет также значение строкового комментария в качестве описания собрания.
UPDATEAUTO COMMIT	<i>commitcount _number</i>	Целое	Начиная с DB2 Версии 9.7 Fix Pack 3, задаёт число изменений индекса, после которого автоматически выполняется принятие для сохранения сделанной работы, для первого обновления индекса или для инкрементных обновлений. <ul style="list-style-type: none"> Первоначальное обновление индекса обрабатывает пакеты документов из базовой таблицы после того, как будет активирован триггер для захвата изменений данных. Когда число изменённых документов достигает COMMITCOUNT, сервер выполняет промежуточное принятие. Записи журнала, сгенерированные необработанными документами, будут удалены из таблицы кэша. Использование опции UPDATEAUTOCOMMIT для первоначального обновления текстового индекса ведёт к значительному увеличению времени выполнения. Для инкрементных обновлений обработанные записи журнала удаляются из таблицы кэша с каждым промежуточным принятием. COMMITCOUNT подсчитывает число изменённых документов, а не число записей таблицы кэша.

Напоминание: Нечисловые значения должны указываться в одинарных кавычках. Знак одинарной кавычки внутри строкового значения указывается как две одинарных кавычки подряд.

Пример:

```
INDEX CONFIGURATION (COMMENT 'Index on User''s Guide column')
```

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий локаль, используемую для всех возвращаемых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Выходной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершенной операции.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен индекс
- Привилегия INDEX для таблицы, на которой определен индекс
и одно из следующих полномочий:
 - Полномочия IMPLICIT_SCHEMA для базы данных, если нет явного или неявного имени схемы индекса
 - Привилегия CREATEIN для схемы, если имя схемы индекса ссылается на существующую схему.
- Полномочия DBADM

Примеры

Пример 1: В следующем примере выполняется создание индекса текстового поиска со значениями *схема_индекса* 'db2ts' и *имя_индекса* 'myTextIndex' с помощью процедуры SYSTS_CREATE. Опция 'UPDATE MINIMUM 10' указывает, что до выполнения инкрементного обновления индекса должно произойти по меньшей мере 10 изменений текстовых документов, связанных с этим индексом. Указано, что сообщения об ошибках должны возвращаться по-английски. В случае успешного завершения используемой команды текстового поиска в выходном параметре сообщение возвращается состояние исполнения команды.

```
CALL SYSPROC.SYSTS_CREATE('db2ts', 'myTextIndex',  
    'myUserSchema.myBaseTable (myTextColumn)', 'UPDATE MINIMUM 10',  
    'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
Значения выходных параметров
```

```
-----  
Имя параметра      : MESSAGE  
Значение параметра: Операция выполнена успешно.  
Состояние возврата = 0
```

Пример 2: В следующем примере SYSTS_CREATE вызывается для создания индекса текстового поиска со значениями *схема_индекса* 'db2ts' и *имя_индекса* 'myTextIndex'. Опции не заданы. В этом примере индекс уже существует, и поэтому возвращается сообщение об ошибке.

```
CALL SYSPROC.SYSTS_CREATE('db2ts', 'myTextIndex',  
    'myUserSchema.myBaseTable (myTextColumn)', '', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

SQL20427N Ошибка при выполнении процедуры или команды управления текстовым поиском. Сообщение об ошибке "CIE00201 Индекс текстового поиска "db2ts"."myTextIndex" уже существует. "

Замечания по использованию

- Процедуры управления текстовыми индексами используют уже существующее соединение с базой данных. В зависимости от завершения этих процедур для текущей транзакции выполняется принятие или откат. Поэтому вам может быть полезно выполнять принятие всех изменений транзакции во избежание неожиданных последствий такого принятия или отката. Один из способов достигнуть этого - включить AUTOCOMMIT для соединения.

Примечание: Если для транзакции не выполнено принятие до вызова операции управления текстовым поиском, может возникнуть тупиковая ситуация, так как могут быть затронуты одни и те же объекты базы данных и операция будет ожидать завершения предыдущего шага. Это может случиться, например, если при выключенной опции AUTOCOMMIT создать таблицу, а затем создать текстовый индекс, не вызвав принятие в явном виде между этими двумя транзакциями.

- Нельзя одновременно вызывать для индекса текстового поиска несколько процедур или команд, если они могут конфликтовать. Некоторые из конфликтующих процедур и команд:
 - Процедура SYSTS_ALTER или команда db2ts ALTER INDEX
 - Процедура SYSTS_CLEAR_EVENTS или команда db2ts CLEAR EVENTS FOR INDEX
 - Процедура SYSTS_DISABLE или команда db2ts DISABLE DATABASE FOR TEXT
 - Команда db2ts STOP FOR TEXT
 - Процедура SYSTS_UPDATE или команда db2ts UPDATE INDEX

В случае конфликта процедура возвращает SQLCODE -20426 и SQLSTATE 38H13.

- В случае успешного завершения команды **CREATE INDEX**:
 - Обновляются данные сервера DB2 Text Search. Создается собрание с именем *экземпляр_имя_базы_данных_идентификатор_индекса_номер*, как в следующем примере:
tigertail_MYTSDB_TS250517_0000

Имя собрания можно получить из производной таблицы SYSIBMTS.TSCOLLECTIONNAMES (столбец COLLECTIONNAME).

- Обновляется информация в каталоге DB2 Text Search. В указанном табличном пространстве создается таблица кэша индексов с соответствующими индексами DB2. Кроме того, в указанном табличном пространстве создается таблица событий индексирования.
- Если DB2 Text Search используется одновременно с DB2 Net Search Extender и уже существует активный индекс Net Search Extender для данного столбца таблицы, новый текстовый индекс деактивируется.
- Свежесозданный индекс текстового поиска не заполняется автоматически. Для заполнения индекса текстового поиска нужно запустить процедуру SYSTS_UPDATE или команду **UPDATE INDEX** вручную или автоматически (в результате определения расписания обновлений для этого индекса с помощью опции **UPDATE FREQUENCY**).
- Обновляется файл данных индекса текстового поиска на сервере баз данных DB2. Для каждого индекса в экземпляре записывается информация о запланированных обновлениях.

Ограничения использования:

- Для таблицы должен быть задан первичный ключ. В DB2 Text Search, можно без каких либо ограничений по типу данных использовать многостолбцовые первичные ключи DB2. Число столбцов в первичном ключе должно быть не более числа столбцов в первичном ключе, разрешенного DB2, минус 2.
- Общая длина всех столбцов первичного ключа для таблицы с индексами DB2 Text Search должна быть не больше максимальной общей длины первичного ключа, разрешенной DB2, минус 15 байтов. Смотрите ограничения DB2 для оператора DB2 CREATE INDEX.

Процедура SYSTS_DROP - Отбрасывание индекса текстового поиска

Эта процедура отбрасывает существующий индекс текстового поиска, связанный со столбцом таблицы.

После успешного выполнения этой процедуры к данному столбцу нельзя применять запросы текстового поиска.

Эта процедура вызывает на сервере баз данных команду администрирования текстового поиска **DROP INDEX**.

Синтаксис

```
► SYSTS_DROP (—схема_индекса—, —имя_индекса—, —————►  
► —локаль_сообщений—, —сообщение—) —————►
```

Схема - SYSPROC.

Параметры процедуры

схема_индекса

Входной аргумент типа VARCHAR(128), задающий схему индекса текстового поиска. Значение *схема_индекса* должно соответствовать ограничениям именования для имен схем DB2. Если аргумент является пустым значением или пустой строкой, используется значение CURRENT SCHEMA. Аргумент *схема_индекса* регистрозависим.

имя_индекса

Входной аргумент типа VARCHAR(128), задающий имя индекса. Вместе с аргументом *схема_индекса* уникальным образом задает индекс текстового поиска в базе данных. Аргумент *имя_индекса* регистрозависим.

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий локаль, используемую для всех возвращаемых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Входной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершенной операции.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен индекс
- Полномочия DBADM

Примеры

Пример 1: В следующем примере выполняется отбрасывание индекса текстового поиска, созданного со значениями *схема_индекса* 'db2ts' и *имя_индекса* 'myTextIndex'. Указано, что сообщения об ошибках должны возвращаться по-английски. При успешном выполнении процедуры вызывающему возвращается сообщение выходного параметра.

```
CALL SYSPROC.SYSTS_DROP('db2ts', 'myTextIndex', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
Значение выходных параметров
-----
Имя параметра   : MESSAGE
Значение параметра: Операция выполнена успешно.
Состояние возврата = 0
```

Пример 2: В следующем примере SYSTS_DROP вызывается для отбрасывания индекса текстового поиска со значениями *схема_индекса* 'db2ts' и *имя_индекса* 'myTextIndex'. Такой индекс не существует, и происходит ошибка.

```
CALL SYSPROC.SYSTS_DROP('db2ts', 'myTextIndex', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
SQL20427N Ошибка при выполнении процедуры или команды
управления текстовым поиском. Сообщение об ошибке "CIE00316 Индекс текстового поиска
"db2ts"."myTextIndex" не существует. ". SQLSTATE 38N14
```

Замечания по использованию

- Процедуры управления текстовыми индексами используют уже существующее соединение с базой данных. В зависимости от завершения этих процедур для текущей транзакции выполняется принятие или откат. Поэтому рекомендуется перед выполнением этой процедуры управления текстовым поиском выполнить принятие всех изменений, внесенных транзакциями, чтобы избежать нежелательного влияния принятия или отката этой процедуры. Один из способов достигнуть этого - включить AUTOCOMMIT.

Примечание: Если для транзакции не выполнено принятие до вызова операции управления текстовым поиском, может возникнуть тупиковая ситуация, так как могут быть затронуты одни и те же объекты базы данных и операция будет ожидать завершения предыдущего шага. Это может случиться, например, если при выключенной опции AUTOCOMMIT создать таблицу, а затем создать текстовый индекс, не вызвав принятие в явном виде между этими двумя транзакциями.

- Нельзя одновременно вызывать для индекса текстового поиска несколько процедур или команд, если они могут конфликтовать. Некоторые из конфликтующих процедур и команд:
 - Процедура SYSTS_ALTER или команда db2ts ALTER INDEX
 - Процедура SYSTS_CLEAR_EVENTS или команда db2ts CLEAR EVENTS FOR INDEX

- Процедура SYSTS_DISABLE или команда db2ts DISABLE DATABASE FOR TEXT
- Команда db2ts STOP FOR TEXT
- Процедура SYSTS_UPDATE или команда db2ts UPDATE INDEX

В случае конфликта процедура возвращает SQLCODE -20426 и SQLSTATE 38H13.

- Отбрасывание пользовательской таблицы в DB2 не приводит к отбрасыванию индексов. Индексы надо отбросить вручную перед отбрасыванием таблицы.
- При выполнении этой процедуры
 - обновляется информация в каталоге текстового поиска. Таблицы кэша индекса и таблицы событий отбрасываются. Триггеры для пользовательской таблицы удаляются.
 - удаляется запись индекса в файле данных индекса текстового поиска. Этот файл содержит постоянное представление расписаний обновлений (в том числе пустых) для каждого индекса в экземпляре.
 - собрание, связанное с этим индексом текстового поиска, удаляется.
- Если после отбрасывания индекса текстового поиска вы планируете создать новый индекс для того же текстового столбца, перед созданием нового индекса текстового поиска надо отсоединиться от базы данных, а затем соединиться с ней снова.

Процедура SYSTS_ALTER - Изменение характеристик обновления индекса

Эта процедура изменяет характеристики обновления индекса

Эта процедура вызывает на сервере баз данных команду администрирования текстового поиска **ALTER INDEX**.

Синтаксис

```

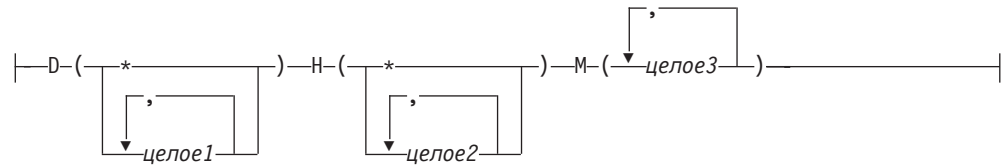
▶▶ SYSTS_ALTER (—схема_индекса—, —имя_индекса—, —————▶
▶ | изменить характеристики |—————▶
▶ | варианты |, —локаль_сообщений—, —сообщение—)————▶
  
```

Изменить характеристики:

```

|—————▶
| UPDATE FREQUENCY — NONE —————▶
| | частота обновления |—————▶
|—————▶
| UPDATE MINIMUM — минимальное_число_изменений —————▶
  
```

частота обновления:



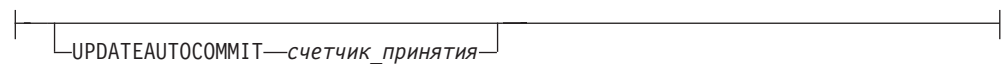
варианты:



опции конфигурации индекса:



значение_опции:



опции активации:

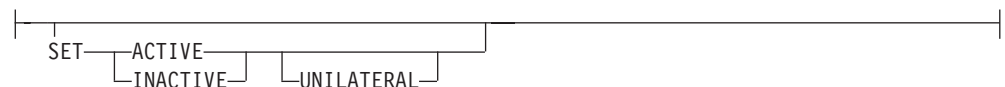


Схема - SYSPROC.

Параметры процедуры

схема_индекса

Входной аргумент типа VARCHAR(128), задающий схему индекса текстового поиска. Значение *схема_индекса* должно соответствовать ограничениям именования для имен схем DB2. Если аргумент является пустым значением или пустой строкой, используется значение CURRENT SCHEMA. Аргумент *схема_индекса* регистрозависим.

имя_индекса

Входной аргумент типа VARCHAR(128), задающий имя индекса. Вместе с аргументом *схема_индекса* уникальным образом задает индекс текстового поиска в базе данных. Аргумент *имя_индекса* регистрозависим.

Изменить характеристики

Входной аргумент типа VARCHAR(32K), задающий опции изменения. Возможны следующие опции изменения:

UPDATE FREQUENCY

Задает частоту выполнения обновлений индекса. Индекс будет обновлен, если число изменений не меньше значения, заданного для **UPDATE MINIMUM**.

Частота обновления **NONE** указывает, что индекс не будет обновляться. Это может пригодиться для текстового столбца в таблице, данные в которой никогда не изменяются. Это полезно также, если пользователь хочет обновлять индекс вручную (с помощью команды **UPDATE INDEX**).

Автоматические обновления могут выполняться, только если выполнена команда **START FOR TEXT** и запущены службы экземпляра DB2 Text Search.

Значение частоты по умолчанию берется из производной таблицы SYSIBM.TSDEFAULTS, из строки с DEFAULTNAME='UPDATEFREQUENCY'.

NONE

К этому текстовому индексу не будет применяться автоматическое обновление. Все последующие обновления индекса должны запускаться вручную.

D День недели для обновления индекса.

* Каждый день недели.

целое1

Определенные дни недели, от воскресенья (0) до субботы (6)

H Час указанного дня для обновления индекса.

* Каждый час дня.

целое2

Определенные часы дня, от полуночи до 11 вечера: от 0 до 23

M Минута указанного часа для обновления индекса.

целое3

Указывается как начало часа (0) или как кратное 5 число минут с начала часа: 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 или 55

Если вы не задали опцию **UPDATE FREQUENCY**, параметры частоты обновления не изменяются.

UPDATE MINIMUM *минимальное_число_изменений*

Задаёт минимальное число изменений в текстовых документах перед инкрементным обновлением индекса. Несколько изменений одного и того же текстового документа рассматриваются как отдельные изменения. Если вы не задали опцию **UPDATE MINIMUM**, этот параметр не изменяется.

INDEX CONFIGURATION (*значение_опции*)

Начиная с версии 9.7 с пакетом Fix Pack 3 и более новыми пакетами Fix Pack, этот необязательный входной аргумент типа VARCHAR(32K) позволяет изменять параметры конфигурации текстового индекса. Поддерживаются следующие опции:

Таблица 10. Спецификации для опции - значения

Опция	Значение	Тип данных	Описание
UPDATEAUTO COMMIT	<i>commitcount</i> <i>_number</i>	Целое	<p>Задаёт число изменений индекса, после которого автоматически выполняется принятие для сохранения сделанной работы, для первого обновления индекса или для инкрементных обновлений.</p> <ul style="list-style-type: none"> Первоначальное обновление индекса обрабатывает пакеты документов из базовой таблицы после того, как будет активирован триггер для захвата изменений данных. Когда число изменённых документов достигает COMMITCOUNT, сервер выполняет промежуточное принятие. Записи журнала, сгенерированные необработанными документами, будут удалены из таблицы кэша. Использование опции UPDATEAUTOCOMMIT для первоначального обновления текстового индекса ведёт к значительному увеличению времени выполнения. Для инкрементных обновлений обработанные записи журнала удаляются из таблицы кэша с каждым промежуточным принятием. COMMITCOUNT подсчитывает число изменённых документов, а не число записей таблицы кэша.

опции активации

Начиная с версии 9.7 с пакетом Fix Pack 3 и более новыми пакетами Fix Pack, этот входной аргумент типа integer задаёт состояние текстового индекса.

ACTIVE

Задаёт активное состояние текстового индекса

INACTIVE

Задаёт неактивное состояние текстового индекса

UNILATERAL

Задаёт одностороннее изменение, которое действует только на состояние индексов текстового поиска DB2. Если указан этот аргумент, меняется на активное или неактивное состояние только индекса текстового поиска DB2. Без аргумента UNILATERAL состояния активации индексов DB2 Text Search и DB2 Net Search Extender меняются синхронно, так что активным будет только один из этих текстовых индексов.

Примечание: После изменения состояния индекса текстового поиска с активного на неактивное или наоборот введите команду **FLUSH PACKAGE CACHE DYNAMIC**. Она удаляет все кэшированные динамические операторы SQL из кэша пакетов и указывает DB2 неявным образом скомпилировать тот же оператор SQL при следующем требовании.

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий локаль, используемую для всех возвращаемых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Выходной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершённой операции.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен текстовый индекс
- Полномочия DBADM

Примеры

Пример 1: В следующем примере изменяются характеристики обновления индекса текстового поиска. Этот индекс был создан изначально со *схемой_индекса* 'db2ts' и *именем_индекса* 'myTextIndex'. Задание 'UPDATE FREQUENCY NONE' указывает более не обновлять индекс текстового поиска, так как, вероятно, изменения в связанном столбце таблицы не ожидаются. Указано, что сообщения об ошибках должны возвращаться по-русски. При успешном выполнении процедуры вызывающему возвращается сообщение выходного параметра.

```
CALL SYSPROC.SYSTS_ALTER('db2ts', 'myTextIndex',
    'UPDATE FREQUENCY NONE', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
Значение выходных параметров
-----
Имя параметра   : MESSAGE
Значение параметра: Операция выполнена успешно.
Return Status = 0
```

Пример 2: В следующем примере хранимая процедура SYSTS_ALTER вызывается для индекса текстового поиска со значениями *схема_индекса* 'db2ts' и *имя_индекса* 'myTextIndex'. Сделана попытка задать обновление этого индекса в начале каждого часа. Однако такой индекс не существует, и происходит ошибка.

```
CALL SYSPROC.SYSTS_ALTER('db2ts', 'myTextIndex',
    'update frequency D(*) H(*) M(0)', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
SQL20427N Ошибка при выполнении процедуры или команды
управления текстовым поиском. Сообщение об ошибке "C1E00316 Индекс текстового поиска
"db2ts"."myTextIndex" не существует. ". SQLSTATE 38N14
```

Замечания по использованию

- Процедуры управления текстовыми индексами используют уже существующее соединение с базой данных. В зависимости от завершения этих процедур для текущей транзакции выполняется принятие или откат. Поэтому вам может быть полезно выполнять принятие всех изменений транзакции во избежание неожиданных последствий такого принятия или отката. Один из способов достигнуть этого - включить AUTOCOMMIT для соединения.

Примечание: Если для транзакции не выполнено принятие до вызова операции управления текстовым поиском, может возникнуть тупиковая ситуация, так как могут быть затронуты одни и те же объекты базы данных и операция будет ожидать завершения предыдущего шага. Это может случиться, например, если при выключенной опции AUTOCOMMIT создать таблицу, а затем создать текстовый индекс, не вызвав принятие в явном виде между этими двумя транзакциями.

- Нельзя одновременно запускать для индекса текстового поиска несколько процедур или команд, если они могут конфликтовать. Некоторые из конфликтующих процедур и команд:

- Процедура SYSTS_ALTER или команда db2ts ALTER INDEX
- Процедура SYSTS_CLEAR_EVENTS или команда db2ts CLEAR EVENTS FOR INDEX
- Процедура SYSTS_DISABLE или команда db2ts DISABLE DATABASE FOR TEXT
- Процедура SYSTS_DROP или команда db2ts DROP INDEX
- Команда db2ts STOP FOR TEXT
- Процедура SYSTS_UPDATE или команда db2ts UPDATE INDEX

В случае конфликта процедура возвращает SQLCODE -20426 и SQLSTATE 38N13.

- При выполнении этой процедуры
 - изменяется содержимое производной таблицы текстового поиска DB2 SYSIBMTS.TSLOCKS.
 - изменяется запись индекса в файле данных индекса текстового поиска. Этот файл содержит постоянное представление расписаний обновлений (в том числе пустых) для каждого индекса в экземпляре.
- Результат активации индексов зависит от исходного состояния индекса. Эти результаты описаны в следующей таблице.

Таблица 11. Изменения состояния при отсутствии недопустимых индексов:

Начальное состояние DB2 Text Search или Net Search Extender	опция активации Active	опция активации Active Unilateral	опция активации Inactive	опция активации Inactive Unilateral
Активен / Неактивен	Без изменений	Без изменений	Неактивен / Активен	Неактивен / Неактивен
Неактивен / Активен	Активен / Неактивен	Ошибка	Без изменений	Без изменений
Неактивен / Неактивен	Активен / Неактивен	Активен / Неактивен	Неактивен / Активен	Без изменений

При конфликтах активных индексов возвращаются сообщения об ошибках SQL20427N и CIE0379E.

Процедура SYSTS_UPDATE - Обновление индекса текстового поиска

Эта процедура обновляет индекс текстового поиска, чтобы отразить текущее содержимое текстового столбца, с которым связан индекс.

Во время выполнения изменения возможен поиск. До завершения изменения поиск работает с частично обновленным индексом.

Эта процедура вызывает на сервере баз данных команду администрирования текстового поиска **UPDATE INDEX**.

Синтаксис

```
►►SYSTS_UPDATE(—схема_индекса—, —имя_индекса—, —————►►
►—опции_изменения—, —локаль_сообщений—, —сообщение—)►►
```


Схема - SYSPROC.

Параметры процедуры

схема_индекса

Входной аргумент типа VARCHAR(128), задающий схему индекса текстового поиска. Значение *схема_индекса* должно соответствовать ограничениям именования для имен схем DB2. Если аргумент является пустым значением или пустой строкой, используется значение CURRENT SCHEMA. Аргумент *схема_индекса* регистрозависим.

имя_индекса

Входной аргумент типа VARCHAR(128), задающий имя индекса. Вместе с аргументом *схема_индекса* уникальным образом задает индекс текстового поиска в базе данных. Аргумент *имя_индекса* регистрозависим.

опции_изменения

Входной аргумент типа VARCHAR(32K), задающий опции обновления. Возможные значения:

- USING UPDATE MINIMUM: этот параметр соответствует параметру UPDATE MINIMUM управляющей команды текстового поиска **CREATE INDEX** и процедуры SYSTS_CREATE.
- UPDATEAUTOCOMMIT: этот параметр переопределяет значение commitcount, определенное в текстовом индексе, на время выполнения обновления.
- NULL или пустая строка (""): обновление запускается безусловно при вызове процедуры.

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий локаль, используемую для всех возвращаемых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Выходной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершенной операции.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен текстовый индекс
- Полномочия DATAACCESS

Примеры

Пример 1: В следующем примере выполняется обновление индекса текстового поиска, созданного со значениями *схема_индекса* 'db2ts' и *имя_индекса* 'myTextIndex'. Пустое значение на месте параметра *опции_обновления* означает, что при вызове этой хранимой процедуры обновление запускается безусловно. Указано, что сообщения об ошибках должны возвращаться по-русски. При успешном выполнении процедуры вызывающему возвращается сообщение выходного параметра.

```
CALL SYSPROC.SYSTS_UPDATE('db2ts', 'myTextIndex', '', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

Значение выходных параметров

```
-----  
Имя параметра      : MESSAGE  
Значение параметра: Операция выполнена успешно.  
Return Status = 0
```

Пример 2: В следующем примере SYSTS_UPDATE вызывается для обновления индекса текстового поиска со значениями *схема_индекса* 'db2ts' и *имя_индекса* 'myTextIndex'. Такой индекс не существует, и происходит ошибка.

```
CALL SYSPROC.SYSTS_UPDATE('db2ts', 'myTextIndex', 'USING UPDATE MINIMUM',  
'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
SQL20427N Ошибка при выполнении процедуры или команды  
управления текстовым поиском. Сообщение об ошибке "CIE00316 Индекс текстового поиска  
"db2ts"."myTextIndex" не существует. ". SQLSTATE 38H14
```

Замечания по использованию

- Процедуры управления текстовыми индексами используют уже существующее соединение с базой данных. В зависимости от завершения этих процедур для текущей транзакции выполняется принятие или откат. Поэтому вам может быть полезно выполнять принятие всех изменений транзакции во избежание неожиданных последствий такого принятия или отката. Один из способов достигнуть этого - включить AUTOCOMMIT.
- Нельзя одновременно запускать для индекса текстового поиска несколько процедур или команд, если они могут конфликтовать. Некоторые из конфликтующих процедур и команд:
 - Процедура SYSTS_ALTER или команда db2ts ALTER INDEX
 - Процедура SYSTS_CLEAR_EVENTS или команда db2ts CLEAR EVENTS FOR INDEX
 - Процедура SYSTS_DISABLE или команда db2ts DISABLE DATABASE FOR TEXT
 - Процедура SYSTS_DROP или команда db2ts DROP INDEX
 - Команда db2ts STOP FOR TEXT
 - Процедура SYSTS_UPDATE или команда db2ts UPDATE INDEX

В случае конфликта процедура возвращает SQLCODE -20426 и SQLSTATE 38H13.

- Возврат из этой процедуры не происходит, пока не будет полностью завершена обработка обновления индекса. Продолжительность зависит от числа индексируемых документов и от числа уже проиндексированных документов. Имя собрания для индекса можно получить из столбца COLLECTIONNAME производной таблицы SYSIBMTS.TSCOLLECTIONNAMES.
- Если в отдельных документах есть ошибки, эти документы надо исправить. Первичные ключи документов с ошибками можно найти в таблице событий для индекса. Если вы измените соответствующие строки в пользовательской таблице, при следующем вызове SYSTS_UPDATE эти документы будут обработаны повторно.
- При выполнении этой процедуры
 - в таблицу событий вставляются строки (с информацией об ошибках анализа) В случае инкрементного обновления удаляется информация из промежуточной таблицы индекса. Перед первым обновлением создаются триггеры для пользовательской таблицы.
 - обновляется собрание; новые или измененные документы анализируются и индексируются, а удаленные документы удаляются из индекса.

Процедура SYSTS_CLEAR_EVENTS - Удаление событий индексации из таблицы событий индекса

Эта процедура удаляет события индексации из таблицы событий индекса, используемой для управления.

Имя таблицы событий можно найти в столбце EVENTVIEWNAME производной таблицы SYSIBMTS.TSINDEXES. Каждая операция обновления индекса, обработавшая хотя бы один документ, заносит информационные записи и, в некоторых случаях, записи об ошибках в таблицу событий. Для автоматических обновлений таблицу событий следует регулярно проверять. Ошибки, связанные с конкретными документами, надо исправлять, изменяя содержимое документа. После исправления этих ошибок события могут быть (и должны быть ради экономии места) очищены.

Эта процедура передает на сервер базы данных команду управления текстовым поиском **CLEAR EVENTS FOR INDEX**.

Синтаксис

```
►►—SYSTS_CLEAR_EVENTS—(—схема_индекса—,—имя_индекса—,——————►  
►—локаль_сообщений—,—сообщение—)—————►
```

Схема - SYSPROC.

Параметры процедуры

схема_индекса

Входной аргумент типа VARCHAR(128), задающий схему индекса текстового поиска. Значение *схема_индекса* должно соответствовать ограничениям именования для имен схем DB2. Если аргумент является пустым значением или пустой строкой, используется значение CURRENT SCHEMA. Аргумент *схема_индекса* регистрозависим.

имя_индекса

Входной аргумент типа VARCHAR(128), задающий имя индекса. Вместе с аргументом *схема_индекса* уникальным образом задает индекс текстового поиска в базе данных. Аргумент *имя_индекса* регистрозависим.

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий локаль, используемую для всех возвращаемых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Выходной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершенной операции.

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегия CONTROL для таблицы, на которой определен индекс
- Полномочия DBADM

Примеры

Пример 1: В этом примере команда SYSTS_CLEAR_EVENTS вызывается для индекса текстового поиска, созданного с именем_схемы 'db2ts' и именем_индекса 'myTextIndex'. Указано, что сообщения об ошибках должны возвращаться по-русски. При успешном выполнении процедуры вызывающему возвращается сообщение выходного параметра.

```
CALL SYSPROC.SYSTS_CLEAR_EVENTS('db2ts', 'myTextIndex', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
Значение выходных параметров
-----
Имя параметра      : MESSAGE
Значение параметра: Операция выполнена успешно.
Состояние возврата = 0
```

Пример 2: В следующем примере SYSTS_CLEAR_EVENTS вызывается для очистки таблицы событий для индекса текстового поиска со схемой_индекса 'db2ts' и именем_индекса 'myTextIndex'. Такой индекс не существует, и происходит ошибка.

```
CALL SYSPROC.SYSTS_CLEAR_EVENTS('db2ts', 'myTextIndex', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
SQL20427N Ошибка при выполнении процедуры или команды
управления текстовым поиском. Сообщение об ошибке "CIE00316 Индекс текстового поиска
"db2ts"."myTextIndex" не существует. ". SQLSTATE 38N14
```

Замечания по использованию

- Процедуры управления текстовыми индексами используют уже существующее соединение с базой данных. В зависимости от завершения этих процедур для текущей транзакции выполняется принятие или откат. Поэтому рекомендуется перед выполнением этой процедуры управления текстовым поиском выполнить принятие всех изменений, внесенных транзакциями, чтобы избежать нежелательного влияния принятия или отката этой процедуры. Один из способов достигнуть этого - включить AUTOCOMMIT.

Примечание: Если для транзакции не выполнено принятие до вызова операции управления текстовым поиском, может возникнуть тупиковая ситуация, так как могут быть затронуты одни и те же объекты базы данных и операция будет ожидать завершения предыдущего шага. Это может случиться, например, если при выключенной опции AUTOCOMMIT создать таблицу, а затем создать текстовый индекс, не вызвав принятие в явном виде между этими двумя транзакциями.

- Нельзя одновременно запускать для индекса текстового поиска несколько процедур или команд, если они могут конфликтовать. Некоторые из конфликтующих процедур и команд:
 - Процедура SYSTS_ALTER или команда db2ts ALTER INDEX
 - Процедура SYSTS_DISABLE или команда db2ts DISABLE DATABASE FOR TEXT
 - Процедура SYSTS_DROP или команда db2ts DROP INDEX
 - Команда db2ts STOP FOR TEXT
 - Процедура SYSTS_UPDATE или команда db2ts UPDATE INDEX

В случае конфликта процедура возвращает SQLCODE -20426 и SQLSTATE 38N13.

- Если планируются регулярные обновления (смотрите опции UPDATE FREQUENCY в процедурах SYSTS_CREATE и SYSTS_ALTER), таблицу событий надо проверять регулярно.

- Чтобы очистить таблицу событий текстового поиска DB2 для индекса текстового поиска, после проверки причины события и устранения источника ошибки используйте процедуру SYSTS_CLEAR_EVENTS или команду db2ts **CLEAR EVENTS FOR INDEX**.
- Убедитесь, что во все строки, упомянутые в таблице событий, внесены изменения. Изменяя строки в пользовательской таблице, вы гарантируете, что при новом запуске процедуры SYSTS_UPDATE или команды db2ts **UPDATE INDEX** будет выполнена новая попытка проиндексировать ошибочные документы.
- При выполнении этой команды таблица событий очищается.

Процедура SYSTS_CLEAR_COMMANDLOCKS - Удаление блокировок команд для индексов текстового поиска

Эта процедура удаляет все блокировки команд для определенного индекса текстового поиска или для всех индексов текстового поиска в базе данных.

Блокировка команды создается в начале выполнения команды текстового поиска и уничтожается при завершении этой команды. Она предотвращает конфликты между различными командами.

Для всех блокировок, связанных с более не выполняемыми процессами, очистка выполняется автоматически. Это делается, чтобы индекс текстового поиска был доступен для нового требования поиска. Использование этой процедуры требуется в том редком случае, когда блокировка остается неудаленной в результате неожиданного поведения системы, и ее требуется очистить явно.

Эта процедура передает на сервер базы данных команду управления текстовым поиском **CLEAR COMMAND LOCKS**.

Синтаксис

```

▶▶ SYSTS_CLEAR_COMMANDLOCKS (—схема_индекса—, —имя_индекса—, —————▶
▶ —локаль_сообщений—, —сообщение—) —————▶▶

```

Схема - SYSPROC.

Параметры процедуры

схема_индекса

Входной аргумент типа VARCHAR(128), задающий схему текстового индекса. Значение *схема_индекса* должно соответствовать ограничениям именования для имен схем DB2. Если аргумент является пустым значением или пустой строкой, используется значение CURRENT SCHEMA. Аргумент *схема_индекса* регистрозависим.

имя_индекса

Входной аргумент типа VARCHAR(128), задающий имя индекса. Вместе с аргументом *схема_индекса* уникальным образом задает индекс текстового поиска в базе данных. Если аргумент является пустым значением или пустой строкой, процедура удаляет блокировки команд для всех индексов текстового поиска в базе данных. Аргумент *имя_индекса* регистрозависим.

локаль_сообщений

Входной аргумент типа VARCHAR(33), задающий локаль, используемую для

всех возвращаемых сообщений об ошибках. Если аргумент - пустое значение или пустая строка, или же на сервере недоступны файлы сообщений для указанной локали, используется 'en_US'.

сообщение

Выходной аргумент типа VARCHAR(32K), задающий предупреждение или информационное сообщение для успешно завершенной операции.

Авторизация

Если имя индекса не задано, у пользователя *имя_пользователя* соединения с базой данных должны быть полномочия DBADM. Для очистки блокировки команд конкретного индекса у пользователя *имя_пользователя* соединения с базой данных должна быть привилегия CONTROL для таблицы, на которой создан этот индекс текстового поиска.

Примеры

Пример 1: В этом примере SYSTS_CLEAR_COMMANDLOCKS вызывается для индекса текстового поиска с *именем_схемы* 'db2ts' и *именем_индекса* 'myTextIndex'. Сообщения об ошибках должны возвращаться по-русски. При успешном выполнении процедуры вызывающему возвращается сообщение выходного параметра.

```
CALL SYSPROC.SYSTS_CLEAR_COMMANDLOCKS('db2ts', 'myTextIndex', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
Значение выходных параметров
-----
Имя параметра      : MESSAGE
Значение параметра: Операция выполнена успешно.
Состояние возврата = 0
```

Пример 2: В следующем примере SYSTS_CLEAR_COMMANDLOCKS вызывается для очистки блокировок команд для индекса текстового поиска со *схемой_индекса* 'db2ts' и *именем_индекса* 'myTextIndex'. Такой индекс не существует, и процедура возвращает ошибку.

```
CALL SYSPROC.SYSTS_CLEAR_COMMANDLOCKS('db2ts', 'myTextIndex', 'ru_RU', ?)
```

Ниже приведен пример вывода этого запроса.

```
SQL20427N Ошибка при выполнении процедуры или команды
управления текстовым поиском. Сообщение об ошибке "CIE00316 Индекс текстового поиска
"db2ts"."myTextIndex" не существует. ". SQLSTATE 38H14
```

Замечания по использованию

- Процедуры управления текстовыми индексами используют уже существующее соединение с базой данных. В зависимости от завершения этих процедур для текущей транзакции выполняется принятие или откат. Поэтому рекомендуется перед выполнением этой процедуры управления текстовым поиском выполнить принятие всех изменений, внесенных транзакциями, чтобы избежать нежелательного влияния принятия или отката этой процедуры. Один из способов достигнуть этого - включить AUTOCOMMIT.

Примечание: Если для транзакции не выполнено принятие до вызова операции управления текстовым поиском, может возникнуть тупиковая ситуация, так как могут быть затронуты одни и те же объекты базы данных и операция будет ожидать завершения предыдущего шага. Это может случиться, например, если при

выключенной опции AUTOCOMMIT создать таблицу, а затем создать текстовый индекс, не вызвав принятие в явном виде между этими двумя транзакциями.

- Для проверки, удерживает ли поток или процесс блокировки, можно использовать информацию о процессе и потоке в производной таблице SYSIBMTS.TSLOCKS. Блокировки существующих процессов, принадлежащие выполняемым процедурам или командам управления текстовым поиском (например, SYSTS_UPDATE или UPDATE INDEX) очищать не следует.
- Вы вызываете эту процедуру потому, что процесс, которому принадлежит блокировка команды, не работает. В таком случае эта команда (владелец блокировки) не может завершиться, и индекс не может стать доступным. Вам надо предпринять действия по исправлению ситуации. Предположим, процесс, выполняющий команду DROP INDEX, неожиданно завис. Он удалил некоторые данные индексов, но не всю информацию каталога и собраний. Блокировка команды остается нетронутой. После очистки блокировки команды DROP INDEX вы можете захотеть повторить процедуру SYSTS_DROP. В другом примере процесс, выполняющий процедуру SYSTS_CREATE, неожиданно завис. Он создал часть информации каталога индексов и собраний, но не всю. Блокировка команды остается нетронутой. После очистки блокировки можно выполнить процедуры SYSTS_DROP и SYSTS_CREATE.
- При выполнении этой процедуры изменяется содержимое производной таблицы текстового поиска DB2 SYSIBMTS.TSLOCKS.

Глава 10. Управляющие производные таблицы текстового поиска

DB2 Text Search создает и обслуживает несколько управляющих производных таблиц, описывающих индексы текстового поиска в базе данных и их свойства.

Не обновляйте эти производные таблицы, если не было прямого указания сделать это.

Следующие производные таблицы отражают текущую конфигурацию вашей системы:

- Производные таблицы уровня базы данных:
 - SYSIBMTS.TSDEFAULTS
 - SYSIBMTS.TSLOCKS
- Производные таблицы уровня индекса:
 - SYSIBMTS.TSINDEXES
 - SYSIBMTS.TSCONFIGURATION
 - SYSIBMTS.TSCOLLECTIONNAMES
 - SYSIBMTS.TSEVENT_#####
 - SYSIBMTS.TSSTAGING_#####

Производная таблица SYSIBMTS.TSDEFAULTS

SYSIBMTS.TSDEFAULTS выводит все значения по умолчанию для всех индексов текстового поиска в базе данных.

Значения по умолчанию доступны в этом представлении как пары атрибут - значение.

Таблица 12. Производная таблица SYSIBMTS.TSDEFAULTS

Имя столбца	Тип данных	Допустимы пустые значения?	Описание
DEFAULTNAME	VARCHAR(30)	НЕТ	Параметры по умолчанию базы данных для текстового поиска
DEFAULTVALUE	VARCHAR(512)	НЕТ	Значения для параметров по умолчанию базы данных для текстового поиска

Следующие значения используются по умолчанию для команд db2ts **CREATE INDEX**, **ALTER INDEX**, **UPDATE INDEX** и **CLEAR EVENTS FOR INDEX**:

- CODEPAGE. Начальное значение по умолчанию кодовой страницы для новых индексов - это кодовая страница базы данных.
- FORMAT. Начальное значение по умолчанию для формата документа - простой текст.
- LANGUAGE. Начальное значение по умолчанию для индексации документов - en_US.
- UPDATEFREQUENCY. Начальное значение по умолчанию для расписания обновления для новых индексов - NONE.

- UPDATEMINIMUM. Начальное значение по умолчанию для обновления для новых индексов - 1, то есть инкрементные обновления могут выполняться после каждого изменения.
- UPDATEAUTOCOMMIT. Начальное значение по умолчанию для обновления новых индексов равно 0, то есть промежуточные принятия при чтении документов из текстовых столбцов DB2 не выполняются. Это значение зарезервировано, и изменить его нельзя.

Командами **db2ts** нельзя изменить значения по умолчанию на уровне базы данных.

Производная таблица SYSIBMTS.TSLOCKS

Можно просмотреть информацию блокировок команд на уровне базы данных и на уровне индекса, используя SYSIBMTS.TSLOCKS.

Таблица 13. Производная таблица SYSIBMTS.TSLOCKS

Имя столбца	Тип данных	Допустимы пустые значения?	Описание
COMMAND	VARCHAR(30)	НЕТ	Имя команды, создавшей блокировку. Возможные значения: CREATE INDEX, ALTER INDEX, DROP INDEX, UPDATE INDEX, CLEAR EVENTS, DISABLE DATABASE.
LOCKSCOPE	VARCHAR(30)	НЕТ	Область действия блокировки. Возможные значения: DATABASE или INDEX.
INDSCHEMA	VARCHAR(128)	ДА	Имя схемы индекса текстового поиска (только при LOCKSCOPE = INDEX)
INDNAME	VARCHAR(128)	ДА	Неспецифицированное имя индекса текстового поиска (только при LOCKSCOPE = INDEX)
PROCESSID	INTEGER	НЕТ	Идентификатор процесса, создавшего блокировку
THREADID	INTEGER	НЕТ	Идентификатор потока, создавшего блокировку
LOCKCREATETIME	TIMESTAMP	НЕТ	Отметка времени, когда была предоставлена блокировка

Информация о процессе или потоке дает возможность проверить, существует ли еще этот процесс или поток. Если они больше не существуют, удалите блокировку командой **db2ts CLEAR COMMAND LOCKS**.

SYSIBMTS.TSCOLLECTIONNAMES view

Производная таблица SYSIBMTS.TSCOLLECTIONNAMES содержит имена собраний.

Каждая строка представляет собрание для индекса текстового поиска.

Таблица 14. SYSIBMTS.TSCOLLECTIONNAMES view

Имя столбца	Тип данных	Допустимы пустые значения?	Описание
INDSCHEMA	VARCHAR(128)	НЕТ	Имя схемы индекса текстового поиска
INDNAME	VARCHAR(128)	НЕТ	Неспецифицированное имя индекса текстового поиска
COLLECTIONNAME	VARCHAR(132)	НЕТ	Имя связанного собрания на сервере текстового поиска

Производная таблица SYSIBMTS.TSINDEXES

При помощи SYSIBMTS.TSINDEXES можно просмотреть параметры команды db2ts **CREATE INDEX** и сгенерированные значения.

Можно запросить производную таблицу, чтобы получить информацию об индексах. В этом примере используется схема индекса:

```
db2 "SELECT COLNAME from SYSIBMTS.TSINDEXES where INDSHEMA=имя-схемы
and INDNAME=имя-индекса"
```

Таблица 15. Производная таблица SYSIBMTS.TSINDEXES

Имя столбца	Тип данных	Допустимы пустые значения?	Описание
INDSCHEMA	VARCHAR (128)	НЕТ	Имя схемы для индекса текстового поиска.
INDNAME	VARCHAR (128)	НЕТ	Неспецифицированное имя индекса текстового поиска.
TABSCHEMA	VARCHAR (128)	НЕТ	Имя схемы базовой таблицы.
TABNAME	VARCHAR (128)	НЕТ	Неспецифицированное имя базовой таблицы.
COLNAME	VARCHAR (128)	НЕТ	Столбец, для которого был создан индекс текстового поиска.
CODEPAGE	INTEGER	НЕТ	Кодовая страница документа для индекса текстового поиска.
LANGUAGE	VARCHAR(5)	НЕТ	Язык документа для индекса текстового поиска.
FORMAT	VARCHAR(30)	ДА	Формат документа.
FUNCTIONSCHEMA	VARCHAR (128)	ДА	Схема для типа столбца.
FUNCTIONNAME	VARCHAR(18)	ДА	Имя функции преобразования типа столбца.
COLLECTIONDIRECTORY	VARCHAR(512)	ДА	Каталог для файлов индекса текстового поиска.
UPDATEFREQUENCY	VARCHAR(300)	НЕТ	Критерий триггера для применения обновлений к индексу.
UPDATEMINIMUM	INTEGER	ДА	Минимальное число записей в таблице журнала перед инкрементным обновлением. Меньшее значение означает лучшее соответствие между столбцом таблицы и индексом текстового поиска. Но при этом также увеличиваются использование памяти на индексацию для текстового поиска.
EVENTVIEWSCHEMA	VARCHAR (128)	НЕТ	Схема для производной таблицы событий, созданной для индекса текстового поиска (всегда SYSIBMTS).
EVENTVIEWNAME	VARCHAR (128)	НЕТ	Имя производной таблицы событий, созданной для индекса текстового поиска.
STAGINGVIEWSCHEMA	VARCHAR (128)	ДА	Схема для производной таблицы журналов, созданной для индекса текстового поиска (всегда SYSIBMTS).
STAGINGVIEWNAME	VARCHAR (128)	ДА	Имя производной таблицы журналов, созданной для индекса текстового поиска.
REORGAUTOMATIC	INTEGER	ДА	Зарезервировано (не поддерживается в этом выпуске). Значение всегда равно 1.
RECREATEONUPDATE	INTEGER	НЕТ	Зарезервировано (не поддерживается в этом выпуске). Значение всегда равно 0.
ATTRIBUTES	VARCHAR(32000)	ДА	Зарезервировано (не поддерживается в этом выпуске).
COLLECTIONNAMEPREFIX	VARCHAR (128)	НЕТ	Префикс имени собрания на сервере текстового поиска.
COMMENT	VARCHAR(512)	ДА	Комментарий, заданный в свойствах индекса команды CREATE INDEX .

Производная таблица SYSIBMTS.TSCONFIGURATION

Информация о параметрах конфигурации индекса доступна в производной таблице SYSIBMTS.TSCONFIGURATION.

Каждая строка представляет параметр индекса текстового поиска.

Следующий пример запроса производной таблицы использует имя индекса:

```
db2 "SELECT VALUE from SYSIBMTS.TSCONFIGURATION where INDSHEMA=имя-схемы  
and INDNAME=имя-индекса and PARAMETER ='параметр'"
```

Таблица 16. Производная таблица SYSIBMTS.TSCONFIGURATION

Имя столбца	Тип данных	Допустимы пустые значения?	Описание
INDSCHEMA	VARCHAR(128)	НЕТ	Имя схемы индекса текстового поиска
INDNAME	VARCHAR(128)	НЕТ	Неспецифицированное имя индекса текстового поиска
PARAMETER	VARCHAR(30)	НЕТ	Имя параметра конфигурации
VALUE	VARCHAR(512)	НЕТ	Значение параметра

Столбец PARAMETER содержит имена параметров конфигурации индекса текстового поиска, заданные при помощи оператора CREATE INDEX, и имена некоторых параметров из производной таблицы SYSIBMTS.TSDEFAULTS.

Производная таблица SYSIBMTS.TSEVENT

Производная таблица содержит информацию о состоянии индексации и о событиях ошибок.

В базе данных может быть несколько производных таблиц с префиксом SYSIBMTS.TSEVENT_. У каждой производной таблицы свое значение *nnnnnn*; это внутренний идентификатор, указывающий на соответствующий текстовый индекс, с которым связана производная таблица. Чтобы выяснить, какой индекс текстового поиска связан с конкретной производной таблицей, запросите производную таблицу SYSIBMTS.TSINDEXES, задав поиск имени схемы и имени производной таблицы в столбцах EVENTVIEWSCHEMA и EVENTVIEWNAME. Этот запрос возвращает одну строку, в которой описываются нужные индекс текстового поиска и пользовательская таблица.

Число столбцов в этом представлении зависит от числа столбцов первичного ключа в пользовательской таблице. Столбцы PK1..PK*n* аналогичны столбцам первичного ключа пользовательской таблицы. Тип данных в каждом из столбцов производной таблицы точно соответствует типу данных соответствующего столбца первичного ключа.

Каждая строка в этой производной таблице представляет сообщение команды **UPDATE INDEX** для индекса текстового поиска. Например, в строке может быть указано, что команда **UPDATE INDEX** была запущена или завершилась. Строка может также описывать ошибку, которая возникла при индексации текстового документа. Этот текстовый документ можно идентифицировать, получив значения из столбцов первичного ключа в строке этой производной таблицы и найдя их в пользовательской таблице.

Примечание: Если документ был вставлен, а потом удален до следующего обновления индекса, в таблице событий будет указано, что удалены два документа.

Можно очистить события командой db2ts **CLEAR EVENTS FOR INDEX**.

Таблица 17. Производная таблица событий

Имя столбца	Тип данных	Допустимы пустые значения?	Описание
OPERATION	INTEGER	ДА	Операция (вставка, изменение или удаление) для базовой таблицы, которая отражается в индексе текстового поиска
TIME	TIMESTAMP	ДА	Отметка времени создания записи о событии
SEVERITY	INTEGER	ДА	Если сообщение относится к одному документу, здесь будет одно из следующих значений: <ul style="list-style-type: none"> • 1 = Информационное • 4 = Части документа были проиндексированы, но с предупреждением, как указано в сообщении • 8 = Документ не был проиндексирован, как указано в сообщении • 0 = Иное
SQLCODE	INTEGER	ДА	Код SQL соответствующей ошибки, если есть
MESSAGE	VARCHAR(1024)	ДА	Текстовая информация о конкретной ошибке
PK01	Тип данных первого столбца первичного ключа базовой таблицы	ДА	Значение первого столбца первичного ключа базовой таблицы индекса текстового поиска для строки, которая обрабатывалась, когда возникла ошибка
PK n	Тип данных последнего столбца первичного ключа базовой таблицы	ДА	Значение последнего столбца первичного ключа базовой таблицы индекса текстового поиска для строки, которая обрабатывалась, когда возникла ошибка

Кроме того, в этой производной таблице доступны информационные события, такие как запуск, принятие и завершение обработки изменений. В этом случае значения PK01, PK n и OPERATION - пустые. Кодовая страница и локаль MESSAGE соответствуют параметрам базы данных.

Производная таблица SYSIBMTS.TSSTAGING

Промежуточная таблица хранит операции изменения пользовательской таблицы, для которой требуется синхронизация с индексом текстового поиска.

Триггеры создаются для пользовательской таблицы, чтобы передавать информацию об изменениях в промежуточную таблицу. Команда **UPDATE INDEX** читает эти записи и, после успешной синхронизации, удаляет их.

В базе данных может быть несколько производных таблиц с префиксом SYSIBMTS.TSSTAGING_. У каждой производной таблицы свое значение $nnnnnn$; это внутренний идентификатор, указывающий на соответствующий текстовый индекс, с которым связана производная таблица. Чтобы выяснить, какой индекс текстового поиска связан с конкретной производной таблицей, запросите производную таблицу SYSIBMTS.TSINDEXES, задав поиск имени схемы и имени производной таблицы в столбцах STAGINGVIEWSCHEMA и STAGINGVIEWNAME. Запрос возвращает одну строку, в которой описываются нужные индекс текстового поиска и пользовательская таблица.

Число столбцов в этом представлении зависит от числа столбцов первичного ключа в пользовательской таблице. Столбцы PK1..PK*n* аналогичны столбцам первичного ключа пользовательской таблицы. Тип данных в каждом из столбцов производной таблицы точно соответствует типу данных соответствующего столбца первичного ключа.

Каждая строка в этой производной таблице представляет операцию вставки, удаления или изменения для строки пользовательской таблицы или текстового документа. Этот текстовый документ можно идентифицировать, получив значения из столбцов первичного ключа в строке этой производной таблицы и найдя их в пользовательской таблице.

Для получения информации о производной таблице можно использовать следующий запрос:

```
db2 "SELECT STAGINGVIEWSHEMA, STAGINGVIEWNAME from SYSIBMTS.TSINDEXES
where INDSHEMA=имя-схемы and INDNAME=имя-индекса"
```

Таблица 18. Производная таблица SYSIBMTS.TSSTAGING

Имя столбца	Тип данных	Допустимы пустые значения?	Описание
OPERATION	INTEGER	НЕТ	Операция (вставка, изменение или удаление) для базовой таблицы, которая отражается в индексе текстового поиска
TIME	TIMESTAMP	Нет	Порядковый ID строки (когда сработал триггер вставки, изменения или удаления). Это отметка времени, которая, однако, не задает точного времени операции.
PK01	Тип данных первого столбца первичного ключа базовой таблицы	ДА	Первый столбец первичного ключа базовой таблицы.
PK <i>n</i>	Тип данных последнего столбца первичного ключа базовой таблицы	ДА	Последний столбец первичного ключа базовой таблицы.

Глава 11. Инструменты текстового поиска

Инструмент конфигурирования для DB2 Text Search

Инструмент конфигурирования - это инструмент командной строки, который можно использовать для начального конфигурирования текстового поиска или для настройки конфигурации текстового поиска, выполняемой после установки экземпляра DB2.

Конфигурирование текстового поиска при установке продукта баз данных DB2 удобнее, чем ручное конфигурирование DB2 Text Search инструментом конфигурирования, поскольку для последнего требуется задать все параметры, тогда как для утилиты установки - только два. Однако инструмент конфигурирования - единственное средство для внесения последующих изменений в следующие параметры:

- Различные пути, такие как путь конфигурации, путь журналов, путь установки и путь временного каталога
- Уровень записи в журнал
- Используемый размер кучи

Синтаксис инструмента

```
►►—configTool—обязательная_опция_команды—————►
►--configPath—абсолютный-путь-к-папке-конфигурации—————►
►┌───────────────────────────────────────────────────┐
│-locale—строка_локали_из_пяти_символов—┘           ──►
└───────────────────────────────────────────────────┘
```

Параметры

Обязательные параметры команды

configureHTTPListener

Позволяет сконфигурировать порт HTTP администрирования для связи между процессами текстового поиска, использующими TCP/IP. Укажите следующий аргумент:

-adminHTTPPort *значение*

Задает порт, который будет использоваться DB2 Text Search

Примечание:

- В разделе Использование перечислены и другие аргументы для **configureHTTPListener**, но их нельзя использовать с системой баз данных DB2 для Linux, UNIX и Windows.
- Перед конфигурированием порта администрирования HTTP надо остановить службы экземпляра текстового поиска.

upgradeInstance

Обновить сервер текстового поиска DB2 для ваших экземпляров можно при помощи команды **upgradeInstance**. Укажите следующий аргумент:

-installedConfigPath *значение*

Задает полный путь к каталогу установленной конфигурации. Например,

- В операционных системах Linux и UNIX этот путь может быть таким:
 - installedConfigPath *\$DB2DIR*/cfg/db2tss/config
 - configPath *\$INSTHOME*/sql1lib/db2tss/config

где *INSTHOME* - домашний каталог экземпляра, а *DB2DIR* - положение копии DB2 Версии 9.7.

- Для операционных систем Windows этот путь может быть таким:
 - installedConfigPath "%*DB2PATH*%\CFG\DB2TSS\CONFIG"
 - configPath "%*INSTPROFDIR*%\имя-экземпляра\DB2TSS\CONFIG"

где *DB2PATH* - положение копии DB2 Версии 9.7, а *INSTPROFDIR* - каталог профиля экземпляра.

configureParams

Позволяет сконфигурировать различные настраиваемые параметры для DB2 Text Search. Задайте один из следующих аргументов:

-logPath *путь*

Задаёт путь журналов, который будет использоваться DB2 Text Search.

-installPath *путь*

Задаёт полный путь к каталогу установки текстового поиска. Например, на платформах UNIX этот путь может быть *\$HOME/sql1lib/db2tss..*

-tempDirPath *путь*

Задаёт путь к временному каталогу, который будет использоваться DB2 Text Search.

-maxHeapSize *значение*

Задаёт размер кучи. Значение по умолчанию - 1,5 гигабайта.

-logLevel *значение*

Задаёт уровень записи в журнал для сообщений в файле журнала для механизма текстового поиска. Значения в порядке серьёзности сообщений:

- FINEST
- FINER
- FINE
- WARNING
- SEVERE
- OFF

generateToken

Генерирует маркер аутентификации. Единственный аргумент, который можно использовать:

-seed *значение*

Играет роль произвольной строки для генерации маркера аутентификации. Если задавать одно и то же значение, инструмент конфигурирования будет каждый раз генерировать один и тот же маркер.

printToken

Выводит маркер аутентификации. Эта опция не требует остановить службы экземпляра.

printAll

Выводит на стандартное устройство вывода значения всех настраиваемых свойств. Эта опция не требует остановки служб экземпляра.

printAdminHTTPPort

Выводит текущее значение порта HTTP управления. Эта команда не требует остановки служб экземпляра.

-configPath *абсолютный_путь_к_папке_конфигурации*

Задаёт абсолютный путь к папке конфигурации текстового поиска, например, /opt/ibm/search/config. Заметим, что если используется **configureParams**, то **configPath** служит не только для указания на каталог конфигурации, но и для изменения конфигурации, чтобы использовать заданный путь конфигурации.

-locale *локаль*

Задаёт пятисимвольную локаль для записи сообщений об ошибках, возвращаемых при использовании инструмента; например, en_US, ru_RU, fr_FR. Значение по умолчанию - en_US. Подробную информацию смотрите в разделе Supported locales (Поддерживаемые локали)

help

Выводит информацию об использовании команды инструмента конфигурирования. Информация об использовании выводится также, если запустить инструмент конфигурирования без обязательного аргумента.

Использование

- Если в описании параметра не указано другое, то для конфигурирования параметра надо остановить службы экземпляра текстового поиска. Если для задания значения конфигурации требуется остановить службы экземпляра, инструмент конфигурирования выявляет соответствующую ситуацию и возвращает сообщение о том, что службы экземпляра запущены и их надо остановить.
- Инструмент конфигурирования и другие инструменты командной строки не аутентифицируют пользователей. Но для запуска этих инструментов требуется доступ к файлам конфигурации. На платформах Windows, если внедрена расширенная модель защиты DB2, запускать инструменты командной строки могут только участники группы DB2ADMNS. Если расширенная модель защиты DB2 не внедрена, доступ к этим инструментам должен контролировать администратор баз данных DB2.
- Инструмент конфигурирования не обновляет файл служб. После использования инструмента конфигурирования для первоначального конфигурирования или для изменения существующей конфигурации надо обновить файл служб вручную.

Инструмент управления для DB2 Text Search

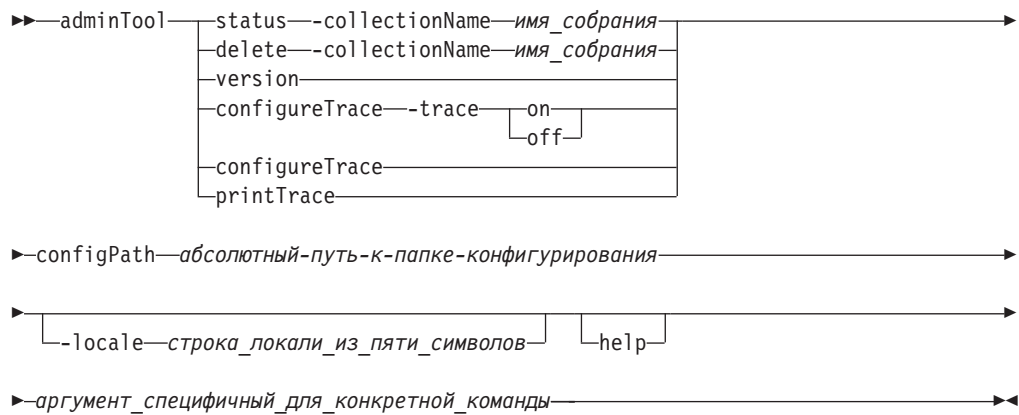
Инструмент управления - это инструмент командной строки, предоставляющий удобный способ выполнять несколько задач управления, связанных с DB2 Text Search.

К этим задачам относятся:

- Проверка состояния и свойств собраний текстового поиска
- Удаление бесхозных собраний текстового поиска
- Вывод версии сервера

Чтобы инструмент управления мог удалять собрания текстового поиска, должны работать службы экземпляра текстового поиска. Если не указано иное, на всех поддерживаемых платформах ко всем отсоединённым инструментам сервера Text Search надо добавлять суффикс .sh, например, adminTool.sh.

Синтаксис инструмента



Параметры

delete

Удаляет собрание текстового поиска, заданное параметром **-collectionName**.

Эта опция удаляет только бесхозные собрания. Убедитесь, что собрание не связано с действительным текстовым индексом. Кроме того, не удаляйте собрание во время индексирования документов.

состояние

Показывает состояние собрания текстового поиска, включая размер индекса, число включенных документов и ход работы по обновлению индекса.

-collectionName *имя_собрания*

Задаёт собрание текстового поиска, для которого нужно показать состояние.

version

Показывает версию сервера в следующем формате:

версия.выпуск.модификация.пакет_исправлений-протокол-сборка

где:

- версия.выпуск.модификация.пакет_исправлений: стандартное соглашение по нумерации версий, используемое для продуктов IBM
- протокол: число, задающее протокол для сериализации аргументов требования и данных, передаваемых между сервером и клиентом.
- сборка: число, указывающее номер сборки сервера

Например,

1.0.0.0-1.1-290

указывает версию сервера 1, протокол 1.1, сборку 290

supportedClientVersions

Выводит список поддерживаемых версий клиента.

configureTrace

Включает или отключает трассировку.

-trace

- ON: изменяет уровень записи в журнал на FINEST
- OFF: изменяет уровень записи в журнал на INFO

printTrace

Выводит текущее значение параметра трассировки.

-collectionName *имя_собрания*

Задаёт имя собрания для удаления или для получения информации о состоянии.

-configPath *абсолютный_путь_к_папке_конфигурации*

Задаёт абсолютный путь к папке конфигурации текстового поиска.

-locale *строка_локали_из_пяти_символов*

Задаёт строку локали из пяти символов для вывода сообщений об ошибках, возвращаемых при использовании инструмента. Если этот параметр не указан, используется локаль сервера.

help

Выводит справку об использовании команд инструмента управления. Эта информация об использовании выводится также, если запустить инструмент управления без обязательных аргументов.

Примеры

Чтобы удалить собрание:

```
adminTool delete -configPath /opt/ibm/search/config -collectionName  
db2inst1_SAMPLE_TS242320_0000
```

Чтобы вывести состояние всех собраний, использующих русскую локаль, введите следующую команду:

```
adminTool status -configPath c:\Programs\IBM\Search\config -locale ru_RU
```

Чтобы включить трассировку, введите следующую команду:

```
adminTool configureTrace -configPath /opt/ibm/search/config -trace on
```

Замечания по использованию

Журнал ошибок или трассировки по умолчанию записывается в `AdminTool.0.log` в каталоге журнала по умолчанию. Этот журнал можно сформатировать с помощью инструмента форматирования журналов.

Запустите инструмент управления из каталога `<домашний_каталог_ECMTS>\bin` на сервере текстового поиска ECM.

- В системах Linux или UNIX: **adminTool.sh**
- В системах Windows: **adminTool.bat**

Инструменты синонимов для DB2 Text Search

Инструменты синонимов - это инструмент командной строки, при помощи которого удобно добавлять словари синонимов в индексы текстового поиска и удалять их оттуда.

При использовании Инструментов синонимов нет необходимости останавливать службы экземпляра текстового поиска.

Синтаксис synonymTool

```
synonymTool [importSynonym-specs | removeSynonym --collectionName имя-собрания | help]
```

► `configPath` — абсолютный-путь-к-папке-конфигурирования —————>

importSynonym-specs:

|— `importSynonym` — `synonymFile` — абсолютный-путь-к-файлу-синонимов —————>

► `collectionName` — имя-собрания — `replace` —

true
false

 —————>

Параметры

importSynonym

Добавляет словарь синонимов, заданный `-synonymFile`, в собрание текстового поиска, заданное `-collectionName`.

`-synonymFile` *абсолютный-путь-к-файлу-синонимов*

Задаёт файл XML, который будет использован как словарь синонимов.

-replace

- true

Задаёт, что синонимы в файле, заданном `-synonymFile`, должны заменить текущие синонимы в существующем словаре синонимов, связанном с собранием текстового поиска.

- false

Задаёт, что синонимы в файле, заданном `-synonymFile`, должны быть добавлены к существующему словарю синонимов, связанному с собранием текстового поиска.

removeSynonym

Удаляет словарь синонимов, если он был связан с собранием текстового поиска, заданным `-collectionName`.

`-collectionName` *имя-собрания*

Задаёт собрание текстового поиска.

`-configPath` *абсолютный-путь-к-папке-конфигурации*

Задаёт абсолютный путь к папке конфигурации текстового поиска.

help

Выводит информацию об использовании команды Инструменты синонимов.

Информация об использовании выводится также, если запустить Инструменты синонимов без обязательного аргумента.

Использование

Файл XML, который будет использован как словарь синонимов, должен находиться на сервере текстового поиска.

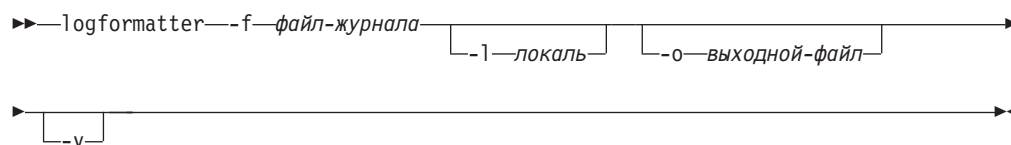
Журнал ошибок или трассировки записывается по умолчанию в файл `SynonymTool.0.log` в каталоге журналов по умолчанию.

Инструмент форматирования журналов для DB2 Text Search

DB2 Text Search записывает в журналы системные сообщения и сообщения трассировки, позволяющие определить источник возможных ошибок. Журналы сервера можно просмотреть и сохранить при помощи инструмента форматирования журнала.

Журналы сервера находятся в каталоге `INSTALL_HOME/log`. По умолчанию журнал трассировки отключен, а для системного журнала задан уровень SEVERE. Для журналов сервера выполняется ротация: когда объем файла журнала достигает 10 мегабайт, создается следующий файл журнала. В каждый момент времени в системе хранится не более пяти файлов журнала.

Синтаксис инструмента



Параметры

logformatter

Разрешает просмотр или сохранение журнала сервера.

-f *файл-журнала*

Задает файл журнала, который вы хотите отформатировать.

-l *локаль*

Задает пятисимвольную локаль для записи переформатированных сообщений журнала. Значение по умолчанию - en_US.

-o *выходной-файл*

Задает файл вывода, куда должны записываться переформатированные сообщения журнала с использованием кодировки UTF-8. Если эта опция не задана, используется стандартный файл вывода.

-v

Задает, что сообщения отладки будут записываться в подробном режиме.

-?

Выводит сообщение справки.

Приложение А. Сравнение DB2 Text Search и Net Search Extender

Хотя запросы, выполняемые при помощи DB2 Text Search и Net Search Extender, выглядят аналогичными, результаты этих запросов могут отличаться. Поэтому важно знать отличия между этими двумя механизмами поиска.

При помощи следующих таблиц, в которых сравниваются возможности и функции DB2 Text Search и Net Search Extender, можно выбрать подходящий для ваших потребностей инструмент поиска.

Таблица 19. Администрирование и среды

Возможность или функция	DB2 Text Search	Модуль DB2 Net Search Extender
Интерфейс командной строки	Поддерживается	Поддерживается
Хранимые процедуры для администрирования	Поддерживается	Не поддерживается
Внешнее хранение текстовых документов	Не поддерживается	Поддерживается
Среды многораздельных баз данных	Не поддерживается	Поддерживаются на всех платформах, кроме Linux PPC, Solaris для AMD64 и Microsoft Cluster server
Обработка ошибок	Коды SQL, специфичные для продукта коды	Специфичные для продукта коды
Лингвистическая обработка	На всех поддерживаемых языках	Только на английском языке

Таблица 20. Поиск

Возможность или функция	DB2 Text Search	Модуль DB2 Net Search Extender
XPath	Поддерживается	Не поддерживается
XQuery	Поддерживается	Поддерживается
Запросы SQL/XML	Поддерживается	Поддерживается
Запросы SQL	Поддерживается	Поддерживается
функция SCORE	Поддерживается	Поддерживается
Функции CONTAINS и xmlcolumn-contains	Поддерживается	Поддерживается
Скалярная функция NUMBEROFMATCHES	Не поддерживается	Поддерживается
Выделение	Не поддерживается	Поддерживается
Весовые коэффициенты или усиление	Поддерживается	Поддерживается
Аргумент RESULTLIMIT	Поддерживается	Поддерживается
Escape-символы	Поддерживается	Поддерживается
Необязательные термины	Поддерживается	Не поддерживается

Таблица 20. Поиск (продолжение)

Возможность или функция	DB2 Text Search	Модуль DB2 Net Search Extender
Поиск атрибутов	Не поддерживается	Поддерживается
Поиск по полю	Для документов XML с использованием выражений поиска XML	Для документов XML с определением поля при помощи модели документа
Контекстный поиск	Не поддерживается	Поддерживается
Поиск неполных соответствий	Не поддерживается	Поддерживается
Поиск с символами подстановки	Поддерживается	Поддерживается
Взаимосвязи в тезаурусе (иерархические, ассоциативные и пользовательские)	Не поддерживается	Поддерживается
Словари синонимов	Поддерживается	Поддерживается

Таблица 21. Индексация и данные

Возможность или функция	DB2 Text Search	Модуль DB2 Net Search Extender
Реорганизация индекса	Поддерживается косвенная реорганизация	Поддерживается и косвенная, и явная реорганизация
Изменения индексов для триггеров	Поддерживается	Поддерживается
Изменения индексов с помощью репликации	Не поддерживается	Поддерживается
Пересоздание индексов при изменении	Не поддерживается	Поддерживается
Создание индексов по столбцам производных таблиц	Не поддерживается	Поддерживается
Создание индексов по псевдонимам в базах данных объединения	Не поддерживается	Поддерживается
Несколько индексов на столбец	Не поддерживается	Поддерживается
Предварительно отсортированные индексы	Не поддерживается	Поддерживается
Кэширование информации индексов (подмножество индексов сохраняется в памяти для ускорения получения частых запросов)	Не поддерживается	Поддерживается
Потоковая передача результатов	Поддерживается	Не поддерживается
Пользовательские функции преобразований	Поддерживается	Поддерживается
Использование файлов моделей для индексации документов XML	Не поддерживается	Поддерживается

Таблица 21. Индексация и данные (продолжение)

Возможность или функция	DB2 Text Search	Модуль DB2 Net Search Extender
Пространства имен XML	Прямо не поддерживаются (нужно использовать обходные приемы)	Ограниченная поддержка
Обработка стоп-слов	Не поддерживается	Поддерживается
Форматы документов	Text, HTML, XML, собственные форматы (с использованием INSO)	Text, HTML, XML, General Purpose Format (GPP), собственные форматы (с использованием INSO)

Примечание: Алгоритмы оценки, используемые DB2 Text Search и Net Search Extender, отличаются, поэтому при использовании функции SCORE результаты могут быть разными. Хотя результаты оценки согласованы среди других документов в результатах одного поиска, значения оценки для конкретного документа могут быть разными в DB2 Text Search и Net Search Extender, поэтому используемые в запросе пределы оценки могут требовать уточнения.

Перенастройка в DB2 Text Search

Начиная с DB2 Версии 9.7 Fix Pack 3 и более новых пакетов исправлений, текстовые индексы DB2 Text Search и Net Search Extender могут одновременно существовать в одной базе данных. Теперь текстовый индекс DB2 Text Search можно создать для того же столбца, для которого уже существует индекс Net Search Extender, однако только один текстовый индекс может быть активен для одного столбца, и все текстовые индексы в одном запросе должны быть одинакового типа.

Команды администрирования можно выполнять и для активных, и для неактивных текстовых индексов. Это позволяет заполнять текстовый индекс DB2 Text Search для столбца в то время, когда существующий индекс Net Search Extender еще активно используется.

Используемый для DB2 Text Search язык запросов синтаксически и семантически отличается от DB2 Net Search Extender. Когда новый индекс DB2 Text Search готов для использования и активация переключена с индекса Net Search Extender на новый индекс, нужно соответственно переключить прикладную программу.

Рекомендуется поддерживать Net Search Extender неактивным, пока вы не будете удовлетворены качеством поиска с новым индексом. Если все работает нормально, неактивный индекс Net Search Extender можно отбросить, в противном случае его можно повторно активировать одновременно с переключением на связанную прикладную программу.

Для перенастройки вашей базы данных следуйте указаниям ниже:

1. Запустите DB2 Text Search.
2. Включите базу данных Net Search Extender для DB2 Text Search.
3. Создайте и измените индекс DB2 Text Search для столбца, у которого есть индекс Net Search Extender.
4. Активируйте новый индекс DB2 Text Search при помощи команды ALTER INDEX с опцией SET ACTIVE и переключите все связанные прикладные программы.
5. Убедитесь, что индекс DB2 Text Search работает правильно. Создайте резервные копии и затем удалите индексы Net Search Extender.

Если в вашей системе возможность совместного существования индексов недоступна, создайте тестовую базу данных с данными о продукции и используйте ее для изучения DB2 Text Search и различий с Net Search Extender. Для переключения способа текстового поиска в базе данных продукции надо выключить Net Search Extender, включить DB2 Text Search, создать и заполнить индексы DB2 Text Search.

Приложение В. Локали, поддерживаемые DB2 Text Search

В следующей таблице перечислены локали, поддерживаемые DB2 Text Search для обработки документов.

Таблица 22. Поддерживаемые локали

Код локали	Язык	Регион
ar_AA	Арабский	Арабские страны или регионы
cs_CZ	Чешский	Чехия
da_DK	Датский	Дания
de_CH	Немецкий	Швейцария
de_DE	Немецкий	Германия
el_GR	Греческий	Греция
en_AU	Английский	Австралия
en_GB	Английский	Великобритания
en_US	Английский	Соединенные Штаты
es_ES	Испанский	Испания
fi_FI	Финский	Финляндия
fr_CA	Французский	Канада
fr_FR	Французский	Франция
it_IT	Итальянский	Италия
ja_JP	Японский	Япония
ko_KR	корейский	Республика Корея
nb_NO	Норвежский (букмол)	Норвегия
nl_NL	Голландский	Нидерланды
nn_NO	Норвежский (нюнорск)	Норвегия
pl_PL	Польский	Польша
pt_BR	Португальский	Бразилия
pt_PT	Португальский	Португалия
ru_RU	Русский	Россия
sv_SE	Шведский	Швеция
zh_CN	китайский	Китай
zh_TW	китайский	Тайвань

Приложение С. Команды DB2

db2iupgrade - обновление экземпляра

Обновляет экземпляр до копии DB2 текущего выпуска от копии DB2 предыдущего выпуска. Копия DB2, из которой вы запускаете команду **db2iupgrade**, должна поддерживать обновление экземпляра от копии DB2, которую вы хотите обновить.

В операционных системах Linux и UNIX эта команда расположена в каталоге *DB2DIR/instance*, где *DB2DIR* - это положение установки, где установлен новый выпуск системы баз данных DB2. Эта команда не поддерживает обновления экземпляра для установки без полномочий root.

В операционных системах Windows эта команда расположена в каталоге **DB2PATH\bin**, где **DB2PATH** - это положение, где установлена копия DB2. Чтобы переместить профиль вашего экземпляра из его текущего положения в другое, используйте опцию **/p** и задайте путь к профилю экземпляра. В противном случае профиль экземпляра останется после обновления в своем исходном положении.

Авторизация

Полномочия пользователя root в операционных системах Linux и UNIX или локального администратора в операционных системах Windows.

Синтаксис команды

Для операционных систем Linux и UNIX:

```
db2iupgrade [-d] [-k] [-a:тип_аутентификации] [-u:ID_изолированного_пользователя] имя_экземпляра
```

Для операционных систем Windows:

```
db2iupgrade имя_экземпляра /u:имя_пользователя,пароль
```

```
/p:путь-профиля-экземпляра /q /a:authType
```

```
/j-"TEXT_SEARCH" ,имя_службы ,номер_порта /?
```

Параметры команды

Для операционных систем Linux и UNIX:

- d Включает режим отладки. Используйте эту опцию только после получения инструкций от службы поддержки баз данных DB2.
- k Сохраняет прежний тип экземпляра, если он поддерживается в копии DB2, из которой вы запускаете команду **db2iupgrade**. Если этот параметр не задан, тип экземпляра изменяется на поддерживаемый тип экземпляра по умолчанию.

-a тип_аутентификации

Задает тип аутентификации (SERVER, CLIENT или SERVER_ENCRYPT) для этого экземпляра. Значение по умолчанию - SERVER.

-u ID_изолированного_пользователя

Задает имя ID пользователя, применяемое для запуска изолированных пользовательских функций и изолированных хранимых процедур. Эта опция требуется при обновлении экземпляра клиента DB2 до экземпляра сервера DB2.

имя_экземпляра

Задает имя экземпляра.

Для операционных систем Windows:

имя_экземпляра

Задает имя экземпляра.

/u:имя-пользователя,пароль

Задает имя учетной записи и пароль для службы DB2. Эта опция обязательна при обновлении многораздельного экземпляра.

/p:путь-профиля-экземпляра

Задает новый путь профиля экземпляра для обновляемого экземпляра.

/q

Запускает команду **db2iupgrade** в режиме без вывода сообщений.

/a:authType

Задает тип аутентификации (SERVER, CLIENT или SERVER_ENCRYPT) для этого экземпляра.

/j "TEXT_SEARCH"

Конфигурирует сервер DB2 Text Search с использованием сгенерированных значений по умолчанию для имени службы и номера порта TCP/IP. Этот параметр нельзя использовать, если тип экземпляра - клиент.

/j "TEXT_SEARCH, имя_службы"

Конфигурирует сервер DB2 Text Search с использованием заданного имени службы и автоматически сгенерированного номера порта. Если для заданного имени службы в файле `services` задан номер порта, используется этот номер порта.

/j "TEXT_SEARCH, имя_службы, номер_порта"

Конфигурирует сервер DB2 Text Search с использованием заданного имени службы и номера порта.

/j "TEXT_SEARCH, номер_порта"

Конфигурирует сервер DB2 Text Search с использованием имени службы по умолчанию и заданного номера порта. Допустимые номера порта должны быть в диапазоне 1024 - 65535.

/?

Выводит информацию об использовании для команды **db2iupgrade**.

Замечания по использованию

Команда **db2iupgrade** вызывает команду **db2ckupgrade** с параметром **-not1** и задает файл `update.log` как файл журнала для **db2ckupgrade**. Создаваемый для **db2iupgrade** файл журнала по умолчанию - это `/tmp/db2ckupgrade.log.processID`. Перед обновлением экземпляра проверьте, готовы ли локальные базы данных для обновления. Параметр **-not1** отключает проверку наличия индексов типа 1. Файл

журнала создается в домашнем каталоге экземпляра для операционных систем Linux и UNIX или в текущем каталоге для операционных систем Windows. Если команда **db2ckupgrade** вернет какую-либо ошибку, обновление экземпляра будет прервано.

Для сред многораздельных баз данных до запуска команды **db2iupgrade** необходимо во всех разделах базы данных запустить команду **db2ckupgrade**. Команда **db2ckupgrade** возвращает ошибки только для того раздела базы данных, где была запущена команда **db2iupgrade**. Если вы не проверите готовность всех разделов базы данных для обновления, последующие обновления баз данных могут закончиться неудачно даже при успешном обновлении экземпляра. Подробности смотрите в описании команды **db2ckupgrade**.

Команду **db2iupgrade** надо запустить однократно с полномочиями root на узле - владельце экземпляра.

Для операционных систем Linux и UNIX:

- Если вы используете команду **db2iupgrade** для обновления экземпляра DB2 от предыдущей версии до текущей версии системы баз данных DB2, глобальные переменные профиля DB2, определенные в старом каталоге установки базы данных DB2, не будут обновлены в новое положение установки. Переменные профиля экземпляра DB2, специфичные для экземпляра, который будет обновляться, будут перенесены в нужное положение после завершения обновления экземпляра.
- Если для получения полномочий root вы используете команду **su** вместо команды **login**, надо ввести команду **su** с опцией **-**, которая указывает задание такой же среды процесса, что и при регистрации в системе при помощи команды **login**.
- Не используйте среду экземпляра DB2 как источник для пользователя root. Если среда экземпляра DB2 не поддерживается в качестве источника, запустите команду **db2iupgrade**.
- В операционных системах UNIX и Linux, если установлен IBM Tivoli Monitoring for Databases: Будет установлен агент DB2 и обновлен экземпляр DB2, а также создан агент мониторинга для экземпляра DB2, если выполнено следующее:
 - Тип экземпляра DB2 - standalone, wse или ese.
 - Версия экземпляра DB2 - 9.5 (или новее).

Кроме этого, продукт ITMA должен быть уже установлен для копии DB2, для которой вы проводите изменение экземпляра. ITMA расположен в каталоге *DB2DIR/i tma*, где *DB2DIR* - это каталог с установленным продуктом баз данных DB2.

- В AIX 6.1 (или в более новых версиях), когда эта команда выполняется из копии DB2 совместного использования в глобальной среде системных разделов рабочей нагрузки (workload partition, WPAR), эта команда должна быть запущена от имени пользователя root.

Для операционных систем Windows:

- В операционных системах Windows, если установлен IBM Tivoli Monitoring for Databases: Будет установлен агент DB2 и изменен экземпляр копии DB2, а также создан агент мониторинга для экземпляра DB2, если выполнено следующее:
 - Тип экземпляра DB2 - автономный, wse или ese.
 - У копии DB2 по умолчанию установлен компонент агента ITM.
 - Версия экземпляра DB2 - 9.5 (или новее).
 - Нет существующего продукта ITM for Databases.

Кроме этого, после создания агента мониторинга для экземпляра DB2 создается агент мониторинга для файлов экземпляра DB2, служба Windows и записи реестра.

db2icrt - Создание экземпляра

Создает экземпляры DB2.

В операционных системах Linux и UNIX эта утилита расположена в каталоге *DB2DIR/instance*, где *DB2DIR* - это положение установки, где установлена текущая версия системы баз данных DB2. В операционных системах Windows эта утилита расположена в каталоге *DB2PATH\bin*, где *DB2PATH* - это положение, где установлена копия DB2.

Команда **db2icrt** создает экземпляры DB2 в домашнем каталоге владельца экземпляра.

Примечание: Эта команда недоступна при установке продуктов баз данных DB2 без полномочий root в операционных системах Linux и UNIX.

Авторизация

Полномочия пользователя root в операционных системах Linux и UNIX или локального администратора в операционных системах Windows.

Синтаксис команды

Для операционных систем Linux и UNIX:

```
db2icrt [-h] [-d] [-a тип_аутентификации] [-p имя_порта]
        [-s тип_экземпляра] [-u ID_изолированного_пользователя]
        имя_экземпляра
```

Для операционных систем Windows

```
db2icrt имя_экземпляра [-s тип_экземпляра]
        [-u имя_пользователя, пароль] [-p InstProfPath] [-h имя_хоста]
        [-r диапазон_портов]
        [-j "TEXT_SEARCH , имя_службы , номер_порта"] [-?]
```

Параметры команды

Для операционных систем Linux и UNIX:

-h | -? Выводит информацию об использовании.

-d Включает режим отладки. Используйте эту опцию только после получения инструкций от службы поддержки баз данных DB2.

-a *тип_аутентификации*

Задает тип аутентификации (SERVER, CLIENT или SERVER_ENCRYPT) для этого экземпляра. Значение по умолчанию - SERVER.

-p *имя_порта*

Задает имя или номер порта, используемого этим экземпляром. Эта опция неприменима к экземплярам клиентов.

-s *тип_экземпляра*

Задает тип создаваемого экземпляра. Используйте опцию **-s** только при создании экземпляра, отличного от экземпляра по умолчанию, связанного с установленным продуктом, из которого запускается команда **db2icrt**.

Допустимые значения:

client Используется для создания экземпляра клиента. Это тип экземпляра по умолчанию для IBM Data Server, IBM Data Server и DB2 Connect Personal Edition.

standalone

Используется для создания экземпляра сервера баз данных с локальными клиентами. Это тип экземпляра по умолчанию для DB2 Personal Edition.

ese Используется для создания экземпляра сервера баз данных с локальными и удаленными клиентами и поддержкой среды многораздельных баз данных. Это тип экземпляра по умолчанию для DB2 Enterprise Server Edition или DB2 Advanced Enterprise Server Edition.

wse Используется для создания экземпляра сервера базы данных с локальными и удаленными клиентами. Это тип экземпляра по умолчанию для DB2 Workgroup Server Edition, DB2 Express Edition или DB2 Express-C и DB2 Connect Enterprise Edition.

Продукты DB2 поддерживают свои экземпляры по умолчанию и более младшие типы экземпляров. Для экземпляра DB2 Enterprise Server Edition поддерживает типы экземпляров **ese**, **wse**, **standalone** и **client**.

-u *ID_изолир_пользователя*

Задает имя ID пользователя, применяемое для запуска изолированных пользовательских функций и изолированных хранимых процедур. Опция **-u** требуется, если вы не создаете экземпляр клиента.

имя_экземпляра

Задает имя экземпляра, совпадающее с именем существующего пользователя в операционной системе. Это значение должно быть последним аргументом команды **db2icrt**.

Для операционных систем Windows

имя_экземпляра

Задает имя экземпляра.

-s *InstType*

Задает тип создаваемого экземпляра. В настоящее время существует четыре типа экземпляров DB2. Допустимые значения:

client Используется для создания экземпляра клиента. Это тип экземпляра по умолчанию для IBM Data Server, IBM Data Server и DB2 Connect Personal Edition.

standalone

Используется для создания экземпляра сервера баз данных с локальными клиентами. Это тип экземпляра по умолчанию для DB2 Personal Edition.

ese Используется для создания экземпляра сервера баз данных с локальными и удаленными клиентами и поддержкой среды многораздельных баз данных.

-s ese -u имя_пользователя, пароль

Опции надо использоваться с командой **db2icrt** для создания типа экземпляра ESE и экземпляра среды многораздельных баз данных.

wse Используется для создания экземпляра сервера базы данных с локальными и удаленными клиентами. Это тип экземпляра по умолчанию для DB2 Workgroup Server Edition, DB2 Express Edition или DB2 Express-C и DB2 Connect Enterprise Edition.

Продукты DB2 поддерживают свои экземпляры по умолчанию и более младшие типы экземпляров. Для экземпляра DB2 Enterprise Server Edition поддерживает типы экземпляров *ese, wse, standalone* и *client*.

-u *имя_пользователя, пароль*

Задаёт имя учётной записи и пароль для службы DB2. Эта опция обязательна при создании экземпляра распределённой базы данных.

-p *InstProfPath*

Задаёт путь профиля экземпляра.

-h *HostName*

Переопределяет имя хоста TCP/IP по умолчанию, если для текущего компьютера есть несколько имен хостов. Это имя хоста TCP/IP используется при создании раздела базы данных по умолчанию (раздел базы данных 0). Эта опция действительна только для экземпляров многораздельных баз данных.

-r *PortRange*

Задаёт диапазон портов TCP/IP, которые используются экземпляром распределённой базы данных при работе в режиме MPP. Например, *-r 50000,50007*. Если эта опция задана, в файле *services* на локальном компьютере будут обновлены следующие записи:

DB2_InstName	начальный_порт/tcp
DB2_InstName_END	конечный порт/tcp

/j **"TEXT_SEARCH"**

Конфигурирует сервер DB2 Text Search с использованием сгенерированных значений по умолчанию для имени службы и номера порта TCP/IP. Этот параметр нельзя использовать, если тип экземпляра - клиент.

/j **"TEXT_SEARCH, имя_службы"**

Конфигурирует сервер DB2 Text Search с использованием заданного имени службы и автоматически сгенерированного номера порта. Если для заданного имени службы в файле *services* задан номер порта, используется этот номер порта.

/j **"TEXT_SEARCH, имя_службы, номер_порта"**

Конфигурирует сервер DB2 Text Search с использованием заданного имени службы и номера порта.

`/j "TEXT_SEARCH, номер_порта"`

Конфигурирует сервер DB2 Text Search с использованием имени службы по умолчанию и заданного номера порта. Допустимые номера порта должны быть в диапазоне 1024 - 65535.

-? Выводит информацию об использовании.

Пример

- На компьютере AIX для создания экземпляра для ID пользователя `db2inst1` введите следующую команду:

На компьютере клиента:

```
DB2DIR/instance/db2icrt db2inst1
```

На компьютере сервера:

```
DB2DIR/instance/db2icrt -u db2fenc1 db2inst1
```

где `db2fenc1` - ID пользователя, от имени которого будут запускаться изолированные пользовательские функции и изолированные хранимые процедуры.

Замечания по использованию

- Папка *домашний_каталог_экземпляра*/`sqllib/db2tss/config` создается командой **db2icrt** в операционных системах Linux и UNIX. Рекомендуется использовать символическую ссылку на область вне каталога `sqllib`.
- Опция **-s** предназначена для ситуаций, когда вы хотите создать экземпляр, не использующий все функции системы. Например, если вы используете DB2 Enterprise Server Edition (ESE) в операционной системе UNIX, но многообразные возможности не требуются, можно создать экземпляр DB2 Workgroup Server Edition (WSE) при помощи опции **-s WSE**.
- Для создания экземпляра DB2, который поддерживает Microsoft Cluster Server, сначала создайте экземпляр, а затем используйте команду **db2mcs** для его перенастройки и работы в MSCS.
- В операционных системах Linux и UNIX под именем пользователя может быть создан только один экземпляр. Если вы хотите создать экземпляр под именем пользователя, у которого уже есть экземпляр, перед созданием нового экземпляра нужно отбросить существующий.
- При создании экземпляров DB2 учитывайте следующие ограничения:
 - Если для создания экземпляров DB2 используются существующие ID, убедитесь, что эти ID не заблокированы и сроки их паролей не истекли.
- Вы можете также использовать команду **db2isetup** для создания и изменения экземпляров DB2 при помощи графического интерфейса во всех поддерживаемых операционных системах Linux и UNIX.
- В операционных системах Linux и UNIX, если получаете полномочия root при помощи команды **su** вместо команды **login**, вводите команду **su** с опцией **-**, которая задает ту же среду процесса, как и при регистрации в системе при помощи команды **login**.
- В операционных системах Linux и UNIX не используйте среду экземпляра DB2 как источник для пользователя root. Выполнение команды **db2icrt**, когда среда экземпляра DB2 используется как источник, не поддерживается.

- В AIX 6.1 (или в более новых версиях), когда эта команда выполняется из копии DB2 совместного использования в глобальной среде системных разделов рабочей нагрузки (workload partition, WPAR), эта команда должна быть запущена от имени пользователя root.
- В операционных системах Windows, если установлен IBM Tivoli Monitoring for Databases: Будет установлен агент DB2 и создан экземпляр DB2, а также создан агент мониторинга для экземпляра DB2, если выполнено следующее:
 - Тип экземпляра DB2 - standalone, wse или ese.
 - У копии DB2 по умолчанию установлен компонент агента ITM.
 - Версия экземпляра DB2 - 9.5 (или новее).
 - Нет существующего продукта ITM for Databases.

Кроме этого, после создания агента мониторинга для экземпляра DB2 создается агент мониторинга для файлов экземпляра DB2, служба Windows и записи реестра.

db2idrop - удаление экземпляра

Удаляет экземпляр DB2, созданный командой **db2icrt**.

Вы можете отбросить только те экземпляры, которые выводятся в списке по команде **db2ilist** для той копии DB2, из которой вы запускаете команду **db2idrop**.

В операционных системах Linux и UNIX эта утилита расположена в каталоге *DB2DIR/instance*, где *DB2DIR* - это положение установки, где установлена текущая версия системы баз данных DB2. В операционных системах Windows эта утилита расположена в каталоге *DB2PATH\bin*, где *DB2PATH* - это положение, где установлена копия DB2.

Примечание: С помощью этой команды нельзя отбросить экземпляр DB2 установки без полномочий root в операционных системах Linux и UNIX. Единственная возможность - деинсталлировать копию DB2, установленную без полномочий root. Подробности смотрите ниже в публикации *Замечания об использовании*.

Авторизация

Полномочия пользователя root в операционных системах Linux и UNIX или локального администратора в операционных системах Windows.

Синтаксис команды

Для операционных систем Linux и UNIX:

```

▶▶ db2idrop [-d] [-f] [-h] [-?] имя_экземпляра
  
```

Для операционных систем Windows

```

▶▶ db2idrop [-f] [-h] имя_экземпляра
  
```

Параметры команды

Для операционных систем Linux и UNIX:

имя_экземпляра

Задает имя экземпляра.

-d Вводит режим отладки для использования службой поддержки баз данных DB2.

-f Этот параметр устарел.

Задает флаг принудительного завершения. Если этот флаг задан, все программы, использующие этот экземпляр, будут принудительно завершены.

-h | -? Выводит информацию об использовании.

Для операционных систем Windows

имя_экземпляра

Задает имя экземпляра.

-f Задает флаг принудительного завершения. Если этот флаг задан, все программы, использующие этот экземпляр, будут принудительно завершены.

-h Выводит информацию об использовании.

Пример

Если вы создали `db2inst1` в операционных системах Linux или UNIX, запустив следующую команду:

```
/opt/IBM/db2/copy1/instance/db2icrt -u db2fenc1 db2inst1
```

Чтобы отбросить `db2inst1`, надо ввести следующую команду:

```
/opt/IBM/db2/copy1/instance/db2idrop db2inst1
```

Замечания по использованию

- Перед отбрасыванием экземпляра убедитесь, что менеджер баз данных DB2 остановлен, а обращающиеся к базе данных DB2 прикладные программы отсоединены и прерваны. Должно быть выполнено резервное копирование связанных с экземпляром баз данных DB2, а данные о конфигурации сохранены для возможного использования в будущем.
- Команда **db2idrop** не удаляет базы данных. Сначала удалите базы данных, если они больше не требуются. Если базы данных не удалены, их всегда можно каталогизировать под другой копией DB2 того же выпуска и продолжать их использование.
- Если вы хотите сохранить конфигурации DB2 Text Search и планируете повторно использовать базы данных экземпляра, необходимо предпринять некоторые дополнительные действия по сохранению каталога `config` (в UNIX: *домашний_каталог_экземпляра/sqllib/db2tss/config*, в Windows: *путь_к_профилю_экземпляра\имя_экземпляра\db2tss\config*) или содержимого каталога `config` перед запуском команды **db2idrop**. После создания нового экземпляра каталог `config` можно восстановить. Однако восстановление каталога `config` применимо только в том случае, если новый экземпляр создан в том же выпуске и на том же уровне пакета Fix Pack.
- Экземпляр установки без полномочий `root` нельзя отбросить в операционных системах Linux и UNIX. Единственная доступная для пользователя возможность удалить такой экземпляр DB2 - деинсталлировать копию DB2, установленную без полномочий `root`, введя команду **db2_deinstall -a**.

- В операционных системах Linux и UNIX, если получаете полномочия root при помощи команды **su** вместо команды **login**, вводите команду **su** с опцией **-**, которая задает ту же среду процесса, как и при регистрации в системе при помощи команды **login**.
- В операционных системах Linux и UNIX не используйте среду экземпляра DB2 как источник для пользователя root. Выполнение команды **db2idrop**, когда среда экземпляра DB2 используется как источник, не поддерживается.
- В AIX 6.1 (или в более новых версиях), когда эта команда выполняется из копии DB2 совместного использования в глобальной среде системных разделов рабочей нагрузки (workload partition, WPAR), эта команда должна быть запущена от имени пользователя root.
- В операционных системах Windows, если установлен IBM Tivoli Monitoring for Databases: Будет установлен агент DB2 и отброшен экземпляр DB2, а также остановлен агент мониторинга для DB2. Будут также удалены все файлы, службы и записи реестра, созданные для агента мониторинга для DB2.
- В операционных системах Windows, если экземпляр включен в кластер при помощи Microsoft Cluster Service (MSCS), перед отбрасыванием экземпляра его можно исключить из кластера, введя команду **db2mcs** или **db2iclus**.

db2iupdt - Обновление экземпляров

Обновляет экземпляр, чтобы запускать копию DB2 с установленным новым продуктом баз данных или опцией DB2 с той же версией, что и копия DB2, связанная с данным экземпляром, или чтобы обновить тип экземпляра до типа экземпляра более высокого уровня.

Команда **db2iupdt** применима к экземплярам одной и той же версии, связанным с одной или разными копиями DB2. Во всех случаях команда **db2iupdt** обновит экземпляр, то есть будет работать с кодом в той копии DB2, в которой запущена. Эту команду нужно запустить, если:

- Вы устанавливаете пакет Fix Pack, и автоматическое обновление экземпляра завершается неудачно.
- Вы устанавливаете новый продукт или возможность баз данных DB2 в копию DB2, связанную с экземпляром DB2.
- Вы хотите обновить экземпляр DB2 от одной копии DB2 до другой копии DB2 той же версии продукта баз данных DB2.

После установки пакета Fix Pack в операционных системах Linux и UNIX команда **db2iupdt** выполняется автоматически.

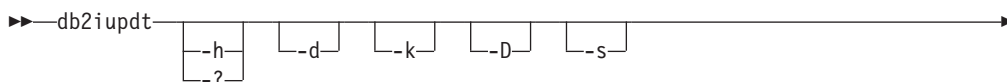
Для изменения экземпляра с помощью команды **db2iupdt** сначала нужно остановить все процессы, выполняющиеся для этого экземпляра.

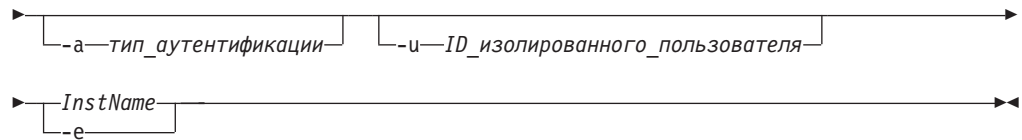
Авторизация

Полномочия пользователя root в операционных системах UNIX и Linux или локального администратора в операционных системах Windows.

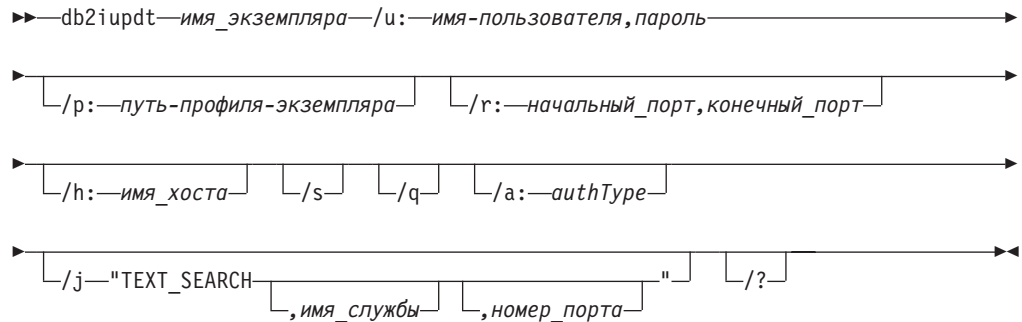
Синтаксис команды

Для операционных систем UNIX и Linux





Для операционных систем Windows:



Параметры команды

Для операционных систем UNIX и Linux

- h | -? Выводит информацию об использовании.
- d Включает режим отладки.
- k Сохраняет текущий тип экземпляра при обновлении.
- D Переносит экземпляр из кода высшего уровня, установленного по одному пути, в код более низкого уровня, установленный по другому пути.
- s Игнорирует существующий каталог журналов SPM.
- a *тип_аутентификации*
Задает тип аутентификации (SERVER, SERVER_ENCRYPT или CLIENT) для этого экземпляра. Значение по умолчанию - SERVER.
- u *Fenced ID*
Задает имя ID пользователя, применяемое для запуска изолированных пользовательских функций и изолированных хранимых процедур. Это единственная опция, необходимая при преобразовании типа экземпляра client в отличный от client тип. Для определения текущего типа экземпляра посмотрите параметр типа узла в выводе команды **GET DBM CFG**. Если тип экземпляра уже отличен от client или тип client остается типом экземпляра (например, при указании опции **-k**), опция **-u** не нужна. Опция **-u** может изменить изолированного пользователя для существующего экземпляра.
- имя_экземпляра*
Задает имя экземпляра.
- e Обновляет каждый экземпляр.

Для операционных систем Windows:

- имя_экземпляра*
Задает имя экземпляра.
- /u: *имя-пользователя,пароль*
Задает имя учетной записи и пароль для службы DB2.

/p:путь-профиля-экземпляра

Задает новый путь профиля экземпляра для обновляемого экземпляра.

/r:начальный_порт,конечный_порт

Задает диапазон портов TCP/IP, которые используются экземпляром распределенной базы данных при работе в режиме MPP. Когда задана эта опция, файл служб на локальном компьютере будет дополнен следующими записями:

```
DB2_InstName      начальный_порт/tcp
DB2_InstName_END  конечный_порт/tcp
```

/h:имя_хоста

Переопределяет имя хоста TCP/IP по умолчанию, если для текущего компьютера есть несколько имен хостов TCP/IP.

/s Изменяет экземпляр в многораздельный экземпляр.

/q Запускает команду **db2iupdt** в режиме без вывода сообщений.

/a:authType

Задает *authType* - тип аутентификации (SERVER, CLIENT или SERVER_ENCRYPT) для этого экземпляра.

/j "TEXT_SEARCH"

Конфигурирует сервер DB2 Text Search с использованием сгенерированных значений по умолчанию для имени службы и номера порта TCP/IP. Этот параметр нельзя использовать, если тип экземпляра - клиент.

/j "TEXT_SEARCH, имя_службы"

Конфигурирует сервер DB2 Text Search с использованием заданного имени службы и автоматически сгенерированного номера порта. Если для заданного имени службы в файле *services* задан номер порта, используется этот номер порта.

/j "TEXT_SEARCH, имя_службы, номер_порта"

Конфигурирует сервер DB2 Text Search с использованием заданного имени службы и номера порта.

/j "TEXT_SEARCH, номер_порта"

Конфигурирует сервер DB2 Text Search с использованием имени службы по умолчанию и заданного номера порта. Допустимые номера порта должны быть в диапазоне 1024 - 65535.

/? Выводит информацию об использовании для команды **db2iupdt**.

Примеры (UNIX и Linux)

- Экземпляр `db2inst2` связан с копией DB2 продукта баз данных DB2, установленного в каталоге `DB2DIR1`. У вас есть другая копия DB2 продукта баз данных DB2 на том же компьютере в каталоге `DB2DIR2` для той же версии продукта баз данных DB2, что и установленная в каталоге `DB2DIR1`. Для изменения экземпляра, чтобы он запускался не из копии DB2, установленной в каталоге `DB2DIR1`, а из копии DB2, установленной в каталоге `DB2DIR2`, введите следующую команду:

```
DB2DIR2/instance/db2iupdt db2inst2
```

Если копия DB2, установленная в каталоге `DB2DIR2`, более старой версии по сравнению с копией DB2, установленной в каталоге `DB2DIR1`, введите следующую команду:

```
DB2DIR2/instance/db2iupdt -D db2inst2
```


Замечания по использованию

Для всех поддерживаемых операционных систем

- Если вы используете команду **db2iupdt** для изменения экземпляра DB2 от одной копии DB2 до другой копии DB2 той же версии продукта баз данных DB2, глобальные переменные профиля DB2, определенные в старом каталоге установки базы данных DB2, не будут обновлены в новое положение установки. Переменные профиля экземпляра DB2, специфичные для экземпляра, будут перенесены в нужное положение после завершения изменения экземпляра.

Для операционных систем UNIX и Linux

- Команда **db2iupdt** расположена в каталоге *DB2DIR/instance*, где *DB2DIR* - это положение, где установлена текущая версия продукта баз данных DB2.
- Если вы хотите изменить экземпляр установки без полномочий root, посмотрите команду изменения таких экземпляров **db2nrupdt**. Команда **db2iupdt** не поддерживает изменение экземпляров установки без полномочий root.
- Если для получения полномочий root вы используете команду **su** вместо команды **login**, надо ввести команду **su** с опцией **-**, которая указывает задание такой же среды процесса, что и при регистрации в системе при помощи команды **login**.
- Не используйте среду экземпляра DB2 как источник для пользователя root. Выполнение команды **db2iupdt**, когда среда экземпляра DB2 используется как источник, не поддерживается.
- В операционных системах UNIX и Linux, если установлен IBM Tivoli Monitoring for Databases: Будет установлен DB2 и изменен экземпляр DB2, а также создан агент мониторинга для экземпляра DB2, если выполнено следующее:
 - Тип экземпляра DB2 - автономный, wse или ese.
 - Версия экземпляра DB2 - 9.5 (или новее).

Кроме этого, продукт ITMA должен быть уже установлен для копии DB2, для которой вы проводите изменение экземпляра. ITMA расположен в каталоге *DB2DIR/itma*, где *DB2DIR* - это каталог с установленным продуктом баз данных DB2.

- В AIX 6.1 (или в более новых версиях), когда эта команда выполняется из копии DB2 совместного использования в глобальной среде системных разделов рабочей нагрузки (workload partition, WPAR), эта команда должна быть запущена от имени пользователя root.

Для операционных систем Windows

- Команда **db2iupdt** расположена в каталоге *DB2PATH\bin*, где *DB2PATH* - это положение, где установлена текущая версия продукта баз данных DB2.
- Экземпляр обновляется до копии DB2, из которой вы запускаете команду **db2iupdt**. Но чтобы переместить профиль вашего экземпляра из его текущего положения в другое, используйте опцию **/p** и задайте путь к профилю экземпляра. В противном случае профиль экземпляра останется после изменения в своем исходном положении. Вместо обновления от предыдущего выпуска до текущего используйте команду **db2iupgrade**.
- В операционных системах Windows, если установлен IBM Tivoli Monitoring for Databases: Будет установлен агент DB2 и изменен экземпляр копии DB2, а также создан агент мониторинга для экземпляра DB2, если выполнено следующее:
 - Тип экземпляра DB2 - автономный, wse или ese.

- У копии DB2 по умолчанию установлен компонент агента ITM.
- Версия экземпляра DB2 - 9.5 (или новее).
- Нет существующего продукта ITM for Databases.

Кроме этого, после создания агента мониторинга для экземпляра DB2 создается агент мониторинга для файлов экземпляра DB2, служба Windows и записи реестра.

Приложение D. Обзор технической информации DB2

Для получения технической информации по DB2 служат следующие инструменты и методы:

- Информационный центр DB2
 - Темы (Задачи, принципы работы и справочные темы)
 - Справка по инструментам DB2
 - Примеры программ
 - Обучающие программы
- Книги по DB2
 - Файлы PDF (загружаемые)
 - Файлы PDF (из DB2 PDF DVD)
 - Печатные книги
- Справка командной строки
 - Справка по командам
 - Справка по сообщениям

Примечание: Темы Информационного центра DB2 обновляются чаще, чем книги в формате PDF или бумажные книги. Чтобы получить наиболее свежую информацию по DB2, устанавливайте обновления сразу после их появления или используйте Информационный центр DB2 Information Center на сайте ibm.com.

Дополнительную техническую информацию о DB2, такую как технические замечания (technotes), оригинальные публикации (white papers) и технические руководства (IBM Redbooks) можно найти на сайте ibm.com. Обратитесь к библиотеке на сайте DB2 Information Management по адресу <http://www.ibm.com/software/data/sw-library/>.

Отзывы о документации

Мы ценим ваши отзывы о документации по DB2. Если у вас есть предложения по совершенствованию документации по DB2, присылайте электронные сообщения на адрес db2docs@ca.ibm.com. Коллектив разработчиков документации DB2 читает все ваши отзывы, но не может отвечать напрямую. Чтобы мы могли лучше понять ваши трудности, по возможности представьте конкретные примеры. Если ваш отзыв касается конкретной темы или файла справки, приложите заголовок справки и URL.

Не обращайтесь по этому адресу с вопросами, требующими службы поддержки заказчиков DB2. Если у вас есть технический вопрос по DB2, который не удастся разрешить по документации, обратитесь за помощью в ваш местный центр обслуживания IBM.

Техническая библиотека DB2 в печатном виде или в формате PDF

Ниже в таблице описан состав библиотеки DB2, доступной в Центре публикаций IBM по адресу www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss. Английские версии руководств Версия 9.7 в формате PDF можно скачать с сайта www.ibm.com/support/docview.wss?uid=swg27015148, а переведенные версии руководств DB2 в формате PDF - с сайта www.ibm.com/support/docview.wss?uid=swg27015149.

В таблицах описаны книги, доступные в печатном виде, тем не менее, в вашей стране или регионе они могут не быть доступны в печатном виде.

Номер формы увеличивается при каждом обновлении руководства. Убедитесь, что вы читаете самую свежую версию документации, как указано ниже.

Примечание: Информационный центр DB2 обновляется чаще, чем книги в формате PDF или бумажные книги.

Таблица 23. Техническая информация по DB2

Имя	Номер формы	Доступна в печатном виде	Последнее изменение
<i>Administrative API Reference</i>	SC27-2435-02	Да	Сентябрь, 2010
<i>Administrative Routines and Views</i>	SC27-2436-02	Нет	Сентябрь, 2010
<i>Call Level Interface Guide and Reference, Volume 1</i>	SC27-2437-02	Да	Сентябрь, 2010
<i>Call Level Interface Guide and Reference, Volume 2</i>	SC27-2438-02	Да	Сентябрь, 2010
<i>Command Reference</i>	SC27-2439-02	Да	Сентябрь, 2010
<i>Data Movement Utilities Guide and Reference</i>	SC27-2440-00	Да	Август 2009
<i>Data Recovery and High Availability Guide and Reference</i>	SC27-2441-02	Да	Сентябрь, 2010
<i>Database Administration Concepts and Configuration Reference</i>	SC27-2442-02	Да	Сентябрь, 2010
<i>Database Monitoring Guide and Reference</i>	SC27-2458-02	Да	Сентябрь, 2010
<i>Database Security Guide</i>	SC27-2443-01	Да	Ноябрь 2009
<i>Руководство по DB2 Text Search</i>	SC89-1250-03	Да	Сентябрь, 2010
<i>Developing ADO.NET and OLE DB Applications</i>	SC27-2444-01	Да	Ноябрь 2009
<i>Developing Embedded SQL Applications</i>	SC27-2445-01	Да	Ноябрь 2009
<i>Developing Java Applications</i>	SC27-2446-02	Да	Сентябрь, 2010
<i>Developing Perl, PHP, Python, and Ruby on Rails Applications</i>	SC27-2447-01	Нет	Сентябрь, 2010
<i>Developing User-defined Routines (SQL and External)</i>	SC27-2448-01	Да	Ноябрь 2009
<i>Getting Started with Database Application Development</i>	GI11-9410-01	Да	Ноябрь 2009

Таблица 23. Техническая информация по DB2 (продолжение)

Имя	Номер формы	Доступна в печатном виде	Последнее изменение
<i>Начинаем работу по установке и управлению DB2 в Linux и Windows</i>	GI43-0094-00	Да	Август 2009
<i>Globalization Guide</i>	SC27-2449-00	Да	Август 2009
<i>Установка серверов DB2</i>	GC89-1142-02	Да	Сентябрь, 2010
<i>Установка клиентов IBM Data Server</i>	GC89-1143-01	Нет	Сентябрь, 2010
<i>Message Reference Volume 1</i>	SC27-2450-00	Нет	Август 2009
<i>Message Reference Volume 2</i>	SC27-2451-00	Нет	Август 2009
<i>Net Search Extender Administration and User's Guide</i>	SC27-2469-02	Нет	Сентябрь, 2010
<i>Partitioning and Clustering Guide</i>	SC27-2453-01	Да	Ноябрь 2009
<i>pureXML Guide</i>	SC27-2465-01	Да	Ноябрь 2009
<i>Query Patroller Administration and User's Guide</i>	SC27-2467-00	Нет	Август 2009
<i>Spatial Extender and Geodetic Data Management Feature User's Guide and Reference</i>	SC27-2468-01	Нет	Сентябрь, 2010
<i>SQL Procedural Languages: Application Enablement and Support</i>	SC27-2470-02	Да	Сентябрь, 2010
<i>SQL Reference, Том 1</i>	SC27-2456-02	Да	Сентябрь, 2010
<i>SQL Reference, Том 2</i>	SC27-2457-02	Да	Сентябрь, 2010
<i>Troubleshooting and Tuning Database Performance</i>	SC27-2461-02	Да	Сентябрь, 2010
<i>Обновление до DB2 Версии 9.7</i>	SC89-1141-02	Да	Сентябрь, 2010
<i>Учебник по Наглядному объяснению</i>	SC89-1147-00	Нет	Август 2009
<i>What's New for DB2 Version 9.7</i>	SC27-2463-02	Да	Сентябрь, 2010
<i>Workload Manager Guide and Reference</i>	SC27-2464-02	Да	Сентябрь, 2010
<i>XQuery Reference</i>	SC27-2466-01	Нет	Ноябрь 2009

Таблица 24. Техническая информация о DB2 Connect

Имя	Номер формы	Доступна в печатном виде	Последнее изменение
<i>Установка и конфигурирование DB2 Connect Personal Edition</i>	SC89-1145-02	Да	Сентябрь, 2010
<i>Установка и конфигурирование серверов DB2 Connect</i>	SC89-1146-02	Да	Сентябрь, 2010
<i>DB2 Connect. Руководство пользователя</i>	SC89-1144-02	Да	Сентябрь, 2010

Таблица 25. Техническая информация об Information Integration

Имя	Номер формы	Доступна в печатном виде	Последнее изменение
<i>Information Integration: Administration Guide for Federated Systems</i>	SC19-1020-02	Да	Август 2009
<i>Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i>	SC19-1018-04	Да	Август 2009
<i>Information Integration: Configuration Guide for Federated Data Sources</i>	SC19-1034-02	Нет	Август 2009
<i>Information Integration: SQL Replication Guide and Reference</i>	SC19-1030-02	Да	Август 2009
<i>Information Integration: Introduction to Replication and Event Publishing</i>	GC19-1028-02	Да	Август 2009

Заказ печатных копий книг DB2

Об этой задаче

Если вам нужны печатные книги по DB2, то вы можете купить их через Интернет во многих (но не во всех) странах и территориях. Вы всегда можете заказать печатные книги по DB2 у своего местного представителя IBM. Имейте в виду, что некоторые книги, которые есть на диске DVD *Документация по DB2 в формате PDF*, нельзя заказать в печатном виде. В частности, оба тома *Справочника по сообщениям DB2* недоступны в печатном виде.

Печатные версии многих публикаций по DB2, находящихся на диске DVD *Документация по DB2 в формате PDF*, можно заказать за плату в IBM. В зависимости от того, где вы находитесь, вы, возможно, сможете заказать эти книги через Интернет, в центре публикаций IBM Publications Center. Если в вашей стране или территории заказ через Интернет не удастся сделать, вы всегда можете заказать

печатные книги по DB2 у вашего местного представителя IBM. Заметим, что некоторые книги, которые есть на диске DVD Документация по DB2 в формате PDF, нельзя заказать в печатном виде.

Примечание: Наиболее свежая и полная информация по DB2 находится в Информационном центре DB2 по адресу <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7>.

Чтобы заказать печатные книги DB2:

Процедура

- Чтобы узнать, можете ли вы в своей стране или территории заказать печатные книги по DB2 в Интернете, проверьте Центр публикаций IBM Publications Center по адресу: <http://www.ibm.com/shop/publications/order>. Чтобы перейти к информации о заказе, вам надо будет выбрать свою страну, область или язык, а затем следовать инструкциям для вашего местоположения.
- Чтобы заказать печатные книги по DB2 у своего местного представителя IBM:
 1. На одном из следующих сайтов найдите контактную информацию вашего местного представителя:
 - Всемирный указатель IBM по контактам: www.ibm.com/planetwide
 - Сайт Центра публикаций IBM <http://www.ibm.com/shop/publications/order>. Чтобы перейти к странице, соответствующей вашему местоположению, вам надо будет выбрать свою страну, область или язык. На этой странице выберите ссылку "Об этом сайте".
 2. Если будете звонить, скажите, что хотите заказать публикацию по DB2.
 3. Сообщите представителю заголовки и форм-номера книг, которые хотите заказать. Эти заголовки и форм-номера можно найти в разделе "Техническая библиотека DB2 в печатном виде или в формате PDF" на стр. 193.

Вызов справки по SQLSTATE из командной строки

Продукты DB2 возвращают SQLSTATE - значения, описывающие ошибки, которые могут возникнуть при выполнении оператора SQL. Справка по SQLSTATE объясняет значение состояний SQL и коды классов состояний SQL.

Процедура

Для вызова справки по SQLSTATE откройте процессор командной строки и введите:

```
? sqlstate или ? код класса
```

где *sqlstate* - допустимый пятизначный код SQL, а *код класса* - первые две цифры *sqlstate*.

Например, ? 08003 выводит справку по состоянию SQL 08003, а ? 08 выводит справку по коду класса 08.

Доступ к различным версиям Информационного центра DB2

Об этой задаче

Темы DB2 Версии 9.8 смотрите в *Информационном центре DB2* по адресу <http://publib.boulder.ibm.com/infocenter/db2luw/v9r8/>.

Для тем DB2 Версии 9.7 URL *Информационного центра DB2* - <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/>.

Для тем DB2 Версии 9.5 URL *Информационного центра DB2* - <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5>.

Для тем DB2 Версии 9.1 URL *Информационного центра DB2* - <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>.

Для тем DB2 Версии перейдите по URL *Информационного центра DB2*:
<http://publib.boulder.ibm.com/infocenter/db2luw/v8/>.

Вывод тем Информационного центра DB2 на предпочитаемом вами языке

Об этой задаче

Информационный центр DB2 по возможности выводит темы на языке, заданном в предпочтениях браузера. Если тема еще не переведена на предпочитаемый вами язык, Информационный центр DB2 выводит ее по-английски.

Процедура

- Для вывода тем на предпочитаемом вами языке в браузере Internet Explorer:
 1. В Internet Explorer выберите **Сервис** —> **Свойства обозревателя** —> **Языки...**
Откроется окно Выбор языка.
 2. Убедитесь, что предпочитаемый вами язык указан в первой строке списка языков.
 - Чтобы добавить в список новый язык, нажмите кнопку **Добавить...**

Примечание: При добавлении языка не гарантируется, что на компьютере есть шрифты, требуемые для вывода тем на предпочитаемом языке.
 - Чтобы переместить язык в начало списка, выделите язык и нажимайте кнопку **Выше**, пока он не окажется первым в списке языков.
 3. Обновите страницу, чтобы вывести Информационный центр DB2 на предпочитаемом вами языке.
- Для вывода тем на предпочитаемом вами языке в браузере Firefox или Mozilla:
 1. Выберите кнопку в разделе **Языки** диалогового окна **Инструменты** —> **Опции** —> **Дополнительно**. В окне Настройки появится панель Языки.
 2. Убедитесь, что предпочитаемый вами язык указан в первой строке списка языков.
 - Чтобы добавить в список новый язык, нажмите кнопку **Добавить...** и выберите нужный язык в окне Добавление языков.
 - Чтобы переместить язык в начало списка, выделите язык и нажимайте кнопку **Выше**, пока он не окажется первым в списке языков.
 3. Обновите страницу, чтобы вывести Информационный центр DB2 на предпочитаемом вами языке.

Результаты

При некоторых сочетаниях браузер-операционная система надо также изменить региональные настройки операционной системы на нужные вам локаль и язык.

Обновление Информационного центра DB2, установленного на вашем компьютере или на сервере интранет

Локально установленный Информационный центр DB2 надо периодически обновлять.

Прежде чем начать

Информационный центр DB2 Версия 9.7 должен уже быть установлен. Подробности смотрите в теме “Установка Информационного центра DB2 при помощи мастера по установке DB2” в книге *Установка серверов DB2*. Все предварительные требования и ограничения для установки Информационного центра касаются также и обновления Информационного центра.

Об этой задаче

Существующий Информационный центр DB2 можно обновлять автоматически или вручную:

- Автоматическое обновление - обновляет существующие возможности и языки Информационного центра. Дополнительное преимущество автоматического обновления состоит в том, что Информационный центр недоступен только короткий период времени в ходе обновления. Кроме того, можно задать периодический запуск автоматического обновления в составе других пакетных работ.
- Обновление вручную - должно выполняться, когда вы хотите в процессе обновления добавить возможности или языки. Например, локальный Информационный центр исходно был установлен на английском и на французском языке, но теперь вы хотите установить также русский язык; при обновлении вручную будет установлен русский язык и обновлена информация на английском и на французском языке. Однако при ручном обновлении от вас требуется вручную остановить, обновить и перезапустить Информационный центр. Информационный центр недоступен в ходе всего процесса обновления.

В данной теме подробно описан процесс автоматического обновления. Инструкции по обновлению вручную смотрите в теме “Обновление Информационного центра DB2, установленного на вашем компьютере или на сервере интранет, вручную”.

Процедура

Чтобы автоматически обновить DB2, установленный на вашем компьютере или на сервере интранета:

1. В операционных системах Linux:
 - a. Перейдите в каталог, где установлен Информационный центр. По умолчанию Информационный центр DB2 установлен в каталоге `/opt/ibm/db2ic/V9.7`.
 - b. Перейдите из каталога установки в каталог `doc/bin`.
 - c. Запустите сценарий `update-ic`:
`update-ic`
2. В операционных системах Windows:
 - a. Откройте командное окно.
 - b. Перейдите в каталог, где установлен Информационный центр. По умолчанию Информационный центр DB2 установлен в каталоге `<Program Files>\IBM\DB2 Information Center\Version 9.7`, где `<Program Files>` соответствует положению каталога Program Files.

- c. Перейдите из каталога установки в каталог doc\bin.
- d. Запустите файл update-ic.bat:
update-ic.bat

Результаты

Информационный центр DB2 будет перезапущен автоматически. Если обновления доступны, в Информационном центре будут показаны новые и обновленные темы. Если обновления Информационного центра недоступны, в журнал будет записано сообщение. Файл журнала находится в каталоге doc\eclipse\configuration. Имя файла журнала - случайным образом сгенерированное число, например, 1239053440785.log.

Обновление Информационного центра DB2, установленного на вашем компьютере или на сервере интранет, вручную

Если вы установили Информационный центр DB2 локально, можно получить обновления документации от IBM и установить их.

Об этой задаче

Для обновления вашего локально установленного *Информационного центра DB2* вручную:

1. Остановите *Информационный центр DB2* на вашем компьютере и перезапустите его в автономном режиме. Запуск Информационного центра в автономном режиме предотвращает доступ к нему других пользователей по сети и дает возможность применить исправления. Версия Информационного центра DB2 для рабочей станции всегда работает в автономном режиме. .
2. Чтобы посмотреть доступные обновления, воспользуйтесь возможностью Изменить. При помощи этой возможности можно получить и установить нужные вам обновления.

Примечание: Если в вашей среде необходимо установить обновления *Информационного центра DB2* на компьютере, не подключенном к интернету, создайте зеркальную копию сайта обновлений в локальной файловой системе, используя компьютер, который подключен к Интернету и на котором установлен *Информационный центр DB2*. Кроме того, создание зеркальной копии и прокси-сервера сайта обновлений полезно, если в вашей сети многие пользователи будут устанавливать обновления документации, и вы хотите сократить время, затраченное на отдельные обновления.

Если пакеты доступны, используйте эту возможность для получения пакетов. Но возможность обновления доступна только в автономном режиме.

3. Остановите Информационный центр в автономном режиме и перезапустите *Информационный центр DB2* на вашем компьютере.

Примечание: В Windows 2008 и Windows Vista (и новее) перечисленные далее в этом разделе команды должны вводиться администратором. Чтобы открыть окно командной строки или графическое средство со всеми привилегиями администратора, щелкните по ярлыку правой кнопкой и выберите **Выполнить от имени администратора**.

Процедура

Чтобы обновить *Информационный центр DB2*, установленный на вашем компьютере или на сервере интранета:

1. Остановите *Информационный центр DB2*.
 - В Windows выберите **Пуск > Панель управления > Администрирование > Службы**. Затем щелкните правой кнопкой мыши по службе **Информационный центр DB2** и выберите **Стоп**.
 - В Linux введите команду:
`/etc/init.d/db2icdv97 stop`
2. Запустите Информационный центр в автономном режиме.
 - В Windows:
 - a. Откройте командное окно.
 - b. Перейдите в каталог, где установлен Информационный центр. По умолчанию *Информационный центр DB2* установлен в каталоге `Program_Files\IBM\DB2 Information Center\Version 9.7`, где `Program_Files` соответствует положению каталога Program Files.
 - c. Перейдите из каталога установки в каталог `doc\bin`.
 - d. Запустите файл `help_start.bat`:
`help_start.bat`
 - В Linux:
 - a. Перейдите в каталог, где установлен Информационный центр. По умолчанию *Информационный центр DB2* установлен в каталоге `/opt/ibm/db2ic/V9.7`.
 - b. Перейдите из каталога установки в каталог `doc/bin`.
 - c. Запустите сценарий `help_start`:
`help_start`

В системном браузере по умолчанию откроется Информационный центр в автономном режиме.
3. Нажмите кнопку **Изменить** (🔧). (В вашем браузере должен быть разрешен JavaScript.) На правой панели Информационного центра выберите **Найти изменения**. Будет выведен список изменений для существующей документации.
4. Чтобы начать процесс установки, пометьте документы, которые вы хотите установить, затем нажмите кнопку **Установить изменения**.
5. Когда процесс установки будет завершен, нажмите кнопку **Готово**.
6. Остановите Информационный центр в автономном режиме:
 - В Windows перейдите в каталоге установки в подкаталог `doc\bin` и запустите файл `help_end.bat`:
`help_end.bat`

Примечание: Пакетный файл `help_end` содержит команды, необходимые для корректной остановки процессов, запущенных пакетным файлом `help_start`. Не используйте сочетание клавиш `Ctrl-C` или другие методы для остановки работы `help_start.bat`.
- В Linux перейдите в каталоге установки в подкаталог `doc/bin` и запустите сценарий `help_end`:
`help_end`

Примечание: Сценарий `help_end` содержит команды, необходимые для корректной остановки процессов, запущенных сценарием `help_start`. Не используйте другие методы для остановки работы сценария `help_start`.

7. Перезапустите *Информационный центр DB2*.

- В Windows выберите **Пуск > Панель управления > Администрирование > Службы**. Затем щелкните правой кнопкой мыши по службе **Информационный центр DB2** и выберите **Пуск**.
- В Linux введите команду:
`/etc/init.d/db2icdv97 start`

Результаты

В обновленном *Информационном центре DB2* будут показаны новые и измененные темы.

Учебные материалы по DB2

Обучающие программы DB2 помогают познакомиться с различными аспектами продуктов DB2. В уроках содержатся пошаговые инструкции.

Прежде, чем вы начнете

XHTML-версию обучающей программы можно просмотреть в Информационном центре на сайте <http://publib.boulder.ibm.com/infocenter/db2help/>.

В некоторых уроках используются примеры данных или кодов программ. Описание необходимых условий для выполнения задач обучающей программы смотрите в самой обучающей программе.

Учебные материалы по DB2

Для просмотра Обучающей программы щелкните по заголовку.

“pureXML” в книге *pureXML Guide*

Конфигурирование базы данных DB2 для хранения данных XML и выполнение основных операций со складом собственных данных XML.

“Учебник по Наглядному объяснению” в книге *Учебник по Наглядному объяснению*

Анализ, оптимизация и настройка операторов SQL для улучшения производительности при помощи Наглядного объяснения.

Информация об устранении неисправностей DB2

Для продуктов баз данных DB2 доступна разнообразная информация по устранению неисправностей и диагностике ошибок.

Документация DB2

Информацию по устранению неисправностей можно найти в публикации *Troubleshooting and Tuning Database Performance* или в разделе Основные понятия баз данных *Информационного центра DB2*. Информация по устранению неисправностей содержит темы о способах поиска и определения проблем с помощью инструментов диагностики и утилит DB2. Здесь вы найдете также решения наиболее часто возникающих проблем и другие советы по решению проблем, с которыми вы можете встретиться при работе с продуктами баз данных DB2.

Портал поддержки IBM

Смотрите портал поддержки IBM если вы сталкиваетесь с проблемами и вам нужна помощь в поиске причин и решений. На сайте технической поддержки есть ссылки на самые свежие публикации по DB2, TechNotes, исправления

программ (APAR), пакеты корректировки и прочие ресурсы. Для решения ваших проблем можно воспользоваться поиском по этой базе знаний.

Обратитесь к portalу поддержки IBM по адресу http://www.ibm.com/support/entry/portal/Overview/Software/Information_Management/DB2_for_Linux_UNIX_and_Windows.

Положения и условия

Разрешения на использование этих публикаций предоставляются в соответствии со следующими положениями и условиями.

Личное использование: вам предоставляется право воспроизводить эти публикации в личных некоммерческих целях при условии, что будут воспроизведены все замечания об авторских правах. Вам запрещается распространять эти публикации, полностью или по частям, демонстрировать их или создавать из них производные продукты без явного на то согласия от IBM.

Коммерческое использование: вам предоставляется право воспроизводить эти публикации исключительно в пределах своего предприятия при условии, что будут воспроизведены все замечания об авторских правах. За пределами вашего предприятия вам запрещается распространять эти публикации, полностью или по частям, демонстрировать их или создавать из них производные продукты без явного на то согласия от IBM.

На публикации или любую информацию, данные, программные средства или иную интеллектуальную собственность, содержащуюся в публикациях, предоставляются только права, явным образом предоставленные настоящим документом; никакие иные разрешения, лицензии и права не предоставляются.

IBM оставляет за собой право отозвать разрешения, предоставленные этим документом, если, по мнению IBM, использование публикаций наносит ущерб IBM или, как это установлено IBM, вышеприведенные инструкции не соблюдаются должным образом.

Вы имеете право загружать, экспортировать или реэкспортировать эту информацию только при условии соблюдения всех применимых законов и нормативных актов, включая все законы и нормативные акты США, касающиеся экспорта.

IBM НЕ ДАЕТ НИКАКИХ ГАРАНТИЙ ОТНОСИТЕЛЬНО СОДЕРЖАНИЯ ЭТИХ ПУБЛИКАЦИЙ. ДАННЫЕ ПУБЛИКАЦИИ ПРЕДОСТАВЛЕНЫ "КАК ЕСТЬ", БЕЗ КАКИХ-ЛИБО ГАРАНТИЙ, ЯВНЫХ ИЛИ ПОДРАЗУМЕВАЕМЫХ, ВКЛЮЧАЯ (НО НЕ ОГРАНИЧИВАЯСЬ ТАКОВЫМИ) ПРЕДПОЛАГАЕМЫЕ ГАРАНТИИ СОБЛЮДЕНИЯ АВТОРСКИХ ПРАВ, РЫНОЧНОЙ ПРИГОДНОСТИ ИЛИ СООТВЕТСТВИЯ ОПРЕДЕЛЕННОЙ ЦЕЛИ.

Приложение Е. Замечания

Эта информация описывает продукты и услуги, предлагаемые в США. Информация о продуктах других фирм (не IBM) основана на данных, доступных к моменту первой публикации этого и может быть изменена.

IBM может не предлагать описанные продукты, услуги и возможности в других странах. Сведения о продуктах и услугах, доступных в настоящее время в вашей стране, можно получить в местном представительстве IBM. Любые ссылки на продукты, программы или услуги IBM не означают явным или неявным образом, что можно использовать только продукты, программы или услуги IBM. Разрешается использовать любые функционально эквивалентные продукты, программы или услуги, если при этом не нарушаются права IBM на интеллектуальную собственность. Однако ответственность за оценку и проверку работы любых продуктов, программ и услуг других фирм лежит на пользователе.

Фирма IBM может располагать патентами или рассматриваемыми заявками на патенты, относящимися к предмету данного документа. Получение этого документа не означает предоставления каких-либо лицензий на эти патенты. Запросы по поводу лицензий следует направлять в письменной форме по адресу:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

По поводу лицензий, связанных с использованием наборов двухбайтных символов (DBCS), обращайтесь в отдел интеллектуальной собственности IBM в вашей стране или направьте запрос в письменной форме по адресу:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

Следующий абзац не применяется в Великобритании или в любой другой стране/регионе, где подобные заявления противоречат местным законам: КОРПОРАЦИЯ INTERNATIONAL BUSINESS MACHINES ПРЕДСТАВЛЯЕТ ДАННУЮ ПУБЛИКАЦИЮ “КАК ЕСТЬ” БЕЗ КАКИХ-ЛИБО ГАРАНТИЙ, ЯВНЫХ ИЛИ ПОДРАЗУМЕВАЕМЫХ, ВКЛЮЧАЯ ПРЕДПОЛАГАЕМЫЕ ГАРАНТИИ СОВМЕСТИМОСТИ, РЫНОЧНОЙ ПРИГОДНОСТИ И СООТВЕТСТВИЯ ОПРЕДЕЛЕННОЙ ЦЕЛИ, НО НЕ ОГРАНИЧИВАЯСЯ ИМИ. В некоторых странах для определенных сделок подобные оговорки не допускаются; таким образом, это утверждение может не относиться к вам.

Данная информация может содержать технические неточности и типографские опечатки. Периодически в информацию вносятся изменения, они будут включены в новые издания этой публикации. Фирма IBM может в любое время без уведомления вносить изменения и усовершенствования в продукты и программы, описанные в этой публикации.

Все ссылки в этой информации на сайты других фирм (не IBM) приводятся только для вашего удобства и ни в коей мере не означают одобрения этих сайтов. Материалы на этих сайтах не входят в состав материалов для данного продукта IBM; используйте их на свою ответственность.

IBM может использовать или распространять присланную вами информацию любым способом, как фирма сочтет нужным, без каких-либо обязательств перед вами.

Если обладателю лицензии на данную программу понадобятся сведения о возможности: (i) обмена данными между независимо разработанными программами и другими программами (включая данную) и (ii) совместного использования таких данных, он может обратиться по адресу:

IBM Canada Limited
U59/3600
3600 Steeles Avenue East
Markham, Ontario L3R 9Z7
CANADA

Такая информация может быть предоставлена на определенных условиях (в некоторых случаях к таким условиям может относиться оплата).

Лицензированная программа, описанная в данном документе, и все лицензированные материалы, доступные с ней, предоставляются IBM на условиях IBM Customer Agreement (Соглашения IBM с заказчиком), Международного соглашения о лицензиях на программы IBM или эквивалентного соглашения.

Приведенные данные о производительности измерены в контролируемой среде. Таким образом, результаты, полученные в других операционных средах, могут существенно отличаться от них. Некоторые показатели получены в системах разработки, и нет никаких гарантий, что в общедоступных системах эти показатели будут теми же. Более того, некоторые результаты могут быть получены путем экстраполяции. Реальные результаты могут отличаться от них. Пользователи должны проверить данные для своих конкретных сред.

Информация о продуктах других фирм получена от поставщиков этих продуктов, из их опубликованных объявлений или из других общедоступных источников. Фирма IBM не проверяла эти продукты и не может подтвердить точность измерений, совместимость или прочие утверждения о продуктах других фирм. Вопросы о возможностях продуктов других фирм следует направлять поставщикам этих продуктов.

Все утверждения о будущих планах и намерениях IBM могут быть изменены или отменены без уведомлений, и описывают исключительно цели фирмы.

Эта информация может содержать примеры данных и отчетов, иллюстрирующие типичные деловые операции. Чтобы эти примеры были правдоподобны, в них включены имена лиц, названия компаний и товаров. Все эти имена и названия вымышлены и любое их сходство с реальными именами и адресами полностью случайно.

ЛИЦЕНЗИЯ НА КОПИРОВАНИЕ:

Эта информация содержит примеры прикладных программ на языках программирования, иллюстрирующих приемы программирования для различных операционных платформ. Разрешается копировать, изменять и распространять эти

примеры программ в любой форме без оплаты фирме IBM для целей разработки, использования, сбыта или распространения прикладных программ, соответствующих интерфейсу прикладного программирования операционных платформ, для которых эти примера программ написаны. Эти примеры не были всесторонне проверены во всех возможных условиях. Поэтому IBM не может гарантировать их надежность, пригодность и функционирование. Программы примеров предоставляются "КАК ЕСТЬ", без каких-либо гарантий. IBM не несет ответственности за какой-либо ущерб, возникший в результате применения вами программ примеров.

Каждая копия программ примеров или программ, созданных на их основе, должна содержать следующее замечание об авторских правах:

© (название вашей фирмы) (год). Части этого кода построены на основе примеров программ IBM Corp. © Copyright IBM Corp. *вставьте год или годы*. Все права защищены.

Товарные знаки

IBM, логотип IBM и ibm.com - товарные знаки или зарегистрированные товарные знаки International Business Machines Corp в Соединенных Штатах и/или других странах. Другие названия продуктов и услуг могут быть товарными знаками IBM или других компаний. Текущий список товарных знаков IBM доступен в Web под заголовком Copyright and trademark information на сайте www.ibm.com/legal/copytrade.shtml.

Следующие термины - товарные знаки или зарегистрированные товарные знаки других компаний.

- Linux - зарегистрированный товарный знак Linus Torvalds в Соединенных Штатах и/или других странах.
- Java и все товарные знаки и логотипы на основе Java - товарные знаки или зарегистрированные товарные знаки Oracle и/или ее филиалов.
- UNIX - зарегистрированный товарный знак The Open Group в Соединенных Штатах и в других странах.
- Intel, логотип Intel, Intel Inside, логотип Intel Inside, Intel Centrino, логотип Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium и Pentium - товарные знаки или зарегистрированные товарные знаки of Intel Corporation или ее филиалов в Соединенных Штатах и/или других странах.
- Microsoft, Windows, Windows NT и логотип Windows - товарные знаки корпорации Microsoft в США и/или других странах.

Названия других компаний, продуктов и услуг могут быть товарными знаками или марками сервиса других фирм.

Индекс

A

Accessories Suite
установка 52, 54

D

DB2 Text Search 19, 21, 78, 79
команда CREATE INDEX 111
команда UPDATE INDEX 123
полномочия
администратор баз данных 12
владелец пользовательской таблицы 12
потребление дискового пространства 20
форматы документов
преобразование неподдерживаемых 25

L

Linux
установка
Мастер по установке DB2 32
серверы DB2 32
файл ответов 38

N

Net Search Extender (NSE)
сравнение с Text Search 173

T

TCP/IP
обновление файла services 45

U

UNIX
установка
Мастер по установке DB2 32
при помощи мастера по установке DB2 32
установка с помощью файла ответов 38

W

Windows
установка
при помощи файлов ответов 37
серверов DB2 при помощи мастера по установке DB2 29

X

XML
текстовый поиск DB2
синтаксис поиска 88
Текстовый поиск DB2 (DB2 Text Search)
грамматика EBNF 85

A

аргументы 77

Б

бесхозные собрания текстового поиска DB2
идентификация 60
удаление 60
библиотеки фильтров 52, 54

В

включение 58
внесение в каталог
узлы TCP/IP 45

Д

Дескрипторы файлов 24
диагностика ошибок
доступная информация 202
учебные материалы 202
дисковое пространство
потребление при использовании текстового поиска DB2 20
документация
обзор 193
печатные 193
положения и условия использования 203
файлы PDF 193

З

заказ книг DB2 196
замечания 205

И

изменения
индексы текстового поиска DB2 67
Информационный центр DB2 199, 200
Информационный центр DB2
версии 197
изменение 199, 200
языки программирования 198

К

книги
заказ 196
команда db2_install
установка продукта DB2 39
команда db2icrt
подробности 182
команда db2idrop
подробности 186
команда db2iupdt
подробности 188

- команда db2iupgrade
 - подробности 179
- команда doce_install
 - установка продуктов 39
- команда HELP
 - Текстовый поиск 128
- команда обновления экземпляра 179
- команда обновления экземпляров 188
- команда создания экземпляра 182
- Команда текстового поиска ALTER INDEX 119
- Команда текстового поиска CLEANUP FOR TEXT 105
- Команда текстового поиска CLEAR COMMAND LOCKS 109
- команда текстового поиска CLEAR EVENTS FOR INDEX 127
- Команда текстового поиска CREATE INDEX 111
- Команда текстового поиска DISABLE DATABASE FOR TEXT 107
- Команда текстового поиска DROP INDEX 117
- Команда текстового поиска ENABLE DATABASE FOR TEXT 106
- Команда текстового поиска START FOR TEXT 104
- Команда текстового поиска STOP FOR TEXT 104
- Команда текстового поиска UPDATE INDEX 123
- команда удаления экземпляра 186
- команды
 - ALTER INDEX 69
 - CLEAR EVENTS FOR INDEX 68
 - db2icrt 182
 - db2idrop
 - подробности 186
 - db2iupdt 188
 - db2iupgrade
 - подробности 179
 - db2ts
 - сводка (summary) 103
 - db2ts ALTER INDEX 119
 - db2ts CLEANUP FOR TEXT 105
 - db2ts CLEAR COMMAND LOCKS 109
 - db2ts CLEAR EVENTS FOR INDEX 127
 - db2ts CREATE INDEX 111
 - db2ts DISABLE DATABASE FOR TEXT 107
 - db2ts DROP INDEX 117
 - db2ts ENABLE DATABASE FOR TEXT 106
 - db2ts HELP 128
 - db2ts START FOR TEXT 104
 - db2ts STOP FOR TEXT 104
 - db2ts UPDATE INDEX 123
 - DISABLE DATABASE FOR TEXT 59
 - DROP INDEX 70
 - ENABLE DATABASE FOR TEXT 51
 - UPDATE INDEX 67
 - вызов из процедуры 131
 - текстовый поиск 103
- команды db2ts
 - ALTER INDEX 119
 - CLEANUP FOR TEXT 105
 - CLEAR COMMAND LOCKS 109
 - CLEAR EVENTS FOR INDEX 127
 - CREATE INDEX 111
 - DISABLE DATABASE FOR TEXT 107
 - DROP INDEX 117
 - ENABLE DATABASE FOR TEXT 106
 - HELP 128
 - START FOR TEXT 104
 - STOP FOR TEXT 104
 - UPDATE INDEX 123
- конфигурация
 - текстовый поиск DB2 56

- конфигурирование 58

Л

- лингвистическая обработка
 - использование сегментации энграмм 7

М

- Мастер по установке DB2
 - установка
 - серверы DB2 (Linux) 32
 - серверы DB2 (UNIX) 32
 - установка серверов DB2
 - Linux 32
 - UNIX 32
 - установка серверов DB2 в UNIX 32
- множественные предикаты 78

Н

- настройка 14
- настройка конфигурации 14

О

- обслуживание индекса текстового поиска
 - восстановление индекса текстового поиска 71
 - резервное копирование индекса текстового поиска 71
- операторы SQL
 - справка
 - вывод 197
- оптимизация индекса 19

П

- параметры, специфичные для индексов, для обновлений 21
- планирование и оптимизация индексов 18
- планирование мощности 12
- положение индексов 21
- положения и условия
 - публикации 203
- потоки индексирования 15
- потребление памяти кучей 14
- производные таблицы для DB2 Text Search
 - информация уровня базы данных
 - SYSIBMTS.TSDEFAULTS 159
 - SYSIBMTS.TSLOCKS 160
 - обзор 159
 - информация уровня индекса
 - SYSIBMTS.TSCOLLECTIONNAMES 160
 - SYSIBMTS.TSCONFIGURATION 162
 - SYSIBMTS.TSEVENT 162
 - SYSIBMTS.TSINDEXES 161
 - SYSIBMTS.TSSTAGING 163
 - обзор 159
- процедура SYSTS_ADMIN_CMD 131
- процедура SYSTS_CREATE 136
- процедура SYSTS_DISABLE 134
- процедуры
 - SYSTS_ADMIN_CMD 131

P

- размер памяти очереди 17
- расширенный текст 58
 - конфигурирование DB2 Text Search 56
 - обзор 10
 - установка DB2 Accessories Suite 52, 54

C

- серверы DB2
 - установка
 - Windows 29
- словари синонимов
 - добавление 72
 - обзор 72
 - удаление 73, 169
- справка
 - задание языка 198
 - операторы SQL 197
- столбцов XML
 - текстовый поиск 98

T

- текстовый поиск 58
- Текстовый поиск 14
- текстовый поиск DB2 12, 14, 15, 77
 - асинхронное индексирование 19
 - выключение поддержки расширенного текста 62
 - деинсталляция DB2 Accessories Suite 56
 - документы XML 88
 - индексы
 - двоичные типы данных 65
 - заполнение 67
 - обновление 67
 - повышение производительности 66
 - создание 63
 - команда ALTER INDEX 119
 - команда CLEAR COMMAND LOCKS 109
 - команда CLEAR EVENTS FOR INDEX 127
 - команда DISABLE DATABASE FOR TEXT 107
 - команда DROP INDEX 117
 - команда ENABLE DATABASE FOR TEXT 106
 - команда HELP 128
 - команда START FOR TEXT 104
 - команда STOP FOR TEXT 104
 - команды
 - ALTER INDEX 119
 - CLEANUP FOR TEXT 105
 - CLEAR COMMAND LOCKS 109
 - CLEAR EVENTS FOR INDEX 127
 - CREATE INDEX 111
 - DISABLE DATABASE FOR TEXT 107
 - DROP INDEX 117
 - ENABLE DATABASE FOR TEXT 106
 - HELP 128
 - START FOR TEXT 104
 - STOP FOR TEXT 104
 - UPDATE INDEX 123
 - команды управления 103
 - конфигурирование
 - обзор 27
 - поддержка документов в расширенном текстовом формате 56
 - способы 42
 - обзор 1

- текстовый поиск DB2 (продолжение)
 - обновление от имени рядового пользователя (не root) 48
 - отбрасывание индексов 70
 - переконфигурирование 42
 - перенастройка
 - Net Search Extender в DB2 Text Search 175
 - планирование мощности
 - требования к портам TCP/IP 23
 - повышение производительности при индексации 66
 - поддержка документов в расширенном текстовом формате 10, 52, 54, 56
 - поиск при помощи функции SCORE 80
 - полномочия
 - пользователь, выполняющий запросы текстового поиска 12
 - просмотр состояния индекса 70
 - триггеры 19
 - управляющие подпрограммы 131
 - управляющие производные таблицы
 - SYSIBMTS.TSINDEXES 161
 - промежуточная таблица (staging table) 163
 - уровня индекса 161
 - установка
 - библиотеки фильтров DB2 Accessories Suite 52, 54
 - обзор 27
 - функции поиска 75
 - функция CONTAINS 75
 - функция SCORE 75
 - Функция xmlcolumn-contains 75
- Текстовый поиск DB2 (DB2 Text Search) 14, 17, 18, 21, 22, 24
 - SQL 79
 - XQuery 79
 - включение для баз данных 51
 - вызов команд 9
 - диагностика ошибок 171
 - добавление словаря синонимов 72, 169
 - документы XML 85
 - запуск 51
 - изменение индексов 69
 - изменение характеристик обновления 69
 - индексы
 - изменение 6
 - обновление 6
 - обслуживание 67
 - поиск 79
 - создание 6, 62
 - создание (двоичные типы данных) 65
 - создание (неподдерживаемые типы данных) 65
 - инструмент synonymTool 169
 - инструмент конфигурирования 44, 165
 - инструмент управления 167
 - инструмент форматирования журналов 171
 - инструменты командной строки 9
 - конфигурирование
 - инструмент конфигурирования 44, 165
 - мастер по установке 29
 - поиск XML 87
 - файл ответов 36
 - лингвистическая обработка 7
 - неподдерживаемые типы данных 65
 - обзор 3, 10, 173
 - обновление 47
 - особенности защиты 10
 - остановка 59
 - отключение базы данных для 59
 - переконфигурирование 44
 - планирование запросов 77

Текстовый поиск DB2 (DB2 Text Search) *(продолжение)*

- повышение производительности поиска 81
- поддерживаемые кодовые страницы 25
- поддерживаемые языки 25
- поиск с использованием индексов 75
- полномочия
 - владелец экземпляра 11
 - роли 11
- роли
 - администратор баз данных 12
 - владелец пользовательской таблицы 12
 - владелец экземпляра 11
 - пользователь, выполняющий запросы 12
- роли пользователей 11
- синтаксис аргумента поиска 81
- словари синонимов
 - обзор 72
 - удаление 169
- сравнение Net Search Extender 173
- столбцов XML 98
- сценарий 3
- таблицы журналов 62
- таблицы событий
 - обзор 62
 - удаление событий 68
- типы данных
 - поддерживаемые 24
 - преобразование неподдерживаемых 25
- триггеры 62
- удаление словаря синонимов 73, 169
- управляющие производные таблицы
 - SYSIBMTS.TSCOLLECTIONNAMES 160
 - SYSIBMTS.TSCONFIGURATION 162
 - SYSIBMTS.TSDEFAULTS 159
 - SYSIBMTS.TSEVENT 162
 - SYSIBMTS.TSLOCKS 160
 - SYSIBMTS.TSSTAGING 163
 - таблица журнала (log table) 163
 - таблица событий 162
 - уровня базы данных 159, 160
 - уровня индекса 159, 160, 162, 163
- усечение документов 26
- установка
 - команда db2_install 39
 - мастер по установке 29
 - файл ответов 36
- форматы документов
 - поддерживаемые 24
- функции 75
- функции поиска SQL 93
- функции поиска XML 93
- функция CONTAINS 93
- функция SCORE 78, 96

У

- установка
 - вручную
 - db2_install 39
 - doce_install 39
 - команда db2_install 39
 - команда doce_install 39
 - Текстовый поиск DB2 (DB2 Text Search) 39
- устранение неисправностей
 - оперативная информация 202
 - учебные материалы 202

- учебные материалы
 - диагностика ошибок 202
 - Наглядное объяснение 202
 - список 202
 - устранение неисправностей 202

Ф

- файл services
 - обновление для связи TCP/IP 45
- файлы ответов
 - запуск установки 37
 - установка
 - Linux 38
 - UNIX 38
- функции
 - XQuery
 - xmlcolumn-contains 98
 - хранимые процедуры
 - SYSTS_ALTER 146
 - SYSTS_CLEAR_COMMANDLOCKS 156
 - SYSTS_CLEAR_EVENTS 154
 - SYSTS_DROP 144
 - SYSTS_ENABLE 132
 - SYSTS_UPDATE 151
- функции XQuery
 - xmlcolumn-contains 98
- функция CONTAINS
 - текстовый поиск 93
- функция RESULTLIMIT 79
- функция SCORE
 - поиск в индексах текстового поиска 96
- функция xmlcolumn-contains 98

Х

- характеристики источников индексов 18
- Хранимая процедура SYSTS_ALTER 146
- Хранимая процедура SYSTS_CLEAR_COMMANDLOCKS 156
- Хранимая процедура SYSTS_CLEAR_EVENTS 154
- Хранимая процедура SYSTS_DROP 144
- Хранимая процедура SYSTS_ENABLE 132
- Хранимая процедура SYSTS_UPDATE 151



Напечатано в Дании

SC89-1250-03



Spine information:

DB2 для Linux, UNIX и Windows версия 9 выпуск 7

Руководство по DB2 Text Search

