

IBM[®] DB2[®] Universal Database



Замечания по выпуску

Версия 7.1 FixPak 3

IBM[®] DB2[®] Universal Database



Замечания по выпуску

Версия 7.1 FixPak 3

Содержание

Добро пожаловать в систему DB2 Universal Database Версии 7.1! xiii

Часть 1. Информация по установке 1

Глава 1. Информация по установке . . . 3

Установка наборов файлов для таблиц преобразования DBCS и Unicode в AIX 3

Часть 2. Особые замечания 5

Глава 2. Особые замечания 7

Специальные возможности DB2 UDB Версии 7.1 7

Ввод с клавиатуры и навигация 7

Средства облегчения работы с экраном 7

Альтернативные средства предупреждения 7

Совместимость с технологиями для людей с физическими недостатками 8

Удобный формат документации 8

Поддерживаемые браузеры Web в операционной системе Windows 2000 8

Дополнительный необходимый уровень исправления для Solaris 8

Поддерживаемые процессоры для DB2 Версии 7.1 для Solaris 8

Поиск электронной информации DB2 в Solaris 9

Проблемы при добавлении узлов в многораздельную базу данных 9

Ошибки при перенастройке 11

Исправление китайской национальной версии для Red Flag Linux 11

Программа установки DB2 может зависнуть, если не подключен съемный диск 11

Дополнительный параметр национальной версии для DB2 for Linux в средах Linux с японским и упрощенным китайским языком 11

Ошибка Центра управления в Microsoft Internet Explorer 12

Несовместимость между DB2 и Sybase в среде Windows 12

Потеря функции Центра управления 13

Часть 3. Установка и конфигурирование 15

Глава 3. Общая информация по установке 17

Загрузка установочных пакетов для всех поддерживаемых клиентов DB2 17

Установка DB2 в Windows 2000 17

Проблемы перенастройки производных таблиц, в определении которых использованы специальные регистры 17

Поддержка протокола IPX/SPX в Windows 2000 17

Остановка процессов DB2 перед обновлением предыдущей версии DB2 18

После установки DB2 выполните db2iupdt, если уже установлен другой продукт DB2 18

Настройка среды Linux для работы DB2 18

DB2 Universal Database Enterprise Edition и DB2

Connect Enterprise Edition для Linux на S/390 19

DB2 Universal Database Enterprise - Extended Edition теперь доступна для Linux 19

Параметр ядра shmseg для HP-UX 19

Глава 4. Data Links Manager Quick Beginnings 21

Ошибка запуска Dlfm с сообщением: "Ошибка получения afsfid для префикса" 21

Задание класса Tivoli Storage Manager для файлов архивов 21

Требования к дисковому пространству для DFS Client Enabler 21

Мониторинг процессов внутренней обработки менеджера файлов связей данных в AIX 21

Установка и конфигурирование менеджера связей данных DB2 для AIX: Особенности установки в средах DCE-DFS 22

Ошибка выполнения команды "dlfm add_prefix" 22

Установка и конфигурирование менеджера связей данных DB2 для AIX: Использование утилиты db2setup для установки менеджера связей данных DB2 в AIX 22

Установка и конфигурирование менеджера связей данных DB2 для AIX: Задачи после установки в DCE-DFS 23

Установка и конфигурирование менеджера связей данных DB2 для AIX: Ручная установка менеджера связей данных DB2 с помощью SMIT 23

Установка и конфигурирование Client Enabler DFS для связей данных DB2 23

Установка и конфигурирование менеджера связей данных DB2 для Solaris 24

Выбор метода резервного копирования для менеджера связей данных DB2 в AIX 24

Выбор метода резервного копирования для менеджера связей данных DB2 в Windows NT 25

Резервное копирование файловой системы с журналами в AIX 27

Привилегии группы администратора в связях данных в Windows NT 27

Минимизация записи в журнал при установке фильтра файловой системы DataLinks (DataLinks File System Filter - DLFF) 28

Запись сообщений в журнал после установки 29

Деинсталляция компонентов DLFM с использованием SMIT может удалить дополнительные наборы файлов Before You Begin/Determine hostname (Прежде, чем вы начнете/Определение имени хоста)	29
Работа с менеджером файлов связей данных: очистка после удаления менеджера связей данных DB2 из базы данных DB2	31
DLFM1001E (новое сообщение об ошибке)	31
Опция задания файла конфигурации DLFM	32
Ошибка при работе сценария связей данных/DFS dmapr_prestart в AIX	32
Глава 4. Installing and Configuring DB2 Data Links Manager for AIX (Установка и конфигурирование менеджера связей данных DB2 для AIX)	32
Общие замечания по установке	32
Приложение А. DB2 Data Links Manager Errors and User Responses (Ошибки менеджера связей данных DB2 и действия пользователя)	32

Глава 5. Дополнение по установке и настройке 33

Связывание утилит базы данных при использовании клиента времени выполнения	33
Доступ к DB2 с использованием ODBC из клиентов UNIX	33
Переключение NetQuestion for OS/2 на использование TCP/IP	33
Глава 26. Настройка системы объединения на работу с источниками данных Oracle	33

Часть 4. Управление 35

Глава 6. Руководство администратора: Планирование 39

Глава 8. Физическая структура базы данных	39
Ключи разделения	39
Разработка групп узлов	39
Глава 9. Разработка распределенных баз данных	39
Изменение нескольких баз данных	39
Глава 13. Высокая доступность в среде Windows NT	40
Необходимость перезагрузки компьютера перед запуском утилиты DB2MCS	40
Глава 14. DB2 и высокая доступность Sun Cluster 2.2	40
Поддержка Veritas в Solaris	40
Приложение В. Правила именования	41
Замечания об ID пользователей и именах схем длиной более 8 символов	41
ID пользователей и пароли	42
Приложение D. Несовместимость выпусков	42
DLFS Windows NT несовместима с Norton Utilities	42
SET CONSTRAINTS изменено на SET INTEGRITY	43
Приложение Е. Поддержка национальных языков	43
Национальные версии DB2 Версии 7.1	43
Национальные настройки для сервера администратора DB2	46
DB2 UDB на платформах Windows поддерживает кодovou страницу для стран Балтии (MS-1257)	46
Определение значений кодовых страниц	46
Поддержка кода страны и кодовой страницы	46
Наборы символов	47

Глава 7. Руководство администратора: Реализация 49

Добавление или расширение контейнеров DMS (новый процесс)	49
Глава 1. Управление DB2 при помощи инструментов GUI	49
Глава 3. Создание базы данных	50
Создание табличного пространства	50
Создание последовательности	51
Сравнение столбцов идентификации (IDENTITY) и последовательностей	53
Создание индекса, расширения индекса или спецификации индекса	53
Глава 4. Изменение базы данных	54
Добавление к табличному пространству SMS контейнера на разделе	54
Изменение столбца идентификации	54
Изменение последовательности	55
Отбрасывание последовательности	55
Переключение состояния табличного пространства	55
Изменение контейнеров в табличном пространстве DMS	56
Глава 5. Управление доступом к базам данных	56
Привилегии последовательностей	56
Шифрование данных	56
Глава 8. Восстановление базы данных	57
Как использовать приостановленный ввод-вывод	58
Инкрементное резервное копирование и восстановление	59
Параллельное восстановление	62
Резервное копирование в именованные конвейеры	62
Резервное копирование из отдельной копии	63
Архивирование журнала по требованию	65
Создание зеркальной копии журнала	65
Поддержка кроссплатформенного резервного копирования и восстановления на Sun Solaris и HP	66
Особенности менеджера связей данных DB2/Особенности утилиты резервного копирования	66
Особенности менеджера связей данных DB2/Особенности утилиты восстановления и повтора транзакций	67
Восстановление баз данных из резервных копий, сделанных в автономном режиме, без повтора транзакций	68
Восстановление баз данных и табличных пространств и повтор транзакций до конца файлов журнала	69
Взаимодействие менеджера связей данных и восстановления	69
Обнаружение ситуации, требующей согласования	72
Приложение С. Обработчик пользователя для восстановления базы данных	73
Приложение I. Высокоскоростная межузловая связь	74
Подготовка DB2 к работе с использованием VI	74

Глава 8. Руководство администратора: Производительность 77

Глава 3. Особенности прикладных программ	77
--	----

Задание уровня изоляции	77	db2look - Утилита извлечения статистики DB2	104
Настройка класса оптимизации	77	db2upd7 - обновление базы данных до уровня	
Динамические составные операторы	78	текущего пакета FixPak Версии 7	104
Глава 4. Особенности среды	78	Новая опция процессора командной строки (-x,	
Использование индексных ключей большего		подавление вывода заголовков столбцов)	105
размера.	78	Для процессора командной строки DB2 необходим	
Глава 5. Статистики системного каталога	79	шрифт True Type	105
Сбор и использование статистики распределения	79	ADD DATALINKS MANAGER	105
Правила обновления статистики каталога	79	ARCHIVE LOG (новая команда)	106
Статистика подэлементов.	80	Archive Log	107
Глава 6. Компилятор SQL.	82	BACKUP DATABASE	108
Реплицируемые сводные таблицы	82	Синтаксическая диаграмма	108
Концепции доступа к данным и оптимизация	84	Особенности менеджера связей данных DB2	109
Глава 8. Производительность работы	84	BIND	109
Управление пулом буферов базы данных	84	CALL	110
Управление несколькими пулами буферов базы		DROP DATALINKS MANAGER (новая команда)	111
данных	85	DROP DATALINKS MANAGER	112
Глава 9. Использование утилиты ограничения		EXPORT	115
ресурсов	86	GET DATABASE CONFIGURATION	115
Глава 13. Конфигурирование DB2	86	GET ROUTINE (новая команда)	116
Размер кучи сортировки (sortheap)	86	GET ROUTINE	117
Порог кучи сортировки (sheapthres).	87	GET SNAPSHOT	117
Максимальный процент списка блокировок перед		IMPORT	117
расширением (maxlocks)	87	LIST HISTORY	118
Конфигурирование для DB2/Менеджера связей		LOAD	119
данных DB2 интервала истечения срока маркера		PING (новая команда).	120
доступа к связям данных dl_exptint	88	PING	121
Параметр конфигурации базы данных		PUT ROUTINE (новая команда)	122
MIN_DEC_DIV_3	88	PUT ROUTINE	123
Размер кучи управления программами		RECONCILE	123
(app_ctl_heap_sz)	89	REORGANIZE TABLE	124
Размер кучи системного монитора базы данных		RESTORE DATABASE	124
(mon_heap_sz).	89	Синтаксис	124
Максимальное число активных прикладных		Особенности менеджера связей данных DB2	125
программ (maxappls)	89	ROLLFORWARD DATABASE	125
Диапазон восстановления и интервал мягких		UPDATE HISTORY FILE	125
контрольных точек (softmax)	90	Ошибка документации в разделе CLP Return Codes	
Разрешить отслеживание измененных страниц		(Коды возврата процессора командной строки)	126
(trackmod)	90		
Изменить путь журнала базы данных (newlogpath)	90		
Положение файлов журналов (logpath).	91		
Максимальная память для списка блокировок			
(locklist)	92		
Приложение А. Переменные реестра DB2 и			
переменные среды	92		
Таблица новых и измененных переменных реестра	92		
Приложение С. Средства объяснения SQL	96		
Глава 9. Command Reference	97	Глава 10. Data Movement Utilities	
db2batch - Замер производительности	97	Guide and Reference	127
db2cap (новая команда)	97	Состояния отложенных действий после операции	
db2cap - Утилита связывания статического пакета		загрузки	127
CLI/ODBC	98	Ограничения загрузки.	127
db2ckrst - Проверка последовательности резервных		Для запуска автозагрузки при значении	
копий для инкрементного восстановления (новая		Authentication YES необходим rexecd	127
команда)	199	Модификатор типа файла totalreespace	127
db2gncol (новая команда)	101	Использование импорта с буферизованными	
db2gncol - Изменить значения генерируемых		вставками	127
столбцов	102		
db2inidb - инициализировать зеркальную копию			
базы данных.	103		
		Глава 11. Replication Guide and	
		Reference	129
		Репликация в Windows 2000	129
		Имена столбцов и таблиц	129
		Репликация столбцов DATALINK	129
		Ограничения на большие объекты.	129
		Репликация и серверы других типов (не IBM)	130
		Предварительное требование для повсеместного	
		изменения	130
		Сценарии репликации	130

Планирование репликации	131
Настройка среды репликации	131
Диагностика ошибок	131
Программы Capture и Apply для AS/400	132
Структуры таблиц	133
Сообщения программ Capture и Apply	133
Запуск программ Capture и Apply из прикладной программы	133

Глава 12. System Monitor Guide and Reference	135
db2ConvMonStream.	135

Глава 13. Troubleshooting Guide	137
Запуск DB2 в Windows 95, Windows 98 и Windows ME, когда пользователь не зарегистрирован в системе	137

Глава 14. Использование DB2 Universal Database на 64-битных платформах	141
Глава 5. Configuration	141
shmsys:shminfo_shmmax	141
DBHEAP	141
LOCKLIST	141
Глава 6. Restrictions	141

Глава 15. XML Extender Administration and Programming	143
--	------------

Часть 5. Инструменты управления 145

Глава 16. Центр управления.	147
Возможность управления серверами DB2 Server for VSE and VM	147
Поддержка Java 1.2 для Центра управления.	147
Ошибка "Неверный ярлык" при использовании электронной справки в операционной системе Windows	148
Центр управления в виде Java в OS/2	148
Ошибка "Доступ к файлу запрещен" при попытке просмотреть завершённое задание в журнале операционной системы Windows	148
Соединение для проверки многоузлового изменения	148
Центр управления для DB2 for OS/390	148
Необходимое исправление для Центра управления для OS/390	150
Изменения в окне Создать географический слой	150
Информация по устранению неисправностей для Центра управления DB2	150
Устранение неисправностей для Центра управления в системах на основе UNIX	150
Возможные проблемы со всплывающими справками в OS/2	150
Запуск нескольких апплетов Центра управления	151
Справка по параметру конфигурации jdk11_path	151

Системная ошибка Solaris (SQL10012N) при использовании Центра сценариев или журнала	151
Справка по файлу DPREPL.DFT	151
Электронная справка по Центру управления, выполняемому как апплет	151
Выполнение Центра управления в режиме апплета (Windows 95).	152
Центр управления DB2 for OS/390	152

Глава 17. Информационный центр 153	
Ошибка "Неверный ярлык" в операционной системе Windows	153
Открытие внешних ссылок Web в Netscape Navigator, если Netscape уже открыт (системы на основе UNIX)	153
Проблемы запуска Информационного центра	153

Глава 18. Мастера 155	
Задание размера экстента в мастере по созданию базы данных	155

Часть 6. Возможности для бизнеса 157

Глава 19. Центр хранилищ данных 159	
Создание шагов SQL с информацией об издании	159
Изменение источников и назначений в окне Модель процесса	159
Добавление описаний к объектам Центра хранилищ данных	159
Запросы примера содержания	159
Editing a Create DDL SQL statement.	159
Перенастройка бизнес-просмотров Visual Warehouse	160
Генерация таблиц назначения и первичных ключей	160
Использование драйверов ODBC Merant.	160
Новый драйвер ODBC	160
Определение источника или потребителя хранилища в базе данных OS/2.	161
Наблюдение за состоянием управляющие базы данных хранилища.	161
Использование SQL Assist с базой данных примера TBC_MD	161
Использование функции FormatDate	161
Изменение параметра языка	161
Использование преобразователя Генерировать таблицу ключей.	162
Поддержание соединений с базами данных	162
Установка удаленного клиента Центра хранилищ данных	162
Определение источника хранилища DB2 for VM	162
Определение таблицы назначения DB2 for VM или DB2 for VSE	162
Включение поддержки идентификаторов с ограничителями	162
Ошибка Data Joiner, указывающая на проблемы связывания	163
Настройка и выполнение репликации с помощью Центра хранилищ данных	163
Советы по устранению неисправностей	166
Поддерживаемые источники и потребители.	166

Дополнения для поддерживаемых источников других типов (не баз данных IBM)	168
Создание источника данных вручную в Центре хранилищ данных	169
Импорт и экспорт метаданных с помощью CWM (Common Warehouse Metadata Interchange - общий обмен метаданными хранилищ)	170
Введение	170
Импорт метаданных	170
Изменение ваших метаданных после запуска утилиты импорта	172
Экспорт метаданных	172
Шаг утилиты Runstats в OS/390	174
Шаг утилиты Load в OS/390	174

Глава 20. Публикации по Центру хранилищ данных 175

Data Warehouse Center Application Integration Guide	175
Data Warehouse Center Administration Guide	175
Устранение неисправностей	175
Задание Excel как источника хранилища	175
Определение и запуск процессов	175
Определение значений для программы VWPMVS	175
Изменения в приложении Пример хранилища данных	176
Сообщения Центра хранилищ данных	176
Создание макета и загрузка данных на сервере интеграции OLAP DB2	176
Использование Classic Connect с Центром хранилищ данных	176
Структура среды Центра хранилищ данных	177
Использование преобразователя инвертирования	177
Доступ агента хранилища DB2 Версии 7.1 к данным DB2 Версии 5	177
Глава 3. Setting up warehouse sources	179
Глава 10. Maintaining the Warehouse Database	179
Сообщения Центра хранилищ данных	179
Электронная справка Центра хранилищ данных	179
Определение таблиц или производных таблиц для репликации	179
Выполнение VWP Essbase с агентом AS/400	180
Использование окна Сделать доступными метаданные Центра хранилищ данных и связанного с ним окна свойств	180
Внешние ключи	180
Записные книжки репликации	180
Импорт языка тегов	180
Ссылки для добавления данных	180
Импорт таблиц	180
Исправления для электронной справки по RUNSTATS и REORGANIZE TABLE	180
Страница Уведомление (записные книжки Свойства хранилища данных и Расписание)	181
Поле Модуль агента в записной книжке Системы агентов	181
Исправленный учебник Business Intelligence Tutorial	181

Глава 21. Управляющая база данных хранилища 183

Управляющая база данных хранилища по умолчанию	183
Окно Менеджер управляющей базы данных хранилища	183
Изменение активной управляющей базы данных хранилища	183
Создание и инициализация управляющей базы данных хранилища	184
Перенастройка управляющих баз данных IBM Visual Warehouse	184
Доступ к управляющим базам данных хранилища	185

Глава 22. Начальный комплект OLAP 187

Web-сайт сервера OLAP	187
Установка Начального комплекта OLAP DB2 в AIX и Solaris	187
Регистрация с рабочего стола Сервера интеграции OLAP	187
Пример регистрации для Начального комплекта	188
Создание и конфигурирование вручную баз данных примера для Сервера интеграции OLAP	188
Известные проблемы и ограничения	189
Начальный комплект OLAP DB2	190
Клиент рабочего стола OLAP DB2	190
Клиенты электронных таблиц	191
Сервер интеграции OLAP DB2	192
Для модуля электронных таблиц Начального комплекта OLAP требуется последнее исправление Windows	194
Отсутствие файлов EQD встраиваемого модуля OLAP для электронных таблиц	194
Поддержка ассоциаций атрибутов	195
Книги, в которые были внесены изменения для нового Начального комплекта OLAP DB2	195

Глава 23. Менеджер хранилищ данных DB2 197

"Менеджер хранилищ данных" следует читать как "Менеджер хранилищ данных DB2"	197
Утилита инициализации менеджера каталогов данных	197
Вопросы лицензирования	197
Вопросы установки	198
Менеджер каталогов данных для Web Information Catalog Manager Administration Guide	198
Доступ к каталогам данных DB2 Версии 5 при помощи менеджера каталогов данных DB2 Версии 7.1	198
Конфигурирование каталога данных	199
Обмен метаданными с другими продуктами	199
Обмен метаданными при помощи командами flgnxoln	200
Обмен метаданными при помощи команды MDISDGC	200
Вызов программ	200
Information Catalog Manager Programming Guide and Reference	200
Коды причин Менеджера каталогов данных	200
Information Catalog Manager User's Guide	201
Менеджер каталогов данных: Сообщения	201

Сообщение FLG0260E.	201
Сообщение FLG0051E.	201
Сообщение FLG0003E.	201
Сообщение FLG0372E.	201
Сообщение FLG0615E.	201
Менеджер каталогов данных: Электронная справка	201
Руководство администратора Query Patroller	202
Клиент DB2 Query Patroller - это отдельный компонент	202
Установка Query Patroller вручную в AIX и Solaris	202
Перенастройка из Версии 6 DB2 Query Patroller с использованием dqpmigrate	206
Включение управления запросами	207
Запуск QueryAdministrator	207
Управление пользователями	207
Создание очереди заданий	207
Использование интерфейса командной строки	207
Замечания по Query Enabler	208
DB2 Query Patroller Tracker может вернуть пустую страницу столбцов	208
Приложение В. Troubleshooting DB2 Query Patroller Clients (Устранение неисправностей клиентов DB2 Query Patroller)	208

Часть 7. Разработка прикладных программ 209

Глава 24. Administrative API Reference 213

db2ArchiveLog (новый API)	213
db2ArchiveLog	214
db2ConvMonStream.	216
db2DatabasePing (новый API)	216
db2DatabasePing - проверка связи с базой данных	217
db2HistData	218
db2HistoryOpenScan	219
db2XaGetInfo (новый API)	220
db2XaGetInfo - Получить информацию для менеджера ресурсов	221
db2XaListIndTrans (новый API, заменяющий sqlxphqr)	221
db2XaListIndTrans - Получить список неоднозначных транзакций	222
db2GetSnapshot – получить снимок	225
sqlaintr – Получить сообщение об ошибке	225
sqlbctcq – Закрыть запрос к контейнеру табличного пространства	225
sqlubkp – Резервное копирование базы данных	225
sqlureot – Реорганизовать таблицу.	226
sqlurestore – Восстановить базу данных	226
Ошибка документации о поддержке расширенной совместно используемой памяти AIX.	226
SQLFUPD	226
locklist.	226
SQLEDBDESC	227
Ошибка документации SQLFUPD	227

Глава 25. Application Building Guide 229

Глава 1. Introduction	229
Поддерживаемое программное обеспечение	229

Примеры программ	230
Глава 3. General Information for Building DB2 Applications	231
Файлы построения, make-файлы и утилиты проверки ошибок	231
Глава 4. Building Java Applets and Applications	231
Задание среды	231
Глава 5. Building SQL Procedures	233
Настройка среды процедур SQL	233
Задание переменных среды компилятора	234
Настройка команды компиляции	235
Удаление промежуточных файлов.	238
Резервное копирование и восстановление	239
Создание процедур SQL	239
Вызов хранимых процедур	239
Распространение скомпилированных процедур SQL	240
Глава 7. Building HP-UX Applications.	240
HP-UX C	240
HP-UX C++	241
Глава 10. Building PTX Applications	241
ptx/C++	241
Глава 12. Building Solaris Applications.	241
SPARCCompiler C++.	241
VisualAge C++ Версии 4.0 в OS/2 и Windows.	242

Глава 26. Application Development Guide. 243

Написание хранимых процедур автоматизации OLE	243
Глава 6. Common DB2 Application Techniques	243
Генерирование последовательных значений	243
Глава 7. Stored Procedures	246
Ошибка типа DECIMAL в Java-процедурах Linux	246
Использование указателей в рекурсивных хранимых процедурах	246
Глава 12. Working with Complex Objects: User-Defined Structured Types	247
Вставка в столбцы атрибутов структурированного типа	247
Глава 20. Programming in C and C++	247
Типы C/C++ для хранимых процедур, пользовательских функций и методов	247
Приложение В. Примеры программ	249
Активация встраиваемых модулей проектов и инструментов IBM DB2 Universal Database для Microsoft Visual C++	249
Сигнатура метода Java в процедурах и функциях с PARAMETER STYLE JAVA	249
Провайдер OLE DB IBM DB2	249
Особенности языка/Программирование на Java/Создание прикладных программ и апплетов Java/Поддержка апплетов в Java	249

Глава 27. CLI Guide and Reference 251

Функции Unicode CLI и SQL_C_WCHAR поддерживаются только в AIX	251
Связывание утилит базы данных при использовании клиента времени выполнения	252
Дополнение к разделу "Using Compound SQL"	252
Глава 3. Using Advanced Features	253

Поддержка указателей с возможностью прокрутки на стороне сервера для OS/390	253
Глава 4. Configuring CLI/ODBC and Running Sample Applications	255
Ключевые слова конфигурации	255
Написание хранимой процедуры в CLI	255
Хранимые процедуры и автоматическое связывание CLI.	255
Дополнение к приложению D "Extended Scalar Functions": функции DAYOFWEEK_ISO() и WEEK_ISO().	255
Приложение K. Using the DB2 CLI/ODBC/JDBC Trace Facility.	256
Использование статического SQL в прикладных программах CLI	256
Ограничения статического профилирования JDBC/ODBC/CLI	256
Исправление описания параметра функции CLI SQLBindFileToParam().	257
SQLNextResult - Связать следующий набор результатов с другим хэндлом оператора	257
Назначение	257
Синтаксис	257
Аргументы функции	257
Использование	257
Коды возврата	258
Диагностика	258
Ограничения.	259
Ссылки	259
Преобразования ADT.	259

Глава 28. Справочник по сообщениям 261

CLI0645E (новое сообщение)	261
CLI0646E (новое сообщение)	261
CLI0647E (новое сообщение)	261
DB21086I (новое сообщение)	261
DB210060E (новое сообщение)	261
DB210061E (новое сообщение)	262
DB210062E (новое сообщение)	262
DB210113E (новое сообщение)	262
DB210114E (новое сообщение)	263
DB210115E (новое сообщение)	263
DB210116E (новое сообщение)	263
DB210117E (новое сообщение)	263
DB210118E (новое сообщение)	263
DB210120E (новое сообщение)	263
DB210200I (новое сообщение)	264
DB210201I (новое сообщение)	264
DBI1793W (новое сообщение)	264
DBI1794E (новое сообщение)	264
DBI1795E (новое сообщение)	264
DBI1796W (новое сообщение)	264
DBI1797I (новое сообщение)	265
DWC13603E (новое сообщение)	265
DWC13700E (новое сообщение)	265
DWC13701E (новое сообщение)	265
DWC13702E (новое сообщение)	265
DWC13703E (новое сообщение)	266
DWC13705E (новое сообщение)	266
DWC13706E (новое сообщение)	266

DWC13707E (новое сообщение)	266
SQL0017N (новое сообщение)	267
SQL0056N (новое сообщение)	267
SQL0057N (новое сообщение)	267
SQL0058N (новое сообщение)	267
SQL0097N (новое сообщение)	267
SQL0224N (новое сообщение)	267
SQL0225N (новое сообщение)	268
SQL0227N (новое сообщение)	268
SQL0228N (новое сообщение)	268
SQL0231W (новое сообщение)	268
SQL0243N (новое сообщение)	269
SQL0244N (новое сообщение)	269
SQL0270N (новый код причины 40)	269
SQL0301N (новый текст объяснения)	269
SQL0303N (новый текст)	270
SQL0336N (новое сообщение)	270
SQL0348N (новое сообщение)	270
SQL0349N (новое сообщение)	271
SQL0357N (новый код причины 3)	271
SQL0358N (новое действие пользователя 26)	271
SQL0368N (новое сообщение)	271
SQL0408N (новый текст)	272
SQL0423N (исправленный текст)	272
SQL0590N (новое сообщение)	272
SQL0670N (исправленный текст)	273
SQL0845N (новое сообщение)	274
SQL0846N (новое сообщение)	274
SQL1179W (исправленный текст)	274
SQL1186N (новое сообщение)	274
SQL1550N (новый SQLCODE)	275
SQL1551N (новый SQLCODE)	275
SQL1552N (новый SQLCODE)	275
SQL1553N (новый SQLCODE)	276
SQL1704N (новые коды причины)	276
SQL2077W (новое сообщение)	276
SQL2078N (новое сообщение)	276
SQL2417N (новое сообщение)	277
SQL2426N (новое сообщение)	277
SQL2571N (новое сообщение)	277
SQL2572N (новое сообщение)	277
SQL2573N (новое сообщение)	278
SQL2574N (новое сообщение)	278
SQL2575N (новое сообщение)	278
SQL2576N (новое сообщение)	279
SQL4942N (новый текст)	279
SQL5012N (новое сообщение)	279
SQL6583N (новое сообщение)	279
SQL20005N (новое сообщение)	280
SQL20117N (изменен код причины 1)	280
SQL20121N (новое сообщение)	280
SQL20133N (новое сообщение)	280
SQL20134N (новое сообщение)	280
SQL20135N (новое сообщение)	280
SQL20143N (новое сообщение)	281
SQL20144N (новое сообщение)	281
SQL20145N (новое сообщение)	281
SQL20146N (новое сообщение)	281
SQL20147N (новое сообщение)	281
SQL20148N (новое сообщение)	282
SQL20153N (новое сообщение)	282

Новые значения SQLSTATE	282
Глава 29. SQL Reference	285
Функции OLAP	285
Процедуры и составные операторы SQL	287
LCASE и UCASE (Unicode)	289
WEEK_ISO	289
Правила именования и неявные спецификаторы имен объектов	289
Queries (Запросы)	290
оператор select/условие fetch-first	290
оператор select/синтаксическая диаграмма	290
Библиотеки, используемые оператором CREATE WRAPPER в Linux	291
Теперь поддерживается возможность изменения ключа разделения	291
Оператор: ALTER TABLE	291
Оператор: CREATE TABLE	291
Оператор: DECLARE GLOBAL TEMPORARY TABLE PARTITIONING KEY (имя-столбца,...)	291
Оператор: SET переходная-переменная	291
Оператор: UPDATE	291
Длинные ключи индекса для баз данных Unicode	291
CREATE TABLE	291
CREATE INDEX	292
ALTER TABLE	292
Включение новых встроенных скалярных функций SQL	292
ABS или ABSVAL	292
MULTIPLY_ALT	293
ROUND	294
Примеры:	295
Language Elements/DATALINK Assignments (Элементы языка/Присваивания DATALINK)	295
Функция ENCRYPT (новая скалярная функция)	296
ENCRYPT	296
Функция DECRYPT_BIN и DECRYPT_CHAR (новая скалярная функция)	298
DECRYPT_BIN и DECRYPT_CHAR	298
Функция GETHINT (новая скалярная функция)	299
GETHINT	299
CREATE SEQUENCE (новый оператор SQL)	300
CREATE SEQUENCE	301
ALTER SEQUENCE (новый оператор SQL)	305
ALTER SEQUENCE	306
GRANT (привилегии для последовательности) (новый оператор SQL)	309
GRANT (привилегии для последовательности)	310
SET ENCRYPTION PASSWORD (новый оператор SQL)	310
SET ENCRYPTION PASSWORD	311
Изменения для оператора ALTER TABLE	311
Изменения раздела Compound SQL (Embedded) (Составные операторы SQL (встроенные))	315
Изменения описания оператора CREATE FUNCTION (с источником или шаблоном)	315
Изменения описания оператора CREATE FUNCTION (скалярная SQL, табличная или строковая)	316
Изменения оператора CREATE METHOD	319
Изменения оператора CREATE TRIGGER	320
Изменения для оператора DROP	322

Изменения для оператора DELETE	323
Изменения для оператора INSERT	324
Изменения для оператора SELECT INTO	325
Изменения для оператора SET переходная-переменная	325
SET переменная	326
Изменения для оператора UPDATE	329
Глава 3. Language Elements (Элементы языка)	329
Expressions (Выражения)	329
Глава 4. Functions (Функции)	333
Процедуры	333
Скалярные функции	336
Глава 6. SQL Statements (Операторы SQL)	340
Составной оператор (динамический)	340
Глава 7 "SQL Procedures" теперь называется "Управляющие операторы SQL"	343
Оператор процедуры SQL	343
Заголовок раздела Compound Statement изменен на Составной оператор процедуры	345
Оператор FOR	345
Оператор RETURN	347
Оператор SIGNAL	349
Приложение D: Catalog Views (Производные таблицы каталога)	352
SYSCAT.SEQUENCES	352

Глава 30. Построитель хранимых процедур DB2 355

Поддержка Java 1.2 для построителя хранимых процедур DB2	355
Удаленная отладка хранимых процедур DB2	356
Построение процедур SQL на платформах Windows, OS/2 или UNIX	356
Использование построителя хранимых процедур DB2 на платформе Solaris	356
Известные проблемы и ограничения	356
Использование построителя хранимых процедур DB2 с традиционным китайским языком	357
Установки в UNIX (AIX, Sun Solaris, Linux) и построитель хранимых процедур	357
Построение хранимых процедур SQL в OS/390	357
Отладка хранимых процедур SQL	357
Экспорт хранимых процедур Java	357
Вставка хранимых процедур в OS/390	358
Задание опций построения для хранимых процедур SQL на сервере рабочих станций	358
Автоматическое обновление адресного пространства WLM для построения хранимых процедур в OS/390	359

Глава 31. Обновления для Unicode 361

Введение	361
Базы данных Unicode и прикладные программы DB2	361
Обновление документации	363
SQL Reference	364
Глава 3. Language Elements	364
Глава 4. Functions	368
CLI Guide and Reference	368
Глава 3. Using Advanced Features	368
Приложение C. DB2 CLI and ODBC	373

Data Movement Utilities Guide and Reference	374
Приложение C. Export/Import/Load Utility File Formats	374

Часть 8. Соединение с хост-системами 375

Глава 32. Дополнение по возможностям соединений 377

Настройка сервера прикладных программ в среде VM.	377
Параметры PATCH1 и PATCH2 и конфигурации CLI/ODBC/JDBC	377

Часть 9. Общие сведения 381

Глава 33. Общие сведения 383

DB2 Universal Database Business Intelligence Quick Tour	383
DB2 Everywhere называется теперь DB2 Everyplace	383
Сообщения об ошибках при попытке запуска Netscape	384
Необходима мышь	384
Требования к конфигурации для Adobe Acrobat Reader в системах на основе UNIX.	385
При попытке связывания из клиента времени выполнения DB2 возникает ошибка "Не найдены файлы связывания"	385
Поиск SEARCH.	385
Окна памяти для HP-UX 11	385
Справочник SQL поставляется в виде одного файла PDF	386
Действия пользователя при возникновении ошибки dlfm client_conf	387
В редких случаях, когда демон копирования не останавливается командой dlfm stop	387
Деинсталляция Client Enabler DFS DB2	387
Аутентификация клиента в Windows NT	387
AutoLoader может зависать при ветвлении потоков	388
Восстановление столбцов DATALINK	388
Определите ID пользователя и пароль в IBM Communications Server для Windows NT (CS/NT)	388
Определение узла	388
Ограничения системы объединения	389
Ограничение DataJoiner	389
Менеджер каталогов данных на иврите для Windows NT	389
Microsoft SNA Server и поддержка многоузлового изменения SNA (двухфазного принятия).	390
Ошибка запуска SPM SNA DB2 после загрузки Windows	390
Национальные настройки для сервера администратора DB2	391
Клавиши быстрого вызова не работают	391

Требования к учетным записям для DB2 в Windows NT и Windows 2000	391
Потеря привилегии EXECUTE для пользователей Query Patroller, созданных в версии 6	392
Ограничения Query Patroller	392
Необходимость принятия всех пользовательских программ, которые будут использованы в Центре хранилищ данных (Data Warehouse Center - DWC)	392
Новая опция команды экспорта Центра хранилищ данных	392
API служб резервного копирования (XBSA).	393
Агент OS/390	393
Обзор установки	394
Подробности установки	394
Настройка дополнительных функций агента	397
Составление расписания шагов хранилища при помощи программы триггера (XTClient).	398
Преобразователи	399
Доступ к другим базам данных (не из семейства DB2)	403
Запуск утилит DB2 for OS/390	404
Репликация	406
Регистрация агента	407
Кэширование на клиенте в Windows NT	408
Пробные программные продукты на компакт-дисках с Enterprise Edition для UNIX	408
Пробные программные продукты на компакт-дисках с DB2 Connect Enterprise Edition для UNIX	409
Отбрасывание Менеджера связей данных	409
Деинсталляция компонентов DLFM с использованием SMIT может удалить дополнительные наборы файлов	410
Ошибка SQL1035N при использовании CLP в Windows 2000	410
Усовершенствование SQL Assist	410
Сообщение об ошибке Windows Millennium Edition (ME) и поисковый сервер NetQuestion.	410

Часть 10. Дополнительная информация 413

Глава 34. Дополнительная информация 415

Электронная поддержка DB2 Universal Database и DB2 Connect	415
DB2 Magazine	415

Часть 11. Приложения 417

Приложение. Замечания 419

Товарные знаки.	421
-------------------------	-----

Индекс 423

Добро пожаловать в систему DB2 Universal Database Версии 7.1!

Примечание: Для облегчения чтения Замечаний по выпуску используйте моноширинный шрифт.

Для вашего удобства PDF-версию Замечаний по выпуску FixPak 3 можно получить там же, откуда вы загрузили этот FixPak. Посмотрите Web-сайт:

<http://www-4.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/download.d2w/version71>

Этот файл содержит информацию, которая не была доступна при подготовке руководств DB2, о следующих продуктах:

- IBM DB2 Universal Database Personal Edition, Версия 7.1
- IBM DB2 Universal Database Workgroup Edition, Версия 7.1
- IBM DB2 Universal Database Enterprise Edition, Версия 7.1
- IBM DB2 Data Links Manager, Версия 7.1
- IBM DB2 Universal Database Enterprise - Extended Edition, Версия 7.1
- IBM DB2 Query Patroller, Версия 7.1
- IBM DB2 Personal Developer's Edition, Версия 7.1
- IBM DB2 Universal Developer's Edition, Версия 7.1
- IBM DB2 Data Warehouse Manager, Версия 7.1
- IBM DB2 Relational Connect, Версия 7.1

Для следующих продуктов поставляется отдельный файл Замечаний по выпуску READCON.TXT:

- IBM DB2 Connect Personal Edition, Версия 7.1
- IBM DB2 Connect Enterprise Edition, Версия 7.1

В книге "Что нового" содержится обзор основных усовершенствований DB2 в Версии 7.1 и подробное описание новых возможностей и улучшений.

Часть 1. Информация по установке

Глава 1. Информация по установке

Установка наборов файлов для таблиц преобразования DBCS и Unicode в AIX

В FixPaks 2, 3 и 4 для DB2 Universal Database on AIX в установочном образе есть дополнительный подкаталог, содержащий таблицы преобразования для DBCS и Unicode. Ниже приводятся указания по установке этих FixPaks с наборами файлов таблиц преобразования для DBCS и Unicode.

Установите выбранные обновления при помощи SMIT или `installp`. Установка FixPak состоит из двух шагов. На первом шаге устанавливаются файлы `.bff`. На втором шаге наборы файлов устанавливаются в подкаталог `extras`, находящийся в том же каталоге, что и файлы `.bff`.

Примечание: Чтобы посмотреть дополнительную информацию о команде `installp` или SMIT в электронной документации, введите `'man installp'` или `'man smit'`.

Чтобы получить список опций установки в этом FixPak, введите:

```
installp -ld устройство
```

где *устройство* - имя устройства, где находится образ обновления.

Чтобы установить файлы `.bff` при выполнении всех предварительных условий применения, выберите в SMIT следующую опцию меню:

```
smit update_all
```

Если вы хотите в дальнейшем иметь возможность деинсталляции этого FixPak, убедитесь, что в SMIT выбраны следующие опции:

```
COMMIT software updates = NO  
SAVE replaced files = YES
```

Далее вы сможете при помощи SMIT принять изменения или отказаться от них.

В образе установки этого FixPak есть дополнительный подкаталог. Он называется `extras`. Подкаталог `extras` содержит наборы файлов для таблиц преобразования DBCS и Unicode, а именно:

- `db2_07_01.cnvucs`
- `db2_07_01.conv.jp`
- `db2_07_01.conv.kr`
- `db2_07_01.conv.sch`
- `db2_07_01.conv.tch`

Чтобы установить эти наборы файлов в подкаталог `extras`, выберите в SMIT следующие опции меню:

```
smit install_all
```

Примечание: При установке подкаталога `extras` будут автоматически наборы файлов для таблиц преобразования DBCS и Unicode. Если вы установили подкаталог `extras`, отказаться позже от установки этого FixPak уже нельзя.

Получить пакеты FixPak для DB2 Universal Database можно по адресу:
<http://www-4.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support>

Часть 2. Особые замечания

Глава 2. Особые замечания

Специальные возможности DB2 UDB Версии 7.1

Продукты семейства DB2 UDB включают ряд возможностей, делающих их более доступными для людей с физическими недостатками. Эти возможности включают:

- Возможности, облегчающие ввод с клавиатуры и использование клавиатуры для навигации
- Возможности, улучшающие свойства экрана
- Возможность выбора звуковых и визуальных предупреждений
- Совместимость с технологиями для людей с физическими недостатками
- Совместимость со специальными возможностями операционной системы
- Удобные форматы документации

Ввод с клавиатуры и навигация

Ввод с клавиатуры

С Центром управления DB2 можно работать, используя только клавиатуру. Для пунктов меню и элементов управления есть клавиши доступа, позволяющие пользователям активировать элемент управления или выбрать пункт меню с помощью клавиатуры. Соответствующие этим клавишам буквы выделяются в названиях элементов управления и пунктов меню подчеркиванием.

Фокус ввода с клавиатуры

В системах на основе UNIX положение фокуса ввода с клавиатуры выделяется на экране; тем самым указывается активная область окна, в которую будут вводиться символы при нажатии клавиатуры.

Средства облегчения работы с экраном

В Центре управления DB2 есть ряд средств, улучшающих пользовательский интерфейс и облегчающих работу для пользователей со слабым зрением. В них входят поддержка параметров высококонтрастного вывода на экран и настраиваемых свойств шрифтов.

Высококонтрастный режим

Интерфейс Центра управления поддерживает обеспечиваемый операционной системой высококонтрастный режим. Это облегчает работу пользователей, которым требуется большая степень контрастности между цветами фона и рисунка.

Параметры шрифтов

Интерфейс Центра управления позволяет пользователям выбрать цвет, размер и тип шрифта, используемого в меню и для диалоговых окон.

Независимость от цвета

Чтобы использовать любые функции этого продукта, пользователям не требуется различать цвета.

Альтернативные средства предупреждения

Пользователь может выбрать, в каком виде получать предупреждения: в виде звуковых или визуальных сигналов.

Совместимость с технологиями для людей с физическими недостатками

Интерфейс Центра управления DB2 совместим с программами чтения с экрана (такими, как Via Voice). В режиме прикладной программы интерфейс Центра управления обладает необходимыми свойствами, позволяющими таким программам облегчения работы получать информацию с экрана и делать ее доступной для слепых пользователей.

Удобный формат документации

Документация для продуктов семейства DB2 доступна в формате HTML. Это позволяет пользователям просматривать документацию, используя предпочтения экрана, заданные для их браузеров. Это позволяет также использовать программы чтения с экрана и другие технологии для людей с физическими недостатками.

Поддерживаемые браузеры Web в операционной системе Windows 2000

В Windows 2000 рекомендуется использовать Microsoft Internet Explorer.

При использовании Netscape учтите следующее:

- При использовании Netscape в Windows 2000 для поиска в электронной информации DB2 может потребоваться длительное время. Netscape будет использовать все доступные ресурсы процессора и может работать очень долго. Хотя в конце концов результаты поиска могут быть возвращены, рекомендуется после запуска поиска изменить фокус, щелкнув по другому окну. Через некоторое время будут возвращены результаты поиска.
- Вы можете также заметить, что когда в Центре управления запрошена справка, она правильно выводится в окне браузера Netscape, однако если оставить окно браузера открытым и впоследствии запросить справку из другой части Центра управления, содержимое окна браузера не изменится. Если закрыть окно браузера и снова запросить справку, она будет показана правильно. Эту ошибку можно исправить, выполнив шаги, описанные в разделе “Сообщения об ошибках при попытке запуска Netscape” на стр. 384. Эту проблему можно также обойти, закрывая окно браузера перед запросом справки Центра управления.
- При запросе справки Центра управления или темы из Информационного центра вы можете получить сообщение об ошибке. Чтобы исправить эту ошибку, выполнив шаги, описанные в разделе “Сообщения об ошибках при попытке запуска Netscape” на стр. 384.

Дополнительный необходимый уровень исправления для Solaris

Для DB2 Universal Database Версии 7.1 для Solaris Версии 2.6 кроме исправлений, перечисленных в руководстве *DB2 for UNIX Быстрый старт*, требуется исправление 106285-02 или более новое.

Поддерживаемые процессоры для DB2 Версии 7.1 для Solaris

Не поддерживаются версии процессоров, более ранние, чем UltraSparc.

Поиск электронной информации DB2 в Solaris

При возникновении ошибок поиска электронной информации DB2 в системе Solaris проверьте параметры ядра в файле `/etc/system`. Здесь показаны минимальные значения параметров ядра, требуемые для системы поиска DB2 NetQuestion:

```
semsys:seminfo_semmni 256
semsys:seminfo_semmap 258
semsys:seminfo_semmns 512
semsys:seminfo_semmnu 512
semsys:seminfo_semmns1 50
shmsys:shminfo_shmmax 6291456
shmsys:shminfo_shmseg 16
shmsys:shminfo_shmmni 300
```

Чтобы задать параметр ядра, добавьте в конец файла `/etc/system` такую строку:

```
set <имя_семафора> = значение
```

Чтобы новые или измененные значения вступили в силу, необходимо перезагрузить систему.

Проблемы при добавлении узлов в многораздельную базу данных

При добавлении узлов в многораздельную базу данных, содержащую одно или несколько системных временных табличных пространств, размер страниц которых отличается от размера страниц по умолчанию (4 Кбайта), вы можете получить сообщение об ошибке “SQL6073N Операция Add Node прервана” и SQLCODE. Это вызвано тем, что при создании узла существует только пул буферов IBMDEFAULTBP с размером страниц 4 Кбайта.

Например, для добавления узла в текущую многораздельную базу данных может использоваться команда **db2start**:

```
DB2START NODENUM 2 ADDNODE HOSTNAME newhost PORT 2
```

Если эта многораздельная база данных содержит системные временные табличные пространства, размер страниц которых равен размеру страниц по умолчанию, будет возвращено следующее сообщение:

```
SQL6075W Операция Start Database Manager успешно добавила узел.
Узел не будет активен, пока все узлы не будут остановлены и запущены еще раз.
```

Однако если системные временные табличные пространства этой многораздельной базы данных имеют размер страниц, отличный от размера страниц по умолчанию, будет возвращено сообщение:

```
SQL6073N Операция Add Node прервана. SQLCODE = "<-902>"
```

Для добавления узла можно также использовать команду **ADD NODE**, предварительно вручную добавив описание нового узла в файл `db2nodes.cfg`. После изменения этого файла и выполнения команды **ADD NODE** для многораздельной базы данных, содержащей системные временные табличные пространства, размер страниц которых равен размеру страниц по умолчанию, будет возвращено следующее сообщение:

```
DB20000I Команда ADD NODE выполнена успешно.
```

Однако если системные временные табличные пространства этой многораздельной базы данных имеют размер страниц, отличный от размера страниц по умолчанию, будет возвращено сообщение:

```
SQL6073N Операция Add Node прервана. SQLCODE = "<-902>"
```

Один из способов предотвратить описанную выше ошибку - выполнить команду:

```
DB2SET DB2_HIDDENBP=16
```

перед выполнением команды **db2start** или ADD NODE. Эта переменная реестра разрешает DB2 выделять скрытые пулы буферов по 16 страниц каждый, используя размер страницы, отличный от размера страницы по умолчанию. Это обеспечивает успешное выполнение операции ADD NODE.

Другой способ предотвращения этой ошибки - задать условие WITHOUT TABLESPACES в команде ADD NODE или **db2start**. После этого нужно будет создать пулы буферов, используя оператор CREATE BUFFERPOOL, и связать системные временные табличные пространства с пулом буферов, используя оператор ALTER TABLESPACE.

При добавлении узлов в существующую группу узлов, содержащую одно или несколько табличных пространств, размер страниц которых отличается от размера страниц по умолчанию (4 Кбайта), вы можете получить сообщение об ошибке "SQL0647N Пул буферов "" в настоящее время неактивен.". Это происходит потому, что созданные на новом узле пулы буферов с размером страниц, отличным от размера страниц по умолчанию, не активируются для этих табличных пространств.

Например, для добавления узла в группу узлов можно использовать оператор ALTER NODEGROUP:

```
DB2START  
CONNECT TO mpp1  
ALTER NODEGROUP ng1 ADD NODE (2)
```

Если эта группа узлов содержит табличные пространства, размер страниц которых равен размеру страниц по умолчанию, будет возвращено следующее сообщение:

```
SQL1759W Для изменения распределения данных для объектов в группе узлов  
"<ng1>" требуется перераспределение группы узлов  
со включением добавленных узлов или исключением отброшенных.
```

Однако если табличные пространства этой группы узлов имеют размер страниц, отличный от размера страниц по умолчанию, будет возвращено сообщение:

```
SQL0647N Пул буферов "" в настоящее время неактивен.
```

Один из способов предотвратить описанную выше ошибку - создать пулы буферов для каждого размера страниц и затем заново соединиться с базой данных перед выполнением оператора ALTER NODEGROUP:

```
DB2START  
CONNECT TO mpp1  
CREATE BUFFERPOOL bp1 SIZE 1000 PAGESIZE 8192  
CONNECT RESET  
CONNECT TO mpp1  
ALTER NODEGROUP ng1 ADD NODE (2)
```

Другой способ предотвращения этой ошибки - выполнить команду:

```
DB2SET DB2_HIDDENBP=16
```

перед выполнением команды **db2start** и операторов CONNECT и ALTER NODEGROUP.

Другая ошибка может возникнуть, когда оператор ALTER TABLESPACE используется для добавления табличного пространства к узлу. Например:

```
DB2START
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2) WITHOUT TABLESPACES
ALTER TABLESPACE ts1 ADD ('ts1') ON NODE (2)
```

Эта последовательность команд и операторов генерирует сообщение об ошибке SQL0647N (а не ожидаемое сообщение SQL1759W).

Чтобы правильно выполнить это изменение, необходимо заново соединиться с базой данных после выполнения оператора ALTER NODEGROUP... WITHOUT TABLESPACES.

```
DB2START
CONNECT TO mpp1
ALTER NODEGROUP ng1 ADD NODE (2) WITHOUT TABLESPACES
CONNECT RESET
CONNECT TO mpp1
ALTER TABLESPACE ts1 ADD ('ts1') ON NODE (2)
```

Другой способ предотвращения этой ошибки - выполнить команду:

```
DB2SET DB2_HIDDENBP=16
```

перед выполнением команды **db2start** и операторов CONNECT, ALTER NODEGROUP и ALTER TABLESPACE.

Ошибки при перенастройке

Во время перенастройки в файле db2diag.log появляются записи об ошибках (база данных не перенастроена), даже если перенастройка выполнена успешно; эти сообщения можно проигнорировать.

Исправление китайской национальной версии для Red Flag Linux

Если вы используете сервер Red Flag Linux Версии 1.1 с упрощенным китайским языком, получите у Red Flag исправление национальной версии для упрощенного китайского. Без исправления национальной версии для упрощенного китайского для Версии 1.1 DB2 не узнает, что для упрощенного китайского используется кодовая страница 1386.

Программа установки DB2 может зависнуть, если не подключен съемный диск

Программа установки DB2 может зависнуть после выбора типа установки, если используется компьютер со съемным диском, который не подключен. Чтобы исправить эту ошибку, запустите программу setup с опцией -a:

```
setup.exe -a
```

Дополнительный параметр национальной версии для DB2 for Linux в средах Linux с японским и упрощенным китайским языком

Чтобы в системе Linux с японским или упрощенным китайским языком можно было использовать инструменты GUI Java, требуется дополнительный параметр национальной версии. Без этого параметра не могут правильно выводиться японские и китайские символы. Включите в профиль пользователя следующую команду или выполняйте ее из командной строки перед каждым запуском Центра управления.

Для систем с японским языком:
export LC_ALL=ja_JP

Для систем с упрощенным китайским языком:
export LC_ALL=zh_CN

Ошибка Центра управления в Microsoft Internet Explorer

Существует ошибка, вызываемая установками опций защиты Internet Explorer (IE). Центр управления использует jar без подписи, поэтому доступ к системной информации отключается менеджером защиты.

Чтобы устранить эту ошибку, переконфигурируйте опции защиты IE следующим образом:

1. Выберите **Свойства обозревателя** в меню **Вид** (IE4) или в меню **Сервис** (IE5).
2. На странице Безопасность выберите **Зона надежных узлов** (IE4) или **Надежные узлы** (IE5).
3. Нажмите кнопку **Добавить узлы** (IE4) или **Узлы** (IE5).
4. Добавьте к списку надежных узлов Web-сервер Центра управления. Если Web-сервер Центра управления находится в том же самом домене, может оказаться полезным добавить только имя Web-сервера (без имени домена).
Например:

```
http://ccWebServer.ccWebServerDomain  
http://ccWebServer
```
5. Нажмите кнопку **ОК**.
6. Выберите **Особый** и нажмите кнопку **Настройка** (IE4) или нажмите кнопку **Другой** (IE5).
7. Найдите **Язык Java** → **Разрешения Java** (IE4) или **Java** → **Java Permissions** (IE5) и выберите **Специальная** (IE4) или **Custom** (IE5).
8. Нажмите кнопку **Правила для языка Java**.
9. Выберите страницу Разрешения правки (Edit Permissions).
10. Найдите Неподписанное содержимое → Запустить неподписанное содержимое → Дополнительные неподписанные разрешения → Системные сведения (Unsigned Content → Run Unsigned Content → Additional Unsigned Permissions → System Information) и выберите **Включить (Enable)**.
11. Нажмите кнопку **ОК** в каждом открытом окне.

Несовместимость между DB2 и Sybase в среде Windows

Установка DB2 версии 7.1 на том же компьютере с Windows NT или Windows 2000, что и Sybase Open Client, приведет к ошибке, и Sybase Utilities перестанут работать. Будет выдано сообщение об ошибке, подобное следующему:

```
Fail to initialize LIBTCL.DLL. Please make sure the SYBASE environment variable is set correctly.
```

Это можно предотвратить, удалив параметр среды LC_ALL из параметров среды Windows. LC_ALL - параметр локальной категории. Локальные категории - константы, используемые процедурами локализации для указания того, какую часть информации о локали использовать для программы. Слово *локаль* означает здесь территорию (или страну), для которой возможна настройка определенных аспектов программы. К локально-зависимым областям относится, например, форматирование дат или формат отображения денежных единиц. LC_ALL влияет на все локально-зависимое поведение (всех категорий).

Если удалить параметр среды LC_ALL с целью совместной работы DB2 и Sybase на платформе Windows, перестанут работать следующие функции DB2:

- Пользователь каталога данных
- Администратор каталога данных
- Менеджер каталогов данных

Потеря функции Центра управления

С устаревшими клиентами Центра управления при применении FixPak 2 или FixPak 3 к серверу DB2 не должно быть никаких проблем. Однако в DB2 Версии 7.2 устаревшие клиенты Центра управления почти полностью теряют работоспособность. "Предыдущие уровни" в данном случае подразумевают любые клиенты Версии 6 с FixPak ранее 6 и любые клиенты Версии 7 с FixPak ранее 2. Клиентов Версии 5 это не касается.

Для исправления рекомендуем обновить все затронутые клиенты. Клиенты версии 6 должны быть обновлены до FixPak 6 или более нового, а клиенты версии 7 - до FixPak 2 или более нового.

Часть 3. Установка и конфигурирование

Глава 3. Общая информация по установке . . . 17

Загрузка установочных пакетов для всех поддерживаемых клиентов DB2	17
Установка DB2 в Windows 2000	17
Проблемы перенастройки производных таблиц, в определении которых использованы специальные регистры	17
Поддержка протокола IPX/SPX в Windows 2000.	17
Остановка процессов DB2 перед обновлением предыдущей версии DB2	18
После установки DB2 выполните db2iupdt, если уже установлен другой продукт DB2.	18
Настройка среды Linux для работы DB2	18
DB2 Universal Database Enterprise Edition и DB2 Connect Enterprise Edition для Linux на S/390	19
DB2 Universal Database Enterprise - Extended Edition теперь доступна для Linux.	19
Параметр ядра shmseg для HP-UX	19

Глава 4. Data Links Manager Quick Beginnings 21

Ошибка запуска Dlfm с сообщением: "Ошибка получения afsfid для префикса"	21
Задание класса Tivoli Storage Manager для файлов архивов.	21
Требования к дисковому пространству для DFS Client Enabler	21
Мониторинг процессов внутренней обработки менеджера файлов связей данных в AIX	21
Установка и конфигурирование менеджера связей данных DB2 для AIX: Особенности установки в средах DCE-DFS	22
Ошибка выполнения команды "dlfm add_prefix".	22
Установка и конфигурирование менеджера связей данных DB2 для AIX: Использование утилиты db2setup для установки менеджера связей данных DB2 в AIX	22
Установка и конфигурирование менеджера связей данных DB2 для AIX: Задачи после установки в DCE-DFS	23
Установка и конфигурирование менеджера связей данных DB2 для AIX: Ручная установка менеджера связей данных DB2 с помощью SMIT	23
Установка и конфигурирование Client Enabler DFS для связей данных DB2	23
Установка и конфигурирование менеджера связей данных DB2 для Solaris.	24
Выбор метода резервного копирования для менеджера связей данных DB2 в AIX	24
Выбор метода резервного копирования для менеджера связей данных DB2 в Windows NT	25
Резервное копирование файловой системы с журналами в AIX	27
Привилегии группы администратора в связях данных в Windows NT.	27

Минимизация записи в журнал при установке фильтра файловой системы DataLinks (DataLinks File System Filter - DLFF)	28
Запись сообщений в журнал после установки.	29
Деинсталляция компонентов DLFM с использованием SMIT может удалить дополнительные наборы файлов Before You Begin/Determine hostname (Прежде, чем вы начнете/Определение имени хоста)	29
Работа с менеджером файлов связей данных: очистка после удаления менеджера связей данных DB2 из базы данных DB2	31
DLFM1001E (новое сообщение об ошибке)	31
Опция задания файла конфигурации DLFM	32
Ошибка при работе сценария связей данных/DFS dmapp_prestart в AIX	32
Глава 4. Installing and Configuring DB2 Data Links Manager for AIX (Установка и конфигурирование менеджера связей данных DB2 для AIX)	32
Общие замечания по установке	32
Перенастройка менеджера файлов DB2 Версии 5.2 в Менеджер связей данных DB2 Версии 7.	32
Приложение A. DB2 Data Links Manager Errors and User Responses (Ошибки менеджера связей данных DB2 и действия пользователя)	32

Глава 5. Дополнение по установке и настройке 33

Связывание утилит базы данных при использовании клиента времени выполнения.	33
Доступ к DB2 с использованием ODBC из клиентов UNIX	33
Переключение NetQuestion for OS/2 на использование TCP/IP	33
Глава 26. Настройка системы объединения на работу с источниками данных Oracle.	33

Глава 3. Общая информация по установке

Загрузка установочных пакетов для всех поддерживаемых клиентов DB2

Установочные пакеты для всех поддерживаемых клиентов DB2, в том числе для клиентов DB2 более ранних версий (до версии 7.1), можно загрузить с сайта IBM DB2 Client Application Enabler Pack по адресу:
<http://www.ibm.com/software/data/db2/db2tech/clientpak.html>

Установка DB2 в Windows 2000

В Windows 2000 при установке поверх более ранней версии DB2 или повторной установке текущей версии убедитесь, что для опций восстановления для всех служб DB2 задано значение "Не выполнять никаких действий".

Проблемы перенастройки производных таблиц, в определении которых использованы специальные регистры

После перенастройки базы данных нельзя использовать производные таблицы, для определения столбцов которых использованы специальные регистры USER или CURRENT SCHEMA. Например:

```
create view v1 (c1) as values user
```

В Версии 5 специальные регистры USER и CURRENT SCHEMA имели тип данных CHAR(8), но начиная с Версии 6 они определены как VARCHAR(128). Если в этом примере производная таблица создана в Версии 5, столбец c1 имеет тип данных CHAR, и после перенастройки базы данных он также будет иметь тип данных CHAR. Если эта производная таблица используется после перенастройки, этот столбец будет вычисляться во время выполнения и это вызовет ошибку из-за несовпадения типов данных.

Решение проблемы: отбросить и заново создать эту производную таблицу. Перед ее отбрасыванием сохраните оператор, использованный для ее создания, запросив производную таблицу каталога SYSCAT.VIEWS. Например:

```
select text from syscat.views where viewname='<>'
```

Поддержка протокола IPX/SPX в Windows 2000

Опубликованная схема поддержки протокола не вполне правильна. Не поддерживается соединение клиента Windows 2000 с сервером в OS/2 или системе на основе UNIX при помощи IPX/SPX. Также не поддерживается соединение клиента OS/2 или клиента в системе на основе UNIX с сервером Windows 2000 при помощи IPX/SPX.

Остановка процессов DB2 перед обновлением предыдущей версии DB2

При обновлении предыдущей версии DB2 на компьютере с системой Windows программа установки выдает сообщение, содержащее список процессов, удерживающих в памяти модули DLL DB2. После этого вы можете вручную остановить процессы, указанные в этом списке, или можете позволить программе установки автоматически завершить эти процессы. Чтобы избежать потери данных, рекомендуется вручную завершить все процессы DB2 перед установкой. Лучший способ убедиться, что процессы DB2 не выполняются - проверить запущенные в системе процессы, используя панель Windows Службы. В панели Windows Службы убедитесь, что не запущены службы DB2, службы OLAP или службы хранилища данных.

Примечание: Одновременно в системе Windows может работать только одна версия DB2. Например, нельзя одновременно запустить на одном компьютере Windows DB2 Версии 7.1 и DB2 Версии 6. При установке DB2 Версии 7.1 на компьютере, на котором установлена DB2 Версии 6, программа установки удалит в процессе установки DB2 Версии 6. Дополнительную информацию о перенастройке из предыдущих версий DB2 смотрите в соответствующем руководстве *Быстрый старт*.

После установки DB2 выполните db2iupdt, если уже установлен другой продукт DB2

При установке DB2 UDB Версии 7.1 в системах на основе UNIX, в которых уже установлен продукт DB2, нужно будет выполнить команду **db2iupdt**, чтобы обновить те экземпляры, которые должны использовать новые возможности этого продукта. Некоторые возможности не будут доступны, пока не будет выполнена эта команда.

Настройка среды Linux для работы DB2

Запустив программу установки DB2 в системе Linux и вернувшись в окно терминала, введите следующие команды, чтобы задать правильную среду для работы Центра управления DB2:

```
su -l <имя экземпляра>
export JAVA_HOME=/usr/jdk118
export DISPLAY=<имя компьютера>:0
```

Затем откройте еще одно окно терминала и введите команды:

```
su root
xhost +<имя компьютера>
```

Закройте это окно терминала и вернитесь в окно терминала, в котором вы зарегистрировались как владелец экземпляра, и введите команду:

```
db2cc
```

чтобы запустить Центр управления.

DB2 Universal Database Enterprise Edition и DB2 Connect Enterprise Edition для Linux на S/390

DB2 Universal Database Enterprise Edition и DB2 Connect Enterprise Edition теперь доступны для Linux на S/390. Перед установкой Linux на компьютер S/390 необходимо ознакомиться с требованиями к программному обеспечению и аппаратуре:

Аппаратура

S/390 9672 Generation 5 или более новое, Multiprise 3000.

Программное обеспечение

- SuSE Linux v7.0 for S/390
- уровень ядра 2.2.16 с исправлениями для S/390 (смотрите ниже)
- glibc 2.1.3
- libstdc++ 6.1

Для Linux на S/390 требуются следующие исправления:

- в настоящее время исправления не требуются.

Последние исправления смотрите в Интернете по адресу <http://www.software.ibm.com/data/db2/linux>.

Примечания:

1. Поддерживаются только 32-битные Linux для платформы Intel и Linux на S/390.
2. Под Linux/390 в DB2 версии 7.1 *недоступны*:
 - DB2 UDB Enterprise - Extended Edition
 - Модули расширения DB2
 - Менеджер связей данных
 - Клиент управления DB2
 - Поддержка изменения пароля
 - Поддержка LDAP

DB2 Universal Database Enterprise - Extended Edition теперь доступна для Linux.

DB2 Universal Database Enterprise - Extended Edition теперь доступна для Linux.

Примечание: Для Linux EEE у каждого физического узла в кластере EEE должны быть одни и те же уровни ядра, glibc и libstdc++.

Параметр ядра shmseg для HP-UX

Рекомендуемое значение параметра ядра shmseg для HP-UX следует игнорировать.

Вместо этого следует использовать значение HP-UX по умолчанию (120).

Глава 4. Data Links Manager Quick Beginnings

Ошибка запуска Dlfm с сообщением: "Ошибка получения afsfid для префикса"

Обратитесь в Службу поддержки IBM, если для менеджера связей данных, работающего в среде DCE-DFS, возникает ошибка запуска dlfm с сообщением:
Ошибка получения afsfid для префикса

Эта ошибка может возникнуть, если набор файлов DFS, зарегистрированный в менеджере связей данных с помощью "dlfm add_prefix", удален.

Задание класса Tivoli Storage Manager для файлов архивов

Чтобы задать, какой класс управления TSM должен использоваться для файлов архивов, задайте соответствующее имя класса управления в переменной реестра DB2 DLFM_TSM_MGMTCLASS.

Требования к дисковому пространству для DFS Client Enabler

DFS Client Enabler - это дополнительный компонент, который можно выбрать при установке клиента или сервера DB2 Universal Database. Нельзя установить DFS Client Enabler, не устанавливая продукт клиента или сервера DB2 Universal Database, даже если DFS Client Enabler работает сам по себе и не требуются клиент или сервер DB2 UDB. Если DFS Client Enabler устанавливается при установке клиента времени выполнения DB2, кроме 2 Мбайт дискового пространства, необходимых для кода DFS Client Enabler, нужно зарезервировать дополнительные 40 Мбайт. Если DFS Client Enabler устанавливается при установке клиента администратора DB2 или сервера DB2, потребуется больший объем дискового пространства. Дополнительную информацию о требуемом дисковом пространстве для продуктов DB2 Universal Database смотрите в руководстве *DB2 for UNIX Быстрый старт*.

Мониторинг процессов внутренней обработки менеджера файлов связей данных в AIX

Изменены выходные данные команды **dlfm see**. Когда эта команда используется для отслеживания процессов внутренней обработки менеджера файлов связей данных в AIX, возвращаемые ей выходные данные будут выглядеть так:

PID	PPID	PGID	RUNAME	UNAME	ETIME	DAEMON NAME
17500	60182	40838	dlfm	root	12:18	dlfm_copypd_(dlfm)
41228	60182	40838	dlfm	root	12:18	dlfm_chownd_(dlfm)
49006	60182	40838	dlfm	root	12:18	dlfm_upcalld_(dlfm)
51972	60182	40838	dlfm	root	12:18	dlfm_gcd_(dlfm)
66850	60182	40838	dlfm	root	12:18	dlfm_retrieved_(dlfm)
67216	60182	40838	dlfm	dlfm	12:18	dlfm_delgrp_(dlfm)
60182	1	40838	dlfm	dlfm	12:18	dlfmd_(dlfm)

DLFM SEE request was successful.

Имя в скобках - это имя экземпляра dlfm, в данном случае "dlfm".

Установка и конфигурирование менеджера связей данных DB2 для AIX: Особенности установки в средах DCE-DFS

В разделе "Предварительные требования для установки" должна быть добавлена новая информация:

Необходимо также установить или e-fix for DFS 3.1, или PTF set 1 (когда он станет доступным). Это http://www.transarc.com/Support/dfs/datalinks/efix_dfs31_main_page.html

А также:

Перед установкой менеджера связей данных должен быть запущен клиент dfs. Используйте db2setup или smitty.

В разделе "Файл keytab" есть ошибка, которую надо исправить так:

Файл keytab, содержащий информацию о принципах и паролях, должен называться datalink.ktb и....

В примере ниже используется правильное имя datalink.ktb. Раздел "Keytab file" надо переставить ниже "DCE-DFS Post-Installation Task", так как этот файл нельзя создать, пока не будет создан экземпляр DLMADMIN.

В разделе "Data Links File Manager servers and clients" должно быть указано, что сервер менеджера связей данных должен быть установлен до установки клиентов менеджера связей данных.

Должен быть добавлен новый раздел "Каталог резервного копирования":

Если для резервного копирования используется локальная файловая система, каталог должен быть задан. Убедитесь, что администратор DFS создал этот набор файлов DFS. Он не должен быть набором файлов DMLFS.

Ошибка выполнения команды "dlfm add_prefix"

Для менеджера связей данных, работающего в среде DCE/DFS, может возникать ошибка команды **dlfm add_prefix** с кодом возврата -2061 (ошибка создания резервной копии). Если это произошло, выполните следующие действия:

1. Остановите процесс демона менеджера связей данных командой **dlfm stop**.
2. Остановите процессы DB2 командой **dlfm stopdbm**.
3. Получите права пользователя DCE root, выполнив команду **dce_login root**.
4. Запустите процессы DB2 командой **dlfm startdbm**.
5. Зарегистрируйте набор файлов на менеджере связей данных с помощью команды **dlfm add_prefix**.
6. Запустите процесс демона менеджера связей данных командой **dlfm start**.

Установка и конфигурирование менеджера связей данных DB2 для AIX: Использование утилиты db2setup для установки менеджера связей данных DB2 в AIX

В разделе "DB2 database DLFM_DB created" база данных DLFM_DB не создается в среде DCE_DFS. Это необходимо сделать после установки.

В разделе "DCE-DFS pre-start registration for DMAPP" Шаг 2 должен быть изменен на следующий текст:

2. В /opt/dcelocal/tcl/user_cmd.tcl добавлены команды, обеспечивающие запуск DMAPP при запуске DFS.

Установка и конфигурирование менеджера связей данных DB2 для AIX: Задачи после установки в DCE-DFS

Должен быть добавлен следующий новый раздел "Завершение установки менеджера связей данных":

Для завершения установки на сервере связей данных нужно выполнить следующие действия:

1. Создайте файл `keytab`, как это описано в подразделе "Keytab file" раздела "Additional Installation Considerations in DCE-DFS Environment" главы "Installing and Configuring DB2 Data Links Manager for AIX".
2. Зарегистрировавшись как пользователь `root`, введите следующие команды для запуска DMAPP:

```
stop.dfs all
start.dfs all
```
3. Выполните команду `dlfm setup`, используя права пользователя `root DCE`:
 - a. Зарегистрируйтесь как администратор менеджера связей данных `DLMADMIN`.
 - b. В качестве пользователя `root` введите команду `dce_login`.
 - c. Введите команду: `dlfm setup`.

Для завершения установки на клиенте связей данных нужно выполнить следующие действия:

1. Создайте файл `keytab`, как это описано в подразделе "Keytab file" раздела "Additional Installation Considerations in DCE-DFS Environment" главы "Installing and Configuring DB2 Data Links Manager for AIX".
2. Зарегистрировавшись как пользователь `root`, введите следующие команды для запуска DMAPP:

```
stop.dfs all
start.dfs all
```

Установка и конфигурирование менеджера связей данных DB2 для AIX: Ручная установка менеджера связей данных DB2 с помощью SMIT

В разделе "SMIT Post-installation Tasks" измените шаг 7, указав в нем, что команда `"dce_login root"` должна выполняться перед командой `"dlfm setup"`. Шаг 11 не нужен. Этот шаг выполняется автоматически после выполнения шага 6 (`dlfm server_conf`) или 8 (`dlfm client_conf`). Удалите также шаг 12 (`dlfm start`). Для завершения установки выполните следующие действия:

1. Создайте файл `keytab`, как это описано в подразделе "Keytab file" раздела "Additional Installation Considerations in DCE-DFS Environment" главы "Installing and Configuring DB2 Data Links Manager for AIX".
2. Зарегистрировавшись как пользователь `root`, введите следующие команды для запуска DMAPP:

```
stop.dfs all
start.dfs all
```

Установка и конфигурирование Client Enabler DFS для связей данных DB2

В разделе "Configuring a DFS Client Enabler" добавьте в шаг 2 следующую информацию:

Обычно конфигурирование завершается командами "secval".

Однако может также понадобится перезагрузка компьютера.

Если возникли ошибки при обращении к файлам READ PERMISSION DB, перезагрузите компьютер, на котором

Установка и конфигурирование менеджера связей данных DB2 для Solaris

После установки менеджера связей данных DB2 для Solaris необходимо выполнить следующие действия:

1. Добавить к файлу /etc/system следующие три строки:

```
set dlfsdrv:glob_mod_pri=0x100800
set dlfsdrv:glob_mesg_pri=0xff
set dlfsdrv:ConfigDlfsUid=UID
```

где *UID* - ID пользователя dlfm.

2. Перезагрузить компьютер, чтобы изменения вступили в силу.

Выбор метода резервного копирования для менеджера связей данных DB2 в AIX

Для резервного копирования файлов с сервера связей данных кроме копирования на диск и XBSA можно также использовать Tivoli Storage Manager (TSM).

Чтобы в качестве сервера архивов использовать Tivoli Storage Manager:

1. Установите Tivoli Storage Manager на сервере связей данных. Дополнительную информацию смотрите в документации по используемому продукту Tivoli Storage Manager.
2. Зарегистрируйте прикладную программу клиента связей данных на сервере Tivoli Storage Manager. Дополнительную информацию смотрите в документации по используемому продукту Tivoli Storage Manager.
3. В файлы сценариев db2profile или db2cshrc администратора менеджера связей данных добавьте следующие переменные среды:

```
(для оболочки Bash, Bourne или Korn)
export DSMI_DIR=/usr/tivoli/tsm/client/api/bin
export DSMI_CONFIG=${HOME}/tsm/dsm.opt
export DSMI_LOG=${HOME}/dldump
export PATH=${PATH}:${DSMI_DIR}
```

```
(для оболочки C)
setenv DSMI_DIR /usr/tivoli/tsm/client/api/bin
setenv DSMI_CONFIG ${HOME}/tsm/dsm.opt
setenv DSMI_LOG ${HOME}/dldump
setenv PATH=${PATH}:${DSMI_DIR}
```

4. Убедитесь, что файл системных опций TSM dsm.sys находится в каталоге `DSMI_DIR`.
5. Убедитесь, что файл пользовательских опций TSM dsm.opt находится в каталоге `INSTHOME/tsm`, где `INSTHOME` - начальный каталог администратора менеджера связей данных.
6. В файле системных опций Tivoli Storage Manager `/usr/tivoli/tsm/client/api/bin/dsm.sys` задайте для опции `PASSWORDACCESS` значение `generate`.
7. Зарегистрируйте пароль TSM с опцией `generate` перед первым запуском менеджера файлов связей данных. Теперь не нужно будет задавать пароль при

установлении соединения менеджера файлов связей данных с сервером TSM. Дополнительную информацию смотрите в документации по используемому продукту TSM.

8. Для переменной реестра `DLFM_BACKUP_TARGET` задайте значение TSM. В этом случае значение переменной реестра `DLFM_BACKUP_DIR_NAME` будет игнорироваться. Для резервного копирования будет использоваться Tivoli Storage Manager.

Примечания:

- a. Имейте в виду, что если во время работы изменить значение переменной реестра `DLFM_BACKUP_TARGET` с TSM на положение на диске, созданные файлы архивов не будут перемещены в новое положение. Например, если менеджер файлов связей данных был запущен со значением TSM переменной реестра `DLFM_BACKUP_TARGET` и затем значение этой переменной реестра было изменено на положение на диске, вновь создаваемые файлы архивов будут сохраняться в этом новом положении на диске. Файлы архивов, сохраненные ранее в TSM, не будут перемещены в это новое положение на диске.
 - b. Чтобы переопределить класс управления TSM по умолчанию, можно использовать новую переменную реестра `DLFM_TSM_MGMTCLASS`. Если эта переменная реестра не задана, используется класс управления TSM по умолчанию.
9. Остановите менеджер файлов связей данных командой **`dlfm stop`**.
 10. Запустите менеджер файлов связей данных командой **`dlfm start`**.

Выбор метода резервного копирования для менеджера связей данных DB2 в Windows NT

При вставке значения `DATALINK` в таблицу со столбцом `DATALINK`, для которого определено восстановление, в расписание вносится операция резервного копирования на сервере архивов соответствующих файлов `DATALINK` с сервера связей данных. В настоящее время поддерживаются две опции резервного копирования файлов на сервере архивов: копирование на диск (метод по умолчанию) и Tivoli Storage Manager. В будущих выпусках менеджера связей данных DB2 для Windows NT будут поддерживаться другие носители и программы резервного копирования.

Копирование на диск (метод по умолчанию)

Когда на сервере DB2 введена команда **`backup`**, для заданных в базе данных файлов на сервере связей данных создается резервная копия в каталоге, задаваемом переменной среды `DLFM_BACKUP_DIR_NAME`. По умолчанию эта переменная имеет значение `c:\d1 fmbbackup`, где `c:\` обозначен диск резервной установки менеджера связей данных.

Чтобы задать для этой переменной значение `c:\dlfmbbackup`, введите следующую команду:

```
db2set -g DLFM_BACKUP_DIR_NAME=c:\d1 fmbbackup
```

Положение, задаваемое в переменной среды `DLFM_BACKUP_DIR_NAME`, *не* должно определяться в файловой системе с помощью фильтра файловой системы связей данных и каталог, заданный для резервного копирования файлов, должен содержать достаточно свободного пространства.

Задайте также для переменной `DLFM_BACKUP_TARGET` значение `LOCAL`, используя следующую команду:

```
db2set -g DLFM_BACKUP_TARGET=LOCAL
```

После задания или изменения этих переменных остановите и перезапустите менеджер файлов связей данных с помощью команд **dlfm stop** и **dlfm start**.

Tivoli Storage Manager

Чтобы в качестве сервера архивов использовать Tivoli Storage Manager:

1. Установите Tivoli Storage Manager на сервере связей данных. Дополнительную информацию смотрите в документации по используемому продукту Tivoli Storage Manager.
2. Зарегистрируйте прикладную программу клиента связей данных на сервере Tivoli Storage Manager. Дополнительную информацию смотрите в документации по используемому продукту Tivoli Storage Manager.
3. Нажмите кнопку **Пуск** и выберите **Настройка → Панель управления → Система**. Откроется окно Свойства системы. Выберите закладку **Среда** и введите следующие переменные среды и их значения:

Переменная	Значение
DSMI_DIR	c:\tsm\baclient
DSMI_CONFIG	c:\tsm\baclient\dsm.opt
DSMI_LOG	c:\tsm\dlldump

4. Убедитесь, что файл системных опций TSM `dsm.sys` находится в каталоге `c:\tsm\baclient`.
5. Убедитесь, что файл пользовательских опций TSM `dsm.opt` находится в каталоге `c:\tsm\baclient`.
6. В файле системных опций Tivoli Storage Manager `c:\tsm\baclient\dsm.sys` задайте для опции `PASSWORDACCESS` значение `generate`.
7. Зарегистрируйте пароль TSM с опцией `generate` *перед* первым запуском менеджера файлов связей данных. Теперь не нужно будет задавать пароль при установлении соединения менеджера файлов связей данных с сервером TSM. Дополнительную информацию смотрите в документации по используемому продукту TSM.
8. Для переменной среды `DLFM_BACKUP_TARGET` задайте значение TSM, используя следующую команду:

```
db2set -g DLFM_BACKUP_TARGET=TSM
```

В этом случае значение переменной среды `DLFM_BACKUP_DIR_NAME` будет игнорироваться. Для резервного копирования будет использоваться Tivoli Storage Manager.

Примечания:

- a. Имейте в виду, что если во время работы изменить значение переменной среды `DLFM_BACKUP_TARGET` с TSM на LOCAL, созданные файлы архивов не будут перемещены в новое положение. Например, если менеджер файлов связей данных был запущен со значением TSM переменной среды `DLFM_BACKUP_TARGET` и затем значение этой переменной среды было изменено на LOCAL, вновь создаваемые файлы архивов будут сохраняться в этом новом положении на диске. Файлы архивов, сохраненные ранее в TSM, не будут перемещены в это новое положение на диске.

- b. Чтобы переопределить класс управления TSM по умолчанию, можно использовать новую переменную среды `DLFM_TSM_MGMTCLASS`. Если эта переменная не задана, используется класс управления TSM по умолчанию.
9. Остановите менеджер файлов связей данных командой **`dlfm stop`**.
10. Запустите менеджер файлов связей данных командой **`dlfm start`**.

Резервное копирование файловой системы с журналами в AIX

В этой книге сказано, что необходимо остановить менеджер связей данных и выполнить резервное копирование файловой системы в автономном режиме. Для пользователей, которым требуется большая доступность системы, предлагается следующий подход, позволяющий избежать необходимости останавливать менеджер связей данных.

1. Возьмите исходный файл CLI `quiesce.c` и сценарий оболочки `online.sh`. Эти файлы находятся в каталоге `/samples/dlfm`.
2. Скомпилируйте `quiesce.c`:

```
xlc -o quiesce -L$HOME/sqllib/lib --I$HOME/sqllib/include -c quiesce.c
```
3. Выполните этот сценарий на узле с файловой системой DLFS.

В сценарии оболочки `online.sh` предполагается, что на узле менеджера связей данных есть запись каталога для каждой базы данных, зарегистрированной на менеджере связей данных. Подразумевается также, что `/etc/filesystem` содержит полную запись для файловой системы DLFS. Этот сценарий оболочки делает следующее:

- Стабилизирует все таблицы в базах данных, зарегистрированных на менеджере связей данных. Это предотвращает все новые операции с ними.
- Размонтирует файловую систему и вновь монтирует ее как файловую систему только для чтения.
- Выполняет резервное копирование файловой системы.
- Размонтирует файловую систему и вновь монтирует ее как файловую систему только для чтения и записи.
- Выводит таблицы DB2 из состояния стабилизации.

Этот сценарий нужно изменить в соответствии с вашей средой:

1. Выберите команду резервного копирования и задайте ее в функции `do_backup` в этом сценарии.
2. Задайте в этом сценарии следующие переменные среды:
 - В качестве значения `DLFM_INST` задайте имя экземпляра DLFM.
 - В качестве значения `PATH_OF_EXEC` задайте путь выполняемого файла `"quiesce"`.

Запустите этот сценарий:

```
online.sh <имя_файловой_системы>
```

Привилегии группы администратора в связях данных в Windows NT

В Windows NT у пользователя, принадлежащего к группе администратора, должны быть те же самые привилегии в отношении файлов, связанных с использованием DataLinks, что и у пользователя UNIX `root` в отношении большинства функций. Эти два набора привилегий сравниваются в следующей таблице.

Операция	Unix (root)	Windows NT (администратор)
Переименование	Да	Да
Доступ к файлу без маркера	Да	Да
Удаление	Да	Нет (смотрите примечание ниже)
Изменение	Да	Нет (смотрите примечание ниже)

Примечание: NTFS отменяет разрешение этих операций для файлов с доступом только для чтения. Пользователь группы администратора может успешно выполнять эти операции, включив для файла разрешение на запись.

Минимизация записи в журнал при установке фильтра файловой системы DataLinks (DataLinks File System Filter - DLFF)

Можно минимизировать запись в журнал при установке фильтра файловой системы DataLinks (DLFF), изменив файл `dlfs_cfg`. Файл `dlfs_cfg` передается процедуре `strload` для загрузки драйвера и параметров конфигурации. Этот файл расположен в каталоге `/usr/lpp/db2_07_01/cfg/`. Через символическую связь этот файл можно также найти в каталоге `/etc`. Формат файла `dlfs_cfg` следующий:

```
d 'имя-драйвера' 'номер vfs' 'id dlfm' 'глобальный приоритет сообщений'
'глобальный приоритет модулей' - 0 1
```

где:

d Параметр `d` указывает, что драйвер должен быть загружен.

имя-драйвера

Имя-драйвера - полный путь загружаемого драйвера. Например, полный путь для DB2 Версии 7: `/usr/lpp/db2_07_01/bin/dlfsdrv`. Имя драйвера - `dlfsdrv`.

номер vfs

Это запись `vfs` для DLFS в `/etc/vfs`.

id dlfm

Это ID пользователя администратора менеджера связей данных.

глобальный приоритет сообщений

Это глобальный приоритет сообщений

глобальный приоритет модулей

Это глобальный приоритет модулей

0 1 0 1 - младшие цифры для создания неклонированных узлов для этого драйвера. Имена узлов создаются путем присоединения младших цифр к имени клонированного узла драйвера. Может быть задано не больше пяти младших цифр (0-4).

Реальный пример может выглядеть так:

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,255,-1 - 0 1
```

Сообщения, записываемые в журнал, зависят от установок для глобального приоритета сообщений и от глобального приоритета модулей. Для минимизации записи в журнал можно изменить значение для глобального приоритета сообщений.

Для приоритета сообщений можно использовать четыре значения:

```
#define LOG_EMERGENCY    0x01
#define LOG_TRACING      0x02
#define LOG_ERROR        0x04
#define LOG_TROUBLESHOOT 0x08
```

У большинства сообщений в DLFF в качестве приоритета сообщений используется LOG_TROUBLESHOOT. Несколько альтернативных примеров конфигурации:

Если вам необходимы аварийные сообщения и сообщения об ошибках, установите для глобального приоритета сообщений значение 5 (1+4) в файле конфигурации dlfs_cfg:

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,5,-1 - 0 1
```

Если требуются только сообщения об ошибках, установите для глобального приоритета сообщений 4:

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,4,-1 - 0 1
```

Если вам не требуется запись в журнал для DLFS, установите для глобального приоритета сообщений 0:

```
d /usr/lpp/db2_07_01/bin/dlfsdrv 14,208,0,-1 - 0 1
```

Запись сообщений в журнал после установки

При необходимости записывать в журнал сообщения об аварийных ситуациях, ошибках и предупреждениях после установки следует изменить файл dlfs_cfg. Файл dlfs_cfg расположен в каталоге /usr/lpp/db2_07_01/cfg. Для глобального приоритета сообщений должно быть установлено 255 (максимальный приоритет) или 13 (8+4+1). При установке для приоритета 13 (8+4+1) будет записываться информация об аварийных ситуациях, ошибках и предупреждениях.

После установки глобального приоритета сообщений демонтируйте файловую систему фильтра DLFS и перезагрузите драйвер dlfsdrv, чтобы во время загрузки были установлены новые значения приоритета. После перезагрузки драйвера dlfsdrv файловая система фильтра DLFS должна быть перемонтирована.

Примечание: Установки для dlfs_cfg будут оставаться для всех последующих загрузок драйвера dlfsdrv до тех пор, пока файл dlfs_cfg не будет снова изменен.

Деинсталляция компонентов DLFM с использованием SMIT может удалить дополнительные наборы файлов

Перед установкой DB2 (Версии 5, 6 или 7) на компьютер с AIX, на котором установлен менеджер связей данных, выполните следующие шаги:

1. В качестве пользователя root сделайте копию /etc/vfs, используя команду:
cp -p /etc/vfs /etc/vfs.bak
2. Деинсталируйте DB2.
3. В качестве пользователя root замените /etc/vfs на резервную копию, сделанную на шаге 1:
cp -p /etc/vfs.bak /etc/vfs

Before You Begin/Determine hostname (Прежде, чем вы начнете/Определение имени хоста)

Вам надо определить имена для каждого из серверов DB2 и серверов связей данных. Эти имена хостов понадобятся для проверки установки. При соединении с менеджером файлов связей данных (DLFM) DB2 сервер DB2 UDB внутренним образом посылает ему следующую информацию:

- Имя базы данных
- Имя экземпляра
- Имя хоста

Затем DLFM сравнивает эту информацию со своими внутренними таблицами, чтобы определить, разрешать ли соединение. Соединение будет разрешено только в том случае, если сочетание имени базы данных, имени экземпляра и имени хоста зарегистрировано на нем при помощи команды `dlfm add_db`. Имя хоста, которое использовалось в команде `dlfm add_db`, должно точно совпадать с именем хоста, внутренним образом посланным с сервера DB2 UDB.

Используйте точное имя хоста, которое можно узнать так:

1. Введите на вашем сервере DB2 команду `hostname`. Допустим, это команда возвращает имя `db2server`.
2. В зависимости от платформы сделайте следующее:
 - На AIX введите команду `host db2server`, где `db2server` - имя, полученное на предыдущем шаге. Эта команда должна вернуть примерно следующую строку:
`db2server.services.com is 9.11.302.341, Aliases: db2server`
 - В Windows NT введите команду `nslookup db2server`, где `db2server` - имя, полученное на предыдущем шаге. Эта команда должна вернуть примерно следующее:
`Server: dnsserv.services.com
Address: 9.21.14.135
Name: db2server.services.com
Address: 9.21.51.178`
 - В Solaris введите `cat /etc/hosts | grep 'имя-хоста'`. Если имя хоста указано в `/etc/hosts` без имени домена, эта команда должна вернуть примерно следующее:
`9.112.98.167 db2server loghost`

Если имя хоста указано с именем домена, вывод будет выглядеть так:
`9.112.98.167 db2server.services.com loghost`

Регистрируясь в базе данных, в команде `dlfm add_db` в качестве имени хоста используйте `db2server.services.com`. Если вместо него в команде `dlfm add_db` указать другой алиас, внутреннее соединение сервера DB2 с DLFM завершится неудачно.

Сервер связей данных регистрируется на базе данных DB2 при помощи команды DB2 `"add datalinks manager for database алиас_базы_данных using node имя_хоста port номер_порта"`.

Имя хоста - это имя сервера связей данных. В этой команде можно использовать любой допустимый алиас сервера связей данных. Значения DATALINK, которые ссылаются на этот сервер связей данных, должны содержать в URL имя хоста, то есть точное имя, которое было использовано в команде `"add datalinks manager"`, надо использовать и при назначении URL столбцам DATALINK. Использование другого алиаса вызовет ошибку при выполнении оператора SQL.

Работа с менеджером файлов связей данных: очистка после удаления менеджера связей данных DB2 из базы данных DB2

Когда менеджер связей данных DB2 удаляется из базы данных командой DROP DATALINKS MANAGER, сама эта команда не очищает соответствующую информацию в менеджере связей данных DB2. Пользователи могут явно запустить отсоединение файлов, связанных с базой данных, и сборку мусора информации копирования. Для этого используется команда `dlfm drop_dlm`. Эта команда запускает асинхронное удаление всей информации для определенной базы данных. Чтобы эта команда сработала, менеджер связей данных DB2 должен быть запущен. Крайне важно использовать эту команду только после удаления менеджера связей данных DB2 из базы данных, иначе важная информация о менеджере связей данных DB2 будет потеряна безвозвратно.

Чтобы запустить удаление связей и чистку мусора информации резервного копирования для определенной базы данных:

1. Войдите в систему как администратор менеджера связей данных DB2.
2. Введите команду:

```
dlfm drop_dlm база-данных экземпляр имя-хоста
```

где:

база-данных - имя удаленной базы данных DB2 UDB;
экземпляр - имя экземпляра, в котором находится эта база данных; и
имя-хоста - имя хоста сервера DB2 UDB,
на котором находится эта база данных.

3. Выйдите из системы.

Полный сценарий использования, где показан контекст выполнения этой команды, смотрите в справочнике *Command Reference*.

Для этой команды введен новый код ошибки (смотрите раздел “DLFM1001E (новое сообщение об ошибке)”).

DLFM1001E (новое сообщение об ошибке)

DLFM1001E: Ошибка при обработке `drop_dlm`.

Причина:

Менеджер связей данных не смог запустить отсоединение связей и сборку мусора для указанной базы данных. Это могло произойти по одной из следующих причин:

- Менеджер связей данных не работает.
- В команде задано неверное сочетание базы данных, имени экземпляра и имени хоста.
- Ошибка одной из служб компонентов менеджера связей данных.

Действия:

Выполните следующие действия:

1. Убедитесь, что менеджер связей данных работает. Если он еще не работает, запустите его.

2. Убедитесь, что сочетание базы данных, имени экземпляра и имени хоста соответствует зарегистрированной базе данных. Это можно сделать при помощи команды "dlfm list registered databases" в менеджере связей данных.
3. Если ошибка повторяется, посмотрите информацию в файле db2diag.log, чтобы проверить, не зарегистрирована ли ошибка одной из служб компонентов (например, службы управления соединениями, службы управления транзакциями и т.п.). Запишите код ошибки из db2diag.log и предпримите действия для этого кода ошибки.

Опция задания файла конфигурации DLFM

Опция задания dlfm.cfg для DFLM исключена. Ссылки на эту опцию в документации надо игнорировать.

Ошибка при работе сценария связей данных/DFS dmapp_prestart в AIX

Если команда
`/usr/sbin/cfgdmpci -a "/usr/lib/drivers/dmlfs.ext"`

завершается неудачно с кодом возврата 1 при выполнении сценария связей данных/DFS "dmapp_prestart", установите DFS 3.1 ptfset1, чтобы исправить cfgdmpci.

Глава 4. Installing and Configuring DB2 Data Links Manager for AIX (Установка и конфигурирование менеджера связей данных DB2 для AIX)

Общие замечания по установке

Перенастройка менеджера файлов DB2 Версии 5.2 в Менеджер связей данных DB2 Версии 7

Информация в шаге 3 неверна. Шаг 3 должен выглядеть так:

"3. Как администратор DLFM выполните команду
`/usr/lpp/db2_07_01/adm/db2dlmmg`.

Приложение A. DB2 Data Links Manager Errors and User Responses (Ошибки менеджера связей данных DB2 и действия пользователя)

SQL357

Менеджер связей данных DB2 "<имя>" в данный момент недоступен.

Код причины 3: Доступ к менеджеру связей данных DB2 в данный момент не разрешен.

Причина: Не удалось соединиться с менеджером связей данных DB2.

Действия: Проверьте все параметры хоста для этого менеджера связей данных.

Глава 5. Дополнение по установке и настройке

Связывание утилит базы данных при использовании клиента времени выполнения

Клиент времени выполнения не может использоваться для связывания утилит базы данных (`import`, `export`, `georg`, процессор командной строки) или файлов связывания CLI DB2 с базами данных, чтобы они могли использоваться с этими базами данных. Вместо него нужно использовать клиент администратора DB2 или клиент разработки прикладных программ DB2.

Прежде чем утилиты базы данных и файлы связывания CLI DB2 можно будет использовать с базой данных, они должны быть связаны с этой базой данных. Если в сетевой среде используется несколько клиентов, работающих в разных операционных системах или имеющих разные уровни версий или служб DB2, необходимо связать утилиты один раз для каждой комбинации операционной системы и версии DB2.

Доступ к DB2 с использованием ODBC из клиентов UNIX

В главе 12 ("Выполнение ваших прикладных программ") сказано, что если с прикладной программой клиента ODBC или SDK ODBC установлен менеджер драйверов ODBC, необходимо изменить содержимое файла `odbcinst.ini`. Это не совсем так. Не нужно изменять `odbcinst.ini`, если установлен продукт Merant ODBC Driver Manager.

Переключение NetQuestion for OS/2 на использование TCP/IP

Инструкции по переключению NetQuestion на использование TCP/IP в системах OS/2 неполны. Указанные в этих инструкциях файлы `*.cfg` находятся в подкаталоге данных каталога установки NetQuestion. Чтобы узнать каталог установки NetQuestion, можно использовать одну из следующих команд:

```
echo %IMNINSTSRV% //для установок SBCS
echo %IMQINSTSRV% //для установок DBCS
```

Глава 26. Настройка системы объединения на работу с источниками данных Oracle

Раздел "Добавление источников данных Oracle в систему объединения" содержит следующие ошибки:

- В процедуре пропущен один из шагов. Правильные шаги:

1. Установите и сконфигурируйте программное обеспечение клиента Oracle на сервере объединения DB2, руководствуясь документацией Oracle.
- 2.

Задайте переменные среды источника данных: внесите изменения в файл `db2dj.ini` и введите команду `db2set`. Команда `db2set` приводит реестр профиля DB2 в соответствие с вашими настройками.

Подробные инструкции по заданию переменных среды приводятся в Главе 26. Настройка системы объединения на работу с источниками данных Oracle Data Sources книги *Дополнение по установке и настройке DB2*.

| 3. Если вы используете сервер объединения DB2 на платформе UNIX, запустите
| сценарий djsxlink, чтобы скомпоновать библиотеки Oracle для SQL*Net или Net8
| под ваш сервер объединения DB2. Сценарий djsxlink находится в каталоге
| /install_directory/bin. Запустите этот сценарий только по окончании
| установки клиентского программного обеспечения Oracle на сервере
| объединения DB2.

- В документации рекомендуется задать:

DB2_DJ_INI = sqllib/cfg/db2dj.ini

Это неверно, правильное значение следующее:

DB2_DJ_INI = \$INSTHOME/sqllib/cfg/db2dj.ini

Часть 4. Управление

Глава 6. Руководство администратора:

Планирование 39

Глава 8. Физическая структура базы данных 39

Ключи разделения 39

Разработка групп узлов 39

Глава 9. Разработка распределенных баз данных 39

Изменение нескольких баз данных 39

Глава 13. Высокая доступность в среде Windows NT 40

Необходимость перезагрузки компьютера перед

запуском утилиты DB2MCS. 40

Глава 14. DB2 и высокая доступность Sun Cluster 2.2 40

Поддержка Veritas в Solaris 40

Приложение В. Правила именования 41

Замечания об ID пользователей и именах схем

длиной более 8 символов 41

ID пользователей и пароли 42

Приложение D. Несовместимость выпусков 42

DLFS Windows NT несовместима с Norton Utilities 42

SET CONSTRAINTS изменено на SET INTEGRITY 43

Приложение E. Поддержка национальных языков 43

Национальные версии DB2 Версии 7.1 43

Наборы файлов Центра управления и

документации. 45

Национальные настройки для сервера

администратора DB2 46

DB2 UDB на платформах Windows поддерживает

кодovou страницу для стран Балтии (MS-1257). 46

Определение значений кодовых страниц 46

Поддержка кода страны и кодовой страницы 46

Наборы символов 47

Глава 7. Руководство администратора:

Реализация 49

Добавление или расширение контейнеров DMS

(новый процесс) 49

Глава 1. Управление DB2 при помощи инструментов

GUI 49

Глава 3. Создание базы данных 50

Создание табличного пространства 50

Использование непосредственного

ввода/вывода в Linux 50

Создание последовательности 51

Сравнение столбцов идентификации (IDENTITY) и

последовательностей 53

Создание индекса, расширения индекса или

спецификации индекса 53

Глава 4. Изменение базы данных 54

Добавление к табличному пространству SMS

контейнера на разделе 54

Изменение столбца идентификации. 54

Изменение последовательности 55

Отбрасывание последовательности. 55

Переключение состояния табличного пространства 55

Изменение контейнеров в табличном пространстве

DMS. 56

Глава 5. Управление доступом к базам данных 56

Привилегии последовательностей 56

Шифрование данных 56

Глава 8. Восстановление базы данных 57

Как использовать приостановленный ввод-вывод 58

Инкрементное резервное копирование и

восстановление 59

Восстановление из инкрементных резервных

копий 61

Параллельное восстановление 62

Резервное копирование в именованные конвейеры 62

Резервное копирование из отдельной копии 63

Архивирование журнала по требованию 65

Создание зеркальной копии журнала 65

Поддержка кроссплатформенного резервного

копирования и восстановления на Sun Solaris и HP . 66

Особенности менеджера связей данных

DB2/Особенности утилиты резервного

копирования 66

Особенности менеджера связей данных

DB2/Особенности утилиты восстановления и

повтора транзакций. 67

Восстановление баз данных из резервных копий,

сделанных в автономном режиме, без повтора

транзакций 68

Восстановление баз данных и табличных

пространств и повтор транзакций до конца файлов

журнала 69

Взаимодействие менеджера связей данных и

восстановления 69

Обнаружение ситуации, требующей согласования 72

Приложение С. Обработчик пользователя для

восстановления базы данных. 73

Приложение I. Высокоскоростная межузловая связь 74

Подготовка DB2 к работе с использованием VI . 74

Глава 8. Руководство администратора:

Производительность 77

Глава 3. Особенности прикладных программ 77

Задание уровня изоляции 77

Настройка класса оптимизации 77

Динамические составные операторы 78

Глава 4. Особенности среды 78

Использование индексных ключей большего

размера. 78

Глава 5. Статистики системного каталога 79

Сбор и использование статистики распределения 79

Правила обновления статистики каталога 79

Статистика подэлементов. 80

Глава 6. Компилятор SQL. 82

Реплицируемые сводные таблицы 82

Концепции доступа к данным и оптимизация 84

Глава 8. Производительность работы 84

Управление пулом буферов базы данных 84

Управление несколькими пулами буферов базы

данных 85

Глава 9. Использование утилиты ограничения ресурсов	86	EXPORT	115
Глава 13. Конфигурирование DB2	86	GET DATABASE CONFIGURATION	115
Размер кучи сортировки (sortheap)	86	GET ROUTINE (новая команда)	116
Порог кучи сортировки (sheapthres).	87	GET ROUTINE	117
Максимальный процент списка блокировок перед расширением (maxlocks)	87	GET SNAPSHOT	117
Конфигурирование для DB2/Менеджера связей данных DB2 интервала истечения срока маркера доступа к связям данных dl_exptnt	88	IMPORT	117
Параметр конфигурации базы данных MIN_DEC_DIV_3	88	LIST HISTORY	118
Размер кучи управления программой (app_ctl_heap_sz)	89	LOAD	119
Размер кучи системного монитора базы данных (mon_heap_sz).	89	PING (новая команда).	120
Максимальное число активных прикладных программ (maxarpls)	89	PING	121
Диапазон восстановления и интервал мягких контрольных точек (softmax)	90	PUT ROUTINE (новая команда)	122
Разрешить отслеживание измененных страниц (trackmod)	90	PUT ROUTINE	123
Изменить путь журнала базы данных (newlogpath)	90	RECONCILE	123
Положение файлов журналов (logpath).	91	REORGANIZE TABLE	124
Максимальная память для списка блокировок (locklist)	92	RESTORE DATABASE	124
Приложение А. Переменные реестра DB2 и переменные среды	92	Синтаксис	124
Таблица новых и измененных переменных реестра	92	Особенности менеджера связей данных DB2	125
Приложение С. Средства объяснения SQL	96	ROLLFORWARD DATABASE	125
Глава 9. Command Reference	97	UPDATE HISTORY FILE	125
db2batch - Замер производительности	97	Ошибка документации в разделе CLP Return Codes (Коды возврата процессора командной строки)	126
db2cap (новая команда)	97	Глава 10. Data Movement Utilities Guide and Reference	127
db2cap - Утилита связывания статического пакета CLI/ODBC	98	Состояния отложенных действий после операции загрузки	127
db2ckrst - Проверка последовательности резервных копий для инкрементного восстановления (новая команда)	100	Ограничения загрузки.	127
db2gncol (новая команда)	101	Для запуска автозагрузки при значении Authentication YES необходим hexecd	127
db2gncol - Изменить значения генерируемых столбцов	102	Модификатор типа файла totalreespace	127
db2inidb - инициализировать зеркальную копию базы данных	103	Использование импорта с буферизованными вставками	127
db2look - Утилита извлечения статистики DB2	104	Глава 11. Replication Guide and Reference	129
db2updv7 - обновление базы данных до уровня текущего пакета FixPak Версии 7	104	Репликация в Windows 2000	129
Новая опция процессора командной строки (-x, подавление вывода заголовков столбцов)	105	Имена столбцов и таблиц	129
Для процессора командной строки DB2 необходим шрифт True Type	105	Репликация столбцов DATALINK	129
ADD DATALINKS MANAGER	105	Ограничения на большие объекты.	129
ARCHIVE LOG (новая команда)	106	Репликация и серверы других типов (не IBM)	130
Archive Log	107	Предварительное требование для повсеместного изменения	130
BACKUP DATABASE	108	Сценарии репликации	130
Синтаксическая диаграмма	108	Планирование репликации	131
Особенности менеджера связей данных DB2	109	Настройка среды репликации	131
BIND	109	Диагностика ошибок	131
CALL	110	Программы Capture и Apply для AS/400	132
DROP DATALINKS MANAGER (новая команда)	111	Структуры таблиц	133
DROP DATALINKS MANAGER	112	Сообщения программ Capture и Apply	133
		Запуск программ Capture и Apply из прикладной программы	133
		Глава 12. System Monitor Guide and Reference	135
		db2ConvMonStream.	135
		Глава 13. Troubleshooting Guide	137
		Запуск DB2 в Windows 95, Windows 98 и Windows ME, когда пользователь не зарегистрирован в системе	137
		Глава 14. Использование DB2 Universal Database на 64-битных платформах	141

Глава 5. Configuration	141
shmsys:shminfo_shmmax	141
DBHEAP	141
LOCKLIST	141
Глава 6. Restrictions	141
Глава 15. XML Extender Administration and Programming	143

Глава 6. Руководство администратора: Планирование

Глава 8. Физическая структура базы данных

Ключи разделения

В подразделе “Особенности проектирования групп узлов” раздела “Проектирование групп узлов” при описании ограничений для ключей разделения следующие пункты не касаются случая, когда DB2_UPDATE_PART_KEY=ON:

Примечание: Если DB2_UPDATE_PART_KEY=OFF, эти ограничения остаются в силе.

Примечание: В FixPak 3 и более новых значение по умолчанию - ON.

- В строке таблицы нельзя изменить значение столбца, входящего в ключ разделения.
- Значения столбца, входящего в ключ разделения, можно только удалять и вставлять.

Разработка групп узлов

В разделе под названием “Разработка групп узлов” в подразделе “Особенности проектирования групп узлов” и подразделе “Реплицируемые таблицы сводок” отбросьте последнее предложение второго абзаца:

Ключевое слово REPLICATED можно задавать только для сводной таблицы, определенной с опцией REFRESH DEFERRED.

Глава 9. Разработка распределенных баз данных

Изменение нескольких баз данных

В разделе “Изменение нескольких баз данных” есть неточность в описании шагов по установке. Шаг 4, который в английском издании утверждает:

Precompile your application program to specify a type 2 connection (that is, specify CONNECT 2 on the PRECOMPILE PROGRAM command), and one-phase commit (that is, specify SYNCPOINT ONEPHASE on the PRECOMPILE PROGRAM command), as described in the Application Development Guide.

следует изменить на:

Прекомпилировать прикладную программу, задав соединение типа 2 (для этого надо задать CONNECT 2 в команде PRECOMPILE PROGRAM) и двухфазное принятие (для этого надо задать SYNCPOINT TWOPHASE в команде PRECOMPILE PROGRAM), как описано в книге Application Development Guide.

Глава 13. Высокая доступность в среде Windows NT

Необходимость перезагрузки компьютера перед запуском утилиты DB2MSCS

Утилита DB2MSCS используется для конфигурирования DB2 для поддержки восстановления после отказов в среде Microsoft Cluster Service. Для успешного запуска утилиты DB2MSCS Cluster Service должна иметь возможность найти библиотеку DLL ресурсов, db2wolf.dll, которая находится в каталоге %ProgramFiles%\SQLLIB\bin.

Программа установки DB2 UDB Версии 7.1 включает каталог %ProgramFiles%\SQLLIB\bin в переменную среды PATH системы. Однако при работе в среде Windows 2000 перезапуска компьютера после установки не требуется.

Если вы хотите запустить утилиту DB2MSCS, надо перезапустить компьютер, чтобы изменить переменную среды PATH для Cluster Service.

Глава 14. DB2 и высокая доступность Sun Cluster 2.2

DB2 Connect поддерживается в Sun Cluster 2.2, если:

- Для связи с хостом используется протокол TCP/IP (не SNA)
- Двухфазное принятие не используется. Это ограничение ослабляется, если пользователь сконфигурирует журнал SPM на совместно используемом диске (это можно сделать при помощи параметра конфигурации менеджера баз данных *spm_log_path*) и если подменяющий компьютер имеет такую же конфигурацию TCP/IP (те же имя хоста, IP-адрес и т.д.).

Поддержка Veritas в Solaris

Теперь DB2 поддерживает Veritas, который обеспечивает поддержку кластеров в Solaris.

Описание

Включает, отключает экземпляры DB2 UDB и наблюдает за их работой.

Точки входа

Online (включение)

Команда db2start включает экземпляр.

Offline (отключение)

Команда db2stop отключает экземпляр.

Monitor (наблюдение)

Определяет, работает ли заданный экземпляр DB2. Использует соответствующий мониторинг процесса и (необязательно) мониторинг баз данных.

Clean (очистка)

Удаляет ресурсы экземпляра DB2.

Атрибут	Тип	Определение
probeDatabase	string	База данных для наблюдения
instanceOwner	string	Имя владельца экземпляра
instanceHome	string	Начальный каталог владельца экземпляра
probeTable	string	Таблица в probeDatabase для наблюдения

monitorLevel	integer	1 - мониторинг процесса, 2 - мониторинг базы данных
nodeNumber	integer	Номер узла для запуска экземпляра (не задан, если EE)

Определение типа

```

type DB2UDB (
    static int CleanTimeout = 240
    static int MonitorTimeout = 30
    static int OfflineTimeout = 240
    static int OnlineRetryLimit = 2
    static int OnlineTimeout = 120
    static int OnlineWaitLimit = 1
    static int RestartLimit = 3
    static int ToleranceLimit = 1
    static str ArgList[] =
{ probeDatabase, instanceOwner, instanceHome, probeTable, monitorLevel, nodeNumber }
    NameRule = resource.db2udb
    str probeDatabase
    str instanceOwner
    str instanceHome
    str probeTable
    int monitorLevel
    int nodeNumber
)

```

Пример конфигурации

```

DB2UDB db2_resource_n0 (
    probeDatabase = sample
    probeTable = vcstable
    instanceOwner = stevera
    instanceHome = "/export/home/stevera"
    monitorLevel = 2
)

```

Установка

- Создайте каталог /opt/VRTSvcs/bin/DB2UDB.
- Скопируйте файлы online, offline, monitor, clean и DB2UDBAgent в /opt/VRTSvcs/bin/DB2UDB и убедитесь, что они помечены как исполняемые.
- Скопируйте файл db2udb.type.cf в /etc/VRTSvcs/conf/config.
- Остановите кластер (например, hastop -all).
- Добавьте строку include db2udb.type.cf в файл main.cf после строки include types.cf
- Проверьте правильность конфигурации кластера командой /opt/VRTSvcs/bin/hacf -verify /etc/VRTSvcs/conf/config

Теперь вы готовы создавать ресурсы DB2, необходимые для управления экземплярами DB2.

Приложение В. Правила именования

Замечания об ID пользователей и именах схем длиной более 8 символов

- Продукты DB2 Версии 7.1 для 32-битных платформ Windows поддерживают ID пользователей длиной до 30 символов. Однако сами системы Windows NT и Windows 2000 ограничивают фактическую максимальную длину ID пользователя 20 символами.

- DB2 Версии 7.1 поддерживает соединения с Windows NT и Windows 2000 от клиентов из других систем (не 32-битных систем Windows) с ID пользователей, длина которых превышает 8 символов, если ID пользователя и пароль задаются явно. Это не так для соединений, использующих аутентификацию клиента или аутентификацию DCE.
- Аутентификация DCE на всех платформах по-прежнему допускает ID пользователя не длиннее 8 символов.
- ID авторизации, возвращаемый в SQLCA успешно выполненным оператором CONNECT или ATTACH, усекается до 8 символов. Если ID авторизации был усечен, поле SQLWARN содержит предупреждения. Дополнительную информацию смотрите в описании оператора CONNECT в руководстве *SQL Reference*.
- ID авторизации, возвращаемый процессором командной строки (CLP) для успешно выполненного оператора CONNECT или ATTACH, усекается до 8 символов. Если ID авторизации был усечен, он дополняется многоточием (...).
- DB2 Версии 7.1 поддерживает имена схем длиной до 30 байт, за исключением следующих случаев:
 - Таблицы с именами схем, длина которых больше 18 байт, не могут быть реплицированы.
 - Имена схем пользовательских типов не могут превышать по длине 8 байт.

ID пользователей и пароли

В разделе “ID пользователей и пароли” слова “от A до Z” надо заменить на:

Однобайтные большие и маленькие латинские буквы (от A до Z и от a до z).
Поддержка других символов зависит от используемой кодовой страницы.
Дополнительную информацию о поддержке кодовых страниц смотрите в приложении “Поддержка национальных языков (NLS)”.

Приложение D. Несовместимость выпусков

DLFS Windows NT несовместима с Norton Utilities

Файловая система связей данных Windows NT несовместима с Norton Utilities. При удалении файла с дисководов, управляемого DLFS, происходит исключительная ситуация ядра: error 0x1E (Kernel Mode Exception Not Handled). Это исключительная ситуация 0xC0000005 (Нарушение доступа). Это нарушение происходит из-за того, что драйвер Norton Utilities загружается после того, как загружен драйвер фильтра DLFS.

Временным обходным приемом для этого положения является загрузка драйвера DLFSД после драйвера Norton Utilities. Это может быть сделано путем перехода на запуск драйвера DLFSД вручную. Нажмите кнопку **Пуск**, выберите **Настройка** → **Панель управления** → **Устройства** → **DLFSД** и установите для нее ручной режим.

Можно создать пакетный файл, который при запуске системы загружает драйвер DLFSД и службу DLFM, и добавить его в папку Автозагрузка. Этот пакетный файл должен выглядеть так:

```
net start dlfsd
net start "dlfm service"
```

Назовите этот пакетный файл start_dlfs.bat и скопируйте его в каталог C:\WINNT\Profiles\Administrator\Start Menu\Programs\Startup.

Привилегия загружать драйвер фильтра DLFS и службу DLFM есть только у администратора.

SET CONSTRAINTS изменено на SET INTEGRITY

Оператор SET CONSTRAINTS заменен оператором SET INTEGRITY. Для обратной совместимости DB2 UDB V7 воспринимает оба оператора.

Приложение Е. Поддержка национальных языков

Национальные версии DB2 Версии 7.1

DB2 Версии 7.1 доступна на английском, французском, немецком, итальянском, испанском, бразильском португальском, японском, корейском, упрощенном китайском, традиционном китайском, датском, финском, норвежском, шведском, чешском, голландском, венгерском, польском, турецком, русском, болгарском и словенском языках.

Для платформ на основе UNIX можно установить сообщения продуктов DB2 и библиотеку для нескольких языков. Утилита установки DB2 помещает наборы файлов каталога сообщений в общепринятые каталоги национальных версий для данной платформы, как показано в следующих таблицах. В Табл. 1 дается информация для AIX, HP-UX и Solaris. В Табл. 2 на стр. 44 дается информация для Linux, Linux/390, SGI и Dynix.

Таблица 1. AIX, HP-UX, Solaris

Операционная система	AIX		HP-UX		Solaris	
Язык	Нац.версия	Код.стр.	Нац.версия	Код.стр.	Нац.версия	Код.стр.
Французский	fr_FR	819	fr_FR.iso88591	819	fr	819
	Fr_FR	850	fr_FR.roman8	1051		
Немецкий	de_DE	819	de_DE.iso88591	819	de	819
	De_DE	850	de_DE.roman8	1051		
Итальянский	it_IT	819	it_IT.iso88591	819	it	819
	It_IT	850	it_IT.roman8	1051		
Испанский	es_ES	819	es_ES.iso88591	819	es	819
	Es_ES	850	es_ES.roman8	1051		
Бразильский португальский	pt_BR	819			pt_BR	819
японский	ja_JP	954	ja_JP.eucJP	954	ja	954
	Ja_JP	932				
Корейский	ko_KR	970	ko_KR.eucKR	970	ko	970
упрощенный китайский	zh_CN	1383	zh_CN.hp15CN	1383	zh	1383
	Zh_CN.GBK	1386				
Традиционный китайский	zh_TW	964	zh_TW.eucTW	964	zh_TW	964
	Zh_TW	950	zh_TW.big5	950		
Датский	da_DK	819	da_DK.iso88591	819	da	819
	Da_DK	850	da_DK.roman8	1051		

Таблица 1. AIX, HP-UX, Solaris (продолжение)

Финский	fi_FI	819	fi_FI.iso88591	819	fi	819
	Fi_FI	850	fi_FI.roman8	1051		
Норвежский	no_NO	819	no_NO.iso88591	819	no	819
	No_NO	850	no_NO.roman8	1051		
Шведский	sv_SE	819	sv_SE.iso88591	819	sv	819
	Sv_SE	850	sv_SE.roman8	1051		
Чешский	cs_CZ	912				
Венгерский	hu_HU	912				
Польский	pl_PL	912				
Голландский	nL_NL	819				
	NL_NL	850				
Турецкий	tr_TR	920				
Русский	ru_RU	915				
Болгарский	bg_BG	915	bg_BG.iso88595	915		
Словенский	sl_SI	912	sl_SI.iso88592	912	sl_SI	912

Таблица 2. Linux, Linux/390, SGI, Dynix

Операционная система	Linux		Linux/390		SGI		Dynix	
	Нац. версия	Код. стр.	Нац. версия	Код. стр.	Нац. версия	Код. стр.	Нац. версия	Код. стр.
Французский	fr	819	fr	819			fr	819
Немецкий	de	819	de	819			de	819
Итальянский							es	819
Испанский								
Бразильский португальский								
японский	ja_JP.ujis	954	ja_JP.ujis	954			ja_JP.EUC	954
Корейский	ko	970	ko	970	ko_KO.euc	970		
упрощенный китайский	zh zh_CN.GBK	1386	zh zh_CN.GBK	1386				
Традиционный китайский	zh_TW.Big5	950	zh_TW.Big5	950				
Датский								
Финский								
Норвежский								
Шведский								
Чешский								
Венгерский								
Польский								
Голландский							nl	819
Турецкий								

Таблица 2. Linux, Linux/390, SGI, Dynix (продолжение)

Русский								
Болгарский								
Словенский								

Если ваша система использует те же коды страниц, но другие имена национальных версий по сравнению с указанными, переведенные сообщения можно будет видеть, если создать связь с соответствующим каталогом сообщений.

Например, если на вашем компьютере AIX по умолчанию используется национальная версия ja_JP.IBM-eucJP, а кодовая страница для ja_JP.IBM-eucJP - 954, можно создать связь из /usr/lpp/db2_07_01/msg/ja_JP.IBM-eucJP в /usr/lpp/db2_07_01/msg/ja_JP при помощи команды:

```
ln -s /usr/lpp/db2_07_01/msg/ja_JP /usr/lpp/db2_07_01/msg/ja_JP.IBM-eucJP
```

После выполнения этой команды все сообщения DB2 выводятся на японском языке.

Наборы файлов Центра управления и документации

Наборы файлов Центра управления, справки Центра управления и документации помещаются на рабочей станции назначения в следующие каталоги:

- DB2 for AIX:
 - /usr/lpp/db2_07_01/cc/%L
 - /usr/lpp/db2_07_01/java/%L
 - /usr/lpp/db2_07_01/doc/%L
 - /usr/lpp/db2_07_01/qp/\$L
 - /usr/lpp/db2_07_01/spb/%L
- DB2 for HP-UX:
 - /opt/IBMdb2/V7.1/cc/%L
 - /opt/IBMdb2/V7.1/java/%L
 - /opt/IBMdb2/V7.1/doc/%L
- DB2 for Linux:
 - /usr/IBMdb2/V7.1/cc/%L
 - /usr/IBMdb2/V7.1/java/%L
 - /usr/IBMdb2/V7.1/doc/%L
- DB2 for Solaris:
 - /opt/IBMdb2/V7.1/cc/%L
 - /usr/IBMdb2/V7.1/java/%L
 - /opt/IBMdb2/V7.1/doc/%L

Наборы файлов Центра управления используют кодовую страницу Unicode.

Документация и справка Центра управления используют кодовую страницу, которую может распознать браузер. Если в вашей системе используется имя национальной версии, отличное от указанного, для запуска переведенной версии Центра управления и получения переведенного варианта справки надо создать связи с соответствующими языковыми каталогами.

Например, если на вашем компьютере AIX по умолчанию используется национальная версия ja_JP.IBM-eucJP, можно создать связь из

/usr/lpp/db2_07_01/cc/ja_JP.IBM-eucJP в /usr/lpp/db2_07_01/cc/ja_JP и из /usr/lpp/db2_07_01/doc/ja_JP.IBM-eucJP в /usr/lpp/db2_07_01/doc/ja_JP при помощи команд:

- `ln -s /usr/lpp/db2_07_01/cc/ja_JP /usr/lpp/db2_07_01/cc/ja_JP.IBM-eucJP`
- `ln -s /usr/lpp/db2_07_01/doc/ja_JP /usr/lpp/db2_07_01/doc/ja_JP.IBM-eucJP`

После выполнения этих команд текст Центра управления и справки выводятся на японском языке.

Примечание: Управляющий центр Web не поддерживается на самих серверах Linux/390 и NUMA-Q. Для управления базами данных на этих платформах его можно запустить с клиентской рабочей станции.

Национальные настройки для сервера администратора DB2

Убедитесь, что национальная версия экземпляра сервера администратора DB2 совместима с национальной версией экземпляра DB2. В противном случае экземпляр DB2 не сможет связываться с сервером администратора DB2.

Если в профиле пользователя сервера администратора DB2 не задана переменная среды LANG, сервер администратора DB2 будет запущен с системной национальной версией по умолчанию. Если системная национальная версия по умолчанию не определена, сервер администратора будет запущен с кодовой страницей 819. Если экземпляр DB2 использует одну из национальных версий DBCS, а сервер администратора DB2 запущен с кодовой страницей 819, этот экземпляр не сможет связываться с сервером администратора DB2. Национальная версия экземпляра сервера администратора DB2 и национальная версия экземпляра DB2 должны быть совместимы. Например, в системе Linux с упрощенным китайским в профиле пользователя сервера администратора DB2 должно быть задано "LANG=zh_CN".

DB2 UDB на платформах Windows поддерживает кодовую страницу для стран Балтии (MS-1257)

В 32-битных операционных системах Windows DB2 UDB поддерживает кодовую страницу стран Балтии, MS-1257. Эта кодовая страница используется для латышского, литовского и эстонского языков.

Определение значений кодовых страниц

В разделе "Определение значений кодовых страниц" первый абзац надо изменить с:

Тем не менее устанавливать переменную реестра DB2CODEPAGE нет необходимости, поскольку DB2 определит необходимое значение кодовой страницы из операционной системы.

на: Как правило, устанавливать переменную реестра DB2CODEPAGE нет необходимости, поскольку DB2 определит необходимое значение кодовой страницы из операционной системы.

Поддержка кода страны и кодовой страницы

В разделе "Поддержка кода страниц и кодовой страницы" добавьте в таблицу следующую строку:

Кодовая страница	Группа	Набор кодов	Об.	Код стр.	Локаль	ОС	Страна
943	D-1	IBM-943	JP	81	ja_JP.PCK	Sun	Япония

Наборы символов

В разделе под названием “Наборы символов” в подразделе “Набор символов для идентификаторов” последние два предложения первого абзаца надо изменить на следующие:

Используйте специальные символы #, @ и \$ в среде NLS с осторожностью, поскольку они не включены в инвариантный набор символов хоста NLS (EBCDIC). В зависимости от применяемой кодовой страницы можно использовать и символы из расширенного набора символов. Если вы используете базу данных в среде с несколькими кодовыми страницами, необходимо убедиться, что все кодовые страницы поддерживают все элементы из того расширенного набора символов, который вы планируете использовать.

Глава 7. Руководство администратора: Реализация

Добавление или расширение контейнеров DMS (новый процесс)

Контейнеры DMS (как контейнеры файлов, так и контейнеры непосредственных устройств) теперь добавляются (во время или после создания табличного пространства) или расширяются программами предварительного чтения параллельно. Чтобы достичь большего параллелизма операций по созданию контейнеров и изменению их размеров, можно увеличить число выполняемых в системе программ предварительного чтения. Единственный процесс, который не выполняется в параллельном режиме - это регистрация этих действий и (для создания контейнеров) маркировка контейнеров.

Примечание: Параллелизм операций CREATE TABLESPACE/ALTER TABLESPACE (для добавления новых контейнеров к существующему табличному пространству) возрастает, пока число программ предварительного чтения не станет равным числу добавляемых контейнеров.

Глава 1. Управление DB2 при помощи инструментов GUI

В разделе, озаглавленном “Центр оповещения”, удалите последние два предложения раздела.

В разделе, озаглавленном “Монитор производительности”, удалите второй пункт списка “Определения переменных производительности” в подразделе “Мониторинг производительности в некоторый момент времени”.

Ниже в том же подразделе несколько последних абзацев раздела следует заменить на:

Для каждого пункта можно наблюдать за множеством переменных производительности. Описание всех переменных производительности смотрите в справочнике по переменным производительности в электронной справке, вызываемой из меню **Справка** любого окна Монитора снимков. Эти переменные собраны по категориям. Существуют следующие категории:

- Экземпляр: Агенты, Соединения, Сортировка
- База данных: Блокировка и тупиковая ситуация, Пул буферов и ввод/вывод, Соединения, Сортировка, Активность операторов SQL
- Таблица: Таблица
- Табличное пространство: Пул буферов и ввод/вывод
- Соединения с базами данных: Пул буферов и ввод/вывод, Блокировка и тупиковая ситуация, Сортировка, Активность операторов SQL

Подробности о создании снимков смотрите в электронной справке.

В том же разделе удалите последнее предложение в подразделе, озаглавленном “Действия в случае появления объекта в Центре оповещений”.

Глава 3. Создание базы данных

Создание табличного пространства

Использование непосредственного ввода/вывода в Linux

В Linux есть пул узлов с устройствами непосредственного ввода/вывода, который нужно связать с устройством блокировок, прежде чем на нем можно будет выполнять операции непосредственного ввода/вывода. Информация связывания непосредственных устройств с устройствами блочного ввода-вывода хранится на контроллере непосредственных устройств. Связывание выполняется при помощи утилиты под именем `raw`, которая обычно поставляется дистрибьютором Linux.

Прежде чем установить в Linux непосредственный ввод-вывод, вам понадобятся:

- один или несколько свободных разделов диска IDE или SCSI
- Ядро Linux 2.4.0 или более новое (некоторые дистрибьюторы Linux предлагают непосредственный ввод/вывод в ядрах 2.2).
- контроллер непосредственных устройств под именем `/dev/rawctl` или `/dev/raw`. Если используется другое имя, создайте символическую ссылку:

```
# ln -s /dev/ваш_контр_непоср_устр /dev/rawctl
```
- утилита `raw`, которая обычно поставляется с дистрибутивом Linux
- Пакет исправления DB2 версии 7.1 FixPак 3 или новее

Примечание: В распространяемых в настоящее время дистрибутивах, поддерживающих непосредственный ввод-вывод, узлы непосредственных устройств называются по-разному:

Дистрибутив	Узлы непосредственных устройств	Контроллер непосредственных устройств
RedHat 6.2	<code>/dev/raw/raw1 to 255</code>	<code>/dev/rawctl</code>
SuSE 7.0	<code>/dev/raw1 to 63</code>	<code>/dev/raw</code>

DB2 поддерживает все вышеперечисленные контроллеры непосредственных устройств и большинство других названий для узлов непосредственных устройств. DB2 не поддерживает непосредственных устройств в Linux/390.

Чтобы сконфигурировать непосредственный ввод-вывод в Linux:

В этом примере используемый раздел `raw - /dev/sda5`. В нем не должно быть никакой существенной информации.

Шаг 1. Вычислите количество страниц по 4096 байт, округляя, при необходимости, в меньшую сторону. Например:

```
# fdisk /dev/sda
Command (m for help): p

Disk /dev/sda: 255 heads, 63 sectors, 1106 cylinders
Units = cylinders of 16065 * 512 bytes

   Device Boot   Start    End  Blocks  Id System
 /dev/sda1             1     523  4200997  83  Linux
 /dev/sda2             524    1106  4682947+  5  Extended
 /dev/sda5             524    1106  4682947  83  Linux

Command (m for help): q
#
```

Число страниц в /dev/sda5 равно

```
num_pages = floor( ((1106-524+1)*16065*512)/4096 )  
num_pages = 11170736
```

Шаг 2. Свяжите с этим разделом неиспользуемый узел непосредственного устройства. Это необходимо делать при каждой перезагрузке компьютера; для этого нужен доступ с полномочиями root. Чтобы посмотреть, какие узлы непосредственных устройств уже используются, введите команду raw -a:

```
# raw /dev/raw/raw1 /dev/sda5  
/dev/raw/raw1: bound to major 8, minor 5
```

Шаг 3. Задайте глобальные права чтения на контроллере непосредственного устройства и разделе диска. Задайте глобальные права чтения и записи на непосредственном устройстве:

```
# chmod a+r /dev/rawctl  
# chmod a+r /dev/sdb1  
# chmod a+rw /dev/raw/raw1
```

Шаг 4. Создайте табличное пространство в DB2, выбрав непосредственное устройство, а не раздел диска. Например:

```
CREATE TABLESPACE dms1  
MANAGED BY DATABASE  
USING (DEVICE '/dev/raw/raw1' 11170736)
```

Табличные пространства на непосредственных устройствах поддерживаются также для всех остальных размеров страниц, поддерживаемых DB2.

Создание последовательности

После раздела, озаглавленного “Определение столбца идентификации в новой таблице”, добавьте раздел “Создание последовательности”:

Последовательность - это объект базы данных, поддерживающий автоматическую генерацию значений. Последовательности идеально подходят для генерации уникальных значений ключей. Прикладные программы могут использовать последовательности для предотвращения конфликтов и проблем производительности из-за генерации счетчика уникальности вне базы данных.

В отличие от атрибута столбца идентификации, последовательность не привязана к конкретному столбцу таблицы и не связана с уникальным столбцом таблицы, и только доступ к ней происходит через этот столбец таблицы.

Последовательность можно создать или изменить, чтобы генерация значений происходила по одному из следующих методов:

- Монотонное увеличение или уменьшение без ограничений
- Монотонное увеличение или уменьшение до определяемого пользователем предела и остановка
- Монотонное увеличение или уменьшение до определяемого пользователем предела и циклическое повторение

Ниже приводится пример создания объекта последовательности:

```
CREATE SEQUENCE order_seq  
START WITH 1  
INCREMENT BY 1  
NOMAXVALUE  
NOCYCLE  
CACHE 24
```

В этом примере последовательность названа `order_seq`. Она начнется с 1 и будет увеличиваться на 1 без ограничений сверху. Поскольку верхний предел не задан, нет смысла задавать возврат к 1 и повторение сначала. Число, связанное с параметром `CACHE`, задает максимальное число значений последовательности, которые менеджер баз данных разместит и будет сохранять в памяти.

Генерируемые числовые последовательности обладают следующими свойствами:

- Значения могут быть любого точного числового типа данных с масштабом 0. К таким типам данных относятся: `SMALLINT`, `BIGINT`, `INTEGER` и `DECIMAL`.
- Последовательные значения могут отличаться на любое указанное целочисленное приращение. Значение приращения по умолчанию - 1.
- Значение счетчика допускает восстановление. Значение счетчика воссоздается по журналам, когда требуется восстановление.
- Значения могут помещаться в кэш для повышения производительности. Размещение и хранение значений в кэше уменьшает синхронный ввод/вывод в журнал при генерации значений для последовательности. При системном сбое все значения в кэше, которые не были приняты, больше не используются и считаются потерянными. Заданное значение `CACHE` - это максимальное число значений последовательности, которое может быть потеряно при этом.

Если база данных, содержащая одну или несколько последовательностей, восстанавливается до более раннего момента времени, возникает опасность генерации повторных значений для некоторых последовательностей. Чтобы исключить появление повторяющихся значений, не следует подвергать базу данных с последовательностями восстановлению до более раннего момента времени.

Последовательности поддерживаются только в одноузловых базах данных.

Для работы с последовательностями используются два выражения.

Выражение `PREVVAL` возвращает сгенерированное предыдущим оператором текущего сеанса значение данной последовательности.

Выражение `NEXTVAL` возвращает следующее значение данной последовательности. Новый член числовой последовательности генерируется, когда в выражении `NEXTVAL` задается имя этой последовательности. Однако если в одном запросе есть несколько экземпляров выражения `NEXTVAL` с одним и тем же именем последовательности, приращение счетчика последовательности производится только один раз для каждой строки результата.

Один и тот же член числовой последовательности может использоваться как значение ключа уникальности в двух отдельных таблицах, если в первой таблице ссылка на член числовой последовательности делается при помощи выражения `NEXTVAL`, а во всех остальных таблицах при помощи выражения `PREVVAL`.

Например:

```
INSERT INTO order (orderno, custno)
VALUES (NEXTVAL FOR order_seq, 123456);
INSERT INTO line_item (orderno, partno, quantity)
VALUES (PREVVAL FOR order_seq, 987654, 1)
```

Выражения `NEXTVAL` и `PREVVAL` можно использовать в следующих контекстах:

- оператор `INSERT`, условие `VALUES`
- оператор `SELECT`, список `SELECT`
- оператор присваивания `SET`

- оператор UPDATE, условие SET
- операторы VALUES и VALUES INTO

Сравнение столбцов идентификации (IDENTITY) и последовательностей

После нового раздела под заголовком “Создание последовательности” добавьте следующий раздел:

При всем сходстве столбцов идентификации и последовательностей между ними есть и различия. Особенности этих двух подходов могут использоваться при проектировании баз данных и прикладных программ.

Особенности столбца идентификации:

- Столбец идентификации может быть определен как часть таблицы только в момент создания таблицы. После того, как таблица создана, добавить к ней столбец идентификации нельзя. (Однако можно изменить характеристики существующего столбца идентификации.)
- Столбец идентификации автоматически генерирует значения для одной таблицы.
- Когда столбец идентификации определен как GENERATED ALWAYS, используемые значения всегда генерируются менеджером баз данных. Прикладным программам не разрешено задавать собственные значения при изменении содержимого таблицы.

Объект последовательности имеет следующие особенности:

- Объект последовательности - это объект базы данных, не привязанный к какой-либо одной таблице.
- Объект последовательности генерирует последовательные значения, которые можно использовать в любом операторе SQL.
- Поскольку объект последовательности может использоваться любой прикладной программой, есть два выражения, управляющие получением либо очередного значения в данной последовательности, либо значения, сгенерированного перед выполнением данного оператора. Выражение PREVVAL возвращает последнее значение, сгенерированное для данной последовательности предыдущим оператором в текущем сеансе. Выражение NEXTVAL возвращает следующее значение данной последовательности. Использование этих выражений допускает, чтобы одно и то же значение использовалось для разных операторов SQL в разных таблицах.

Хотя это и не все особенности этих двух подходов, изложенное поможет вам определить, какой из них использовать с учетом структуры вашей базы данных и использующих базу данных прикладных программ.

Создание индекса, расширения индекса или спецификации индекса

В разделе, озаглавленном “Создание индекса, расширения индекса или спецификации индекса”, к абзацу, начинающемуся предложением “Длина столбца, входящий в ключ индекса, не может быть больше 255 байт” добавьте следующее примечание:

Примечание: Переменная реестра DB2_INDEX_2BYTEVARLEN дает разрешение включать в состав ключа индекса столбцы, длина которых превышает 255.

Глава 4. Изменение базы данных

В разделе “Изменение табличного пространства” надо добавить следующие новые разделы:

Добавление к табличному пространству SMS контейнера на разделе

К табличному пространству SMS можно добавить контейнер на разделе (или узле), который в настоящее время не содержит контейнеров.

Содержимое табличного пространства равномерно распределяется по всем контейнерам. В процессе этого перераспределения доступ к табличному пространству не ограничивается. Если нужно добавить несколько контейнеров, все их необходимо добавлять одновременно.

Чтобы добавить контейнер к табличному пространству SMS с помощью командной строки, введите команду:

```
ALTER TABLESPACE <имя>
ADD ('<путь>')
ON NODE (<номер_раздела>)
```

Раздел задается его номером; каждый раздел (или узел) в этом диапазоне разделов должен существовать в группе узлов, в которой определено это табличное пространство. номер_раздела можно задавать только явно или внутри ровно одного условия *ON NODES* для оператора.

В следующем примере показано, как добавить новый контейнер на третий раздел группы узлов, используемой табличным пространством “plans” в операционной системе на основе UNIX:

```
ALTER TABLESPACE plans
ADD ('/dev/rhdisk0')
ON NODE (3)
```

После раздела под заголовком “Изменение атрибутов таблиц” добавьте следующие разделы:

Изменение столбца идентификации

Для изменения атрибутов существующего столбца идентификации используйте оператор ALTER TABLE. Дополнительную информацию об этом операторе, включая его синтаксис, смотрите в справочнике *SQL Reference*.

Есть разные способы изменить столбец идентификации, чтобы придать ему некоторые свойства последовательностей.

Некоторые задачи уникальны для ALTER TABLE и столбца идентификации:

- RESTART сбрасывает последовательность, связанную со столбцом идентификации, к значению, неявно или явно заданному в качестве начального значения первоначально при создании столбца идентификации.
- RESTART WITH <числовая-константа> сбрасывает последовательность, связанную со столбцом идентификации, к значению числовой константы точного типа. Числовая константа может представлять собой любое положительное или отрицательное значение без ненулевых цифр после десятичной точки, которое может быть присвоено столбцу идентификации.

Изменение последовательности

Для изменения атрибутов существующей последовательности используется оператор ALTER SEQUENCE. Дополнительную информацию об этом операторе, включая его синтаксис, смотрите в справочнике *SQL Reference*.

Возможны следующие изменения атрибутов последовательности:

- Изменение приращения между будущими значениями
- Установка новых минимальных или максимальных значений
- Изменение числа кэшируемых членов последовательности
- Установка или отмена циклического повторения последовательности
- Установка или отмена генерации членов последовательности по запросу
- Перезапуск последовательности

Есть две задачи, не встречающиеся в ходе создания последовательности. Это:

- RESTART. Сбрасывает последовательность к значению, неявно или явно заданному в качестве начального значения при создании последовательности.
- RESTART WITH числовая-константа. Сбрасывает последовательность к значению числовой константы точного типа. Числовая константа может представлять собой любое положительное или отрицательное значение без ненулевых цифр после десятичной точки.

После перезапуска последовательности или превращения ее в цикл возникает опасность генерации повторных членов последовательности. Оператор ALTER SEQUENCE влияет только на будущие члены последовательности.

Тип данных последовательности изменять нельзя. Вместо этого нужно отбросить существующую последовательность и затем создать новую последовательность, выбрав новый тип данных.

Все кэшируемые значения последовательности, не используемые DB2, при изменении последовательности теряются.

Отбрасывание последовательности

Чтобы удалить последовательность, используйте оператор DROP. Дополнительную информацию об этом операторе, включая его синтаксис, смотрите в справочнике *SQL Reference*.

Конкретную последовательность можно отбросить так:

```
DROP SEQUENCE имя_последовательности
```

где имя_последовательности - имя отбрасываемой последовательности, включая имя явной или неявной схемы для точного указания существующей последовательности.

Последовательности, создаваемые системой для столбцов идентификации (IDENTITY), нельзя отбрасывать при помощи оператора DROP SEQUENCE.

После того, как последовательность отброшена, все привилегии в отношении этой последовательности также отбрасываются.

Переключение состояния табличного пространства

Если контейнеры, связанные с табличным пространством, вновь стали доступными, можно использовать условие SWITCH ONLINE оператора ALTER TABLESPACE,

чтобы перевести это табличное пространство из состояния OFFLINE в состояние ONLINE. Табличное пространство переводится в состояние ONLINE, в то время как остальная база данных остается в рабочем состоянии и продолжает использоваться.

Вместо того, чтобы использовать это условие, можно отсоединить от базы данных все прикладные программы и затем вновь установить соединения прикладных программ с базой данных. При этом табличное пространство будет переведено из состояния OFFLINE в состояние ONLINE.

Чтобы переключить табличное пространство в состояние ONLINE из командной строки, введите команду:

```
ALTER TABLESPACE <имя>  
SWITCH ONLINE
```

Изменение контейнеров в табличном пространстве DMS

Табличные пространства DMS теперь создаются и изменяют размеры параллельно, что повышает производительность. Уровень параллелизма равен числу предварительных выборов плюс 1.

Глава 5. Управление доступом к базам данных

После раздела под заголовком “Привилегии индексов” добавьте следующие разделы:

Привилегии последовательностей

Создатель последовательности автоматически получает привилегию USAGE. Привилегия USAGE позволяет использовать выражения NEXTVAL и PREVVVAL для этой последовательности. Чтобы разрешить другим пользователям использовать выражения NEXTVAL и PREVVVAL, привилегии последовательности надо предоставить всем (то есть группе PUBLIC). Это позволит всем пользователям использовать выражения для данной последовательности.

После раздела под заголовком “Управление доступом к данным с помощью утилиты аудита” добавьте следующий раздел:

Шифрование данных

Составной частью плана защиты может быть шифрование данных. Для этого можно использовать встроенные функции шифрования и дешифровки: ENCRYPT, DECRYPT_BIN, DECRYPT_CHAR и GETHINT. Дополнительную информацию об этих функциях, включая их синтаксис, смотрите в разделе Справочник SQL Замечаний по выпуску.

Функция ENCRYPT шифрует данные при помощи метода шифрации на основе паролей. Эти функции также позволяют встроить подсказку к паролю. Подсказка к паролю встраивается в зашифрованные данные. После шифрования единственная возможность расшифровки данных - это использование точного пароля. Разработчики, решившие использовать эти функции, должны запланировать мероприятия в случае забытых паролей и недоступных данных.

Результат функций ENCRYPT имеет тот же тип данных, что и первый аргумент.

Шифрование возможно только для длинных переменных (VARCHAR).

Объявляемая длина результата - одна из следующих:

- Длина аргумента данных плюс 42, если задан необязательный параметр hint (подсказка).
- Длина аргумента данных плюс 10, если не задан необязательный параметр hint.

Функции DECRYPT_BIN и DECRYPT_CHAR расшифровывают данные при помощи дешифровки, основанной на пароле.

Результат функций DECRYPT_BIN и DECRYPT_CHAR имеет тот же тип данных, что и первый аргумент.

Объявляемая длина результата равна длине первоначальных данных.

Функция GETHINT возвращает встроенную подсказку к паролю. Подсказка к паролю - это фраза, которая помогает владельцам данных вспоминать пароли. Например, слово “океан” можно использовать как подсказку, помогающую вспомнить пароль “Тихий”.

Пароль, используемый для шифрации данных, определяется при помощи одного из двух способов:

- Аргумент Password. Пароль - это строка, явно передаваемая при вызове функции ENCRYPT. Данные будут шифроваться и дешифроваться при помощи данного пароля.
- Пароль из специального реестра. Оператор SET ENCRYPTION PASSWORD шифрует значение пароля и посылает зашифрованный пароль менеджеру баз данных, который сохраняет его в специальном реестре. Функции ENCRYPT, DECRYPT_BIN и DECRYPT_CHAR, вызванные без параметра пароля, используют значение специального реестра ENCRYPTION PASSWORD.

Первоначальное значение специального реестра или значение по умолчанию - пустая строка.

Допустимые длины паролей - от 6 до 127 включительно. Допустимые длины подсказок - от 0 до 32 включительно.

Когда специальный реестр ENCRYPTION PASSWORD задается со стороны клиента, пароль шифруется у клиента, посылается серверу базы данных, а затем расшифровывается. Чтобы не оставлять пароль в читаемом виде, его повторно шифруют на сервере базы данных. Функции DECRYPT_BIN и DECRYPT_CHAR должны расшифровывать специальный реестр перед использованием. Значение, находящееся в ENCRYPTION PASSWORD, также не оставляется в читаемом виде. Защита шлюза не поддерживается.

Глава 8. Восстановление базы данных

В разделе “Tivoli Storage Manager”, подраздел “Управление резервными копиями и архивами журналов в TSM” в третьем абзаце непосредственно перед “Примерами использования db2adutl:” в последнем предложении в правой части страницы пропущена следующая информация:

Можно также задать эту команду с параметром OLDER [THAN] <отметка_времени> или <n> DAYS. Это позволит удалить резервные копии, сделанные ранее указанной даты (*отметка_времени*) или указанного числа дней. Можно также выбрать диапазон выводимых журналов, а не просматривать их все. Кроме того, можно выбрать конкретную резервную копию для удаления, задав условие TAKEN AT <отметка_времени>.

Как использовать приостановленный ввод-вывод

В главу 8 “Восстановление базы данных” надо добавить следующий новый раздел по использованию функции приостановленного ввода/вывода:

С DB2 поставляется новый инструмент `db2inidb`, который позволяет выполнить восстановление после аварии, перевести базу данных в состояние отложенного повтора транзакций и выполнить для нее восстановление с повтором транзакций.

Приостановленный ввод-вывод поддерживает постоянную доступность системы, полностью реализуя оперативное отделение зеркальной копии, то есть отделение зеркальной копии без закрытия базы данных. Если пользователь не может получить автономные или оперативные резервные копии большой базы данных, резервные копии или копии системы можно получить из ее зеркальной копии, выполнив приостановку ввода-вывода и отделение зеркальной копии.

Варианты использования `db2inidb` зависят от способа зеркального копирования устройств хранения. В следующих вариантах использования предполагается, что вся база данных копируется зеркально, согласованно с системой хранения.

В многоузловой среде, прежде чем можно будет использовать отделенный образ на любом из разделов, нужно запустить инструмент `db2inidb` на каждом разделе. Инструмент `db2inidb` можно запустить на всех разделах одновременно.

1. Создание клона базы данных

Цель этой операции - создание копии исходной базы данных, используемой только для чтения. Следующие процедуры описывают, как получить клон базы данных:

- a. Приостановите ввод-вывод в исходной системе, введя следующую команду:
`db2 set write suspend for database`
- b. Воспользуйтесь командой уровня операционной системы, чтобы отделить зеркальную копию от исходной базы данных.
- c. Возобновите ввод-вывод в исходной системе, введя следующую команду:
`db2 set write resume for database`

После выполнения этой команды база данных в исходной системе должна вернуться в обычное состояние.

- d. Подключитесь к зеркальной копии базы данных с другого компьютера.
- e. Запустите этот экземпляр базы данных, введя команду:
`db2start`
- f. Запустите восстановление после аварии DB2, введя следующую команду:
`db2inidb имя_базы_данных AS SNAPSHOT`

Примечание: Эта команда выполнит откат изменений, внесенных транзакциями, начатыми во время отделения зеркальной копии.

Никакие резервные копии, снятые с клона базы данных, нельзя использовать для восстановления с повтором транзакций исходной базы данных с использованием файлов журналов, полученных в исходной базе данных после отделения зеркальной копии.

2. Использование отделенной зеркальной копии в качестве резервной базы данных

Так как для зеркальной (резервной) базы данных непрерывно выполняется повтор транзакций с использованием журналов, новые журналы, которые создаются в

исходной базе данных, постоянно считываются из исходной системы. Следующая процедура описывает, как отделенная зеркальная копия может использоваться в качестве резервной базы данных:

- a. Приостановите в исходной базе данных операции записи ввода-вывода.
 - b. Отделите зеркальную копию от исходной системы.
 - c. Возобновите в исходной базе данных запись ввода-вывода, чтобы вернуть ее в обычный режим.
 - d. Присоедините зеркальную копию базы данных к другому экземпляру.
 - e. Скопируйте журналы, установив программу обработчика пользователя для получения файлов журналов из исходной системы, чтобы гарантировать доступность для данной зеркальной копии базы данных самых новых журналов.
 - f. Переведите зеркальную копию в состояние отложенного повтора и выполните для нее повтор транзакций. Запустите инструмент `db2inidb (db2inidb <алиас_базы_данных> as standby)`, чтобы перевести базу данных в состояние отложенного повтора, отменить состояние приостановленной записи и выполнить повтор транзакций до окончания журналов.
 - g. Вернитесь к шагу e и повторяйте данный процесс, пока исходная база данных не будет закрыта.
3. Использование отделенной зеркальной копии в качестве резервной копии
- Следующая процедура описывает, как использовать зеркальную копию системы в качестве резервной копии для восстановления исходной системы:
- a. Воспользуйтесь командами операционной системы, чтобы скопировать данные и журналы зеркальной копии на место первичной системы.
 - b. Запустите этот экземпляр базы данных, введя команду:
`db2start`
 - c. Чтобы перевести зеркальную базу данных в состояние отложенного повтора, отменить состояние приостановленной записи и выполнить повтор транзакций до окончания журналов, введите следующую команду:
`db2inidb алиас_базы_данных AS MIRROR`

Инкрементное резервное копирование и восстановление

В главу 8 “Восстановление базы данных” вставьте следующий раздел об инкрементном резервном копировании и восстановлении:

По мере того, как размер баз данных, особенно хранилищ, продолжает расширяться до терабайтных и петабайтных масштабов, одновременно растут время и аппаратные ресурсы, требуемые для резервного копирования и восстановления этих баз данных. При работе с большими базами данных полное резервное копирование баз данных и табличных пространств не всегда оказывается лучшим подходом, поскольку требования к устройствам хранения для многочисленных копий таких баз данных оказываются непомерными. Рассмотрим следующие соображения:

- Когда меняется лишь малый процент данных в хранилище, должна быть возможность обходиться без резервного копирования всей базы данных.
- Добавление табличных пространств к существующим базам данных и последующее резервное копирование только табличного пространства рискованно, поскольку могут измениться данные вне копируемых табличных пространств.

Теперь DB2 поддерживает инкрементное резервное копирование и восстановление (за исключением данных длинных полей и больших объектов). *Инкрементная резервная копия* - это резервная копия, содержащая лишь те страницы, которые были изменены после снятия предыдущей резервной копии. Помимо страниц с измененными

данными и индексом, каждая инкрементная резервная копия содержит все метаданные первоначальной базы данных (такие как конфигурация базы данных, определения табличных пространств, хронологию базы данных и так далее), которые сохраняются при обычном полном резервном копировании.

Поддерживаются два типа инкрементного резервного копирования:

- *Инкрементный*. Инкрементная резервная копия - это копия всех данных базы данных, которые изменились со времени последней успешной операции полного резервного копирования. Ее называют также кумулятивной резервной копией, поскольку каждая следующая инкрементная резервная копия будет содержать информацию всех предыдущих. Предшественником инкрементной резервной копии всегда является последняя успешная полная резервная копия того же объекта.
- *Разностный*. Разностная (или инкрементная разностная) копия, - это копия всех данных базы данных, которые изменились со времени последней успешной операции резервного копирования (полного, инкрементного или типа дельта) рассматриваемого табличного пространства. Она называется также дифференциальной резервной копией. Предшественником разностной резервной копии является последняя успешная резервная копия со всеми табличными пространствами, входящими в данную разностную резервную копию.

Главное различие между инкрементной и разностной копиями заключается в их поведении при снятии последовательных резервных копий объекта, непрерывно менявшегося в течение какого-то времени. Каждая следующая инкрементная копия включает все содержимое предыдущей инкрементной копии плюс все новые или изменившиеся с момента изготовления предыдущей копии данные. Разностные резервные копии содержат только страницы, изменившиеся с момента изготовления предыдущей копии.

Допускается сочетание инкрементного резервного копирования базы данных и табличного пространства, как в оперативном режиме, так и в автономном. Тщательно планируйте стратегию резервного копирования, поскольку сочетание инкрементного резервного копирования базы данных и табличного пространства приводит к тому, что предшественник резервной копии базы данных (или резервной копии нескольких табличных пространств) не всегда представляет собой единую копию, а может оказаться уникальным набором сделанных в разное время предыдущих резервных копий базы данных и табличных пространств.

Чтобы восстановить базу данных или табличное пространство до согласованного состояния, процесс восстановления должен начаться с согласованной резервной копии всего воссоздаваемого объекта (базы данных или табличного пространства) и должен затем использовать все соответствующие инкрементные копии в описанном ниже порядке (смотрите раздел "Метод восстановления").

Чтобы разрешить отслеживание изменений базы данных, DB2 поддерживает новый параметр конфигурации базы данных TRACKMOD, который может иметь одно из двух допустимых значений:

- NO. Инкрементное резервное копирование в этой конфигурации не разрешено. Изменения страниц баз данных не отслеживаются и никак не регистрируются.
- YES. Инкрементное резервное копирование в этой конфигурации разрешено. Когда разрешается отслеживание изменений, новая конфигурация вступает в силу при первом успешном соединении с любой базой данных экземпляра. До первого инкрементного резервного копирования должно пройти полное резервное копирование базы данных.

Значение TRACKMOD по умолчанию для существующих баз данных - NO; для новых баз данных - YES.

Уровень отслеживания в этом случае - табличное пространства (как для SMS, так и для DMS).

Даже минимальное отслеживание изменений в базе данных может сказаться на производительности при выполнении транзакций, связанных с изменениями или вставкой данных.

Восстановление из инкрементных резервных копий

Операция восстановления из инкрементных резервных копий всегда содержит следующие шаги:

1. Определение инкрементной копии назначения. Сначала администратор базы данных должен найти последнюю копию, подлежащую восстановлению, и из утилиты восстановления DB2 запросить операцию инкрементного восстановления. Этот образ при инкрементном восстановлении называется целевым образом, поскольку он будет последним из восстановленных образов. Команда инкрементного восстановления для этого образа может инициировать создание новой базы данных, конфигурация и определения табличных пространств которой будут взяты из этой целевой копии. Инкрементная целевая копия задается при помощи параметра TAKEN AT в команде RESTORE DATABASE.
2. Восстановление наиболее поздней полной базы данных или табличного пространства в качестве отправной точки, к которой можно применять последовательные инкрементные резервные копии.
3. Восстановление каждой из требуемых инкрементных резервных полных копий или копий табличного пространства в том порядке, в котором они создавались, начиная с исходной копии, восстановленной на шаге 2.
4. Повторение шага 3, пока не будет вторично прочитана копия назначения из шага 1. Всего в ходе операции инкрементного восстановления происходит два обращения к целевой копии. Во время первого обращения из копии считываются только начальные данные; никакие пользовательские данные не считываются. Полная копия считывается и обрабатывается только во время второго обращения.

Два обращения к целевой копии при операции инкрементного восстановления нужны для того, чтобы убедиться в корректности изначально заданной хронологии, конфигурации базы данных и определений табличных пространств для базы данных, которая будет создана в ходе операции восстановления. В тех случаях, когда табличное пространство отбрасывалось после снятия первоначальной полной копии базы данных, данные табличного пространства для этой копии будут считываться из резервных копий, но игнорироваться в ходе инкрементного восстановления.

Чтобы восстановить набор инкрементных резервных копий, задайте опцию TAKEN AT *отметка_времени* в команде RESTORE DATABASE. Задайте отметку времени для последней из копий, которые нужно восстановить.

Например:

```
1. db2 restore database sample incremental  
taken at <отметка>
```

где:
<отметка> указывает на последнюю инкрементную резервную копию, которую нужно восстановить

```
2. db2 restore database sample incremental taken at <отметка1>
```

где:

- <отметка1> указывает на начальную полную копию базы данных (или табличного пространства)
3. db2 restore database sample incremental taken at <отметкаХ>
- где:
<отметкаХ> указывает на каждую из инкрементных резервных копий в порядке их создания
4. Повторить шаг 3, восстанавливая все инкрементные резервные копии вплоть до копии <отметка>

Если предпринимается попытка операции восстановления базы данных и создавались инкрементные резервные образы табличного пространства, образы табличного пространства должны восстанавливаться в хронологическом порядке в соответствии их отметкам времени резервного копирования.

Параллельное восстановление

Теперь DB2 использует несколько агентов как при восстановлении после аварии, так и при восстановлении с повтором транзакций базы данных. Можно ожидать повышения производительности при этих операциях, особенно в симметричной многопроцессорной среде (SMP); использование нескольких агентов при восстановлении базы данных использует дополнительные процессоры, доступные на компьютерах SMP.

С этим усовершенствованием связан новый тип агента - db2agnsc. DB2 выбирает число агентов, используемых при восстановлении базы данных, на основе числа процессоров в компьютере. Для компьютеров SMP число используемых агентов равно числу процессоров + 1. На компьютерах с одним процессором несколько агентов используются для более производительного чтения журналов, обработки записей журналов и предварительной выборки страниц данных.

DB2 распределяет записи журнала между агентами так, чтобы они могли применять их одновременно, где возможно. Например, параллельно может идти обработка записей журнала, связанная с операциями вставки, удаления, изменения, добавления ключа и удаления ключа. Поскольку записи журнала делаются параллельно на уровне страниц (записи журнала на одной и той же странице данных обрабатываются одним агентом), производительность растет даже в том случае, когда вся работа выполняется на одной таблице.

Резервное копирование в именованные конвейеры

Теперь доступна поддержка резервного копирования базы данных в локальные именованные конвейеры (и восстановление из них) в системах на основе UNIX. Программа записи и программа чтения именованного конвейера должны находиться на одном и том же компьютере. Конвейер должен существовать и располагаться в локальной файловой системе. Поскольку именованный конвейер обрабатывается как локальное устройство, нет необходимости указывать, что назначение является именованным конвейером. Вот пример для AIX:

1. Создайте именованный конвейер:
`mkfifo /u/dbuser/mypipe`
2. Используйте этот конвейер как назначение для операции резервного копирования базы данных:
`db2 backup db sample to /u/dbuser/mypipe`
3. Восстановите базу данных:
`db2 restore db sample into mynewdb from /u/dbuser/mypipe`

Резервное копирование из отдельной копии

Теперь DB2 поддерживает полное автономное резервное копирование базы данных на отделенной зеркальной копии базы данных. Оперативное резервное копирование не поддерживается и в нем нет необходимости, поскольку база данных, которая находится в состоянии отложенного повтора транзакций, все равно недоступна. Когда происходит восстановление отдельной зеркальной резервной копии, она нуждается в повторе транзакций, поскольку во время отделения могли быть активные транзакции.

Примечание: Для пакета исправления DB2 версии 7.1 FixPak 3 и DB2 версии 7.2 эта поддержка ограничена базами данных, которые содержат только табличные пространства DMS. При попытке резервного копирования базы данных после отделения и базы данных, содержащей табличные пространства SMS, резервное копирование не будет выполнено.

После того, как база данных была разделена, нужно задать при помощи утилиты **db2inidb** одну из следующих опций:

- **Snapshot** (снимок). Эта опция инициирует восстановление после аварии, приводя базу данных к согласованному состоянию. Запускается новая последовательность файлов журнала, и база данных не сможет повторять транзакции по каким-либо журналам, начинающимся с первоначальной базы данных. База данных доступна для всех операций, включая резервное копирование.
- **Standby** (Резервная база данных). Эта опция переводит базу данных в состояние отложенного повтора транзакций. Восстановление после аварии *не* выполняется, и база данных остается в несовместимом состоянии.
- **Mirror** (Зеркальная копия). Эта опция заменяет первоначальную базу данных на зеркальную копию. База данных переводится в состояние отложенного повтора транзакций, а режим WRITE SUSPEND выключается. Восстановление после аварии *не* выполняется, и база данных остается в несовместимом состоянии.

Вот несколько сценариев использования:

- Создание клона базы данных.

Цель этой операции состоит в том, чтобы иметь копию исходной базы данных, доступную только для чтения и используемую, например, для создания отчетов. Для этого выполните следующие действия:

1. Приостановите ввод/вывод в исходной системе:

```
db2 set write suspend for
database
```

2. Отделите зеркальную копию. Воспользуйтесь командами уровня операционной системы, чтобы отделить зеркальную копию от исходной базы данных.
3. Восстановите ввод/вывод в исходной системе:

```
db2 set write resume for database
```

База данных в исходной системе должна теперь вернуться в обычное состояние.

4. Смонтируйте отделенные зеркальные копии базы данных на другом хосте.

5. Запустите экземпляр:

```
db2start
```

6. Запустите восстановление после аварии DB2:

```
db2inidb <имя_базы_данных>
as snapshot
```

Тот же процесс можно использовать для автономного резервного копирования, но если восстановление происходит в исходной системе, эту резервную копию нельзя будет использовать для повтора транзакций, поскольку последовательность файлов журнала ей не соответствует.

- Использование отделенной зеркальной копии в качестве резервной базы данных.

Идея состоит в том, что для зеркальной (резервной) базы данных непрерывно выполняется повтор транзакций с использованием журналов, и даже новые журналы, которые создаются в исходной базе данных, непрерывно считываются из исходной системы. Чтобы использовать отделенную зеркальную копию в качестве резервной базы данных, выполните следующие действия:

1. Приостановите ввод/вывод в исходной системе:

```
db2 set write suspend for
database
```

2. Отделите зеркальную копию. Воспользуйтесь командами уровня операционной системы, чтобы отделить зеркальную копию от исходной базы данных.

3. Восстановите ввод/вывод в исходной системе:

```
db2 set write resume for database
```

База данных в исходной системе должна теперь вернуться в обычное состояние.

4. Смонтируйте отделенные зеркальные копии базы данных на другом хосте.

5. Скопируйте журналы. Установите программу обработчика пользователя для получения файлов журналов из архивного каталога исходной системы, чтобы гарантировать доступность для данной зеркальной копии базы данных самых новых журналов.

6. Переведите зеркальную базу данных в состояние отложенного повтора, отмените состояние приостановленной записи и выполните на зеркальной копии повтор транзакций до окончания журналов:

```
db2inidb <имя_базы_данных> as standby
db2 rollforward db <имя_базы_данных> to end of logs
```

7. Повторяйте процесс с шага 7, пока исходная база данных не придет в рабочее состояние.

- Использование отделенной зеркальной копии для восстановления исходной системы.

Следующая процедура описывает, как использовать зеркальную копию системы в качестве резервной копии для восстановления исходной системы:

1. Выполните полное копирование. Воспользуйтесь командами операционной системы, чтобы скопировать данные и журналы зеркальной копии на место первичной системы.

2. Запустите экземпляр:

```
db2start
```

3. Переведите восстановленную зеркальную базу данных в состояние отложенного повтора и выполните на зеркальной копии повтор транзакций до окончания журналов:

```
db2inidb
<имя_базы> as mirror
```

- Резервное копирование без выполнения восстановления после аварии.

Выполнение автономного резервного копирования на отделенной зеркальной базе данных без выполнения восстановления после аварии означает, что вы можете восстановить эту резервную копию поверх исходной системы. Для этого выполните следующие действия:

1. Приостановите ввод/вывод в исходной системе:

```
db2 set write suspend for
database
```

2. Отделите зеркальную копию. Воспользуйтесь командами уровня операционной системы, чтобы отделить зеркальную копию от исходной базы данных.
3. Восстановите ввод/вывод в исходной системе:

```
db2 set write resume for database
```

База данных в исходной системе должна теперь вернуться в обычное состояние.

4. Смонтируйте отделенные зеркальные копии базы данных на другом хосте.
5. Запустите экземпляр:

```
db2start
```

6. Переведите базу данных с зеркальными копиями в состояние отложенного повтора транзакций.

```
db2inidb <имя_базы_данных> as standby
```

7. Запустите операцию резервного копирования базы данных:

```
db2 backup database
<имя_базы_данных>
```

Это создаст неявное соединение с базой данных, но не вызовет восстановления после аварии DB2.

Архивирование журнала по требованию

Теперь DB2 поддерживает закрытие (и, если включена опция обработчика пользователя, то и архивирование) активного журнала восстановимой базы данных в любое время. Это дает возможность собрать полный комплект файлов журнала до известного момента, и затем использовать эти файлы журнала для изменения резервной базы данных.

Примечание: Архивирование журналов по требованию не гарантирует, что файлы журналов будут сархивированы немедленно; оно прерывает запись журнала и выдает запрос на архивирование, но обработчик пользователя может вызвать некоторую задержку

Вы можете инициировать архивирование журнала по требованию, введя новую команду DB2 ARCHIVE LOG или вызвав новый API **db2ArchiveLog**.

Создание зеркальной копии журнала

В главу 8 “Восстановление базы данных” следует добавить следующий новый раздел по использованию функции приостановленного ввода/вывода:

Теперь DB2 поддерживает создание зеркальной копии журнала на уровне базы данных. Создание зеркальной копии файлов журнала помогает защитить базу данных от таких опасностей, как:

- Случайное удаление активного журнала
- Разрушение данных из-за неисправности оборудования

Если вы опасаетесь, что ваши активные журналы могут быть повреждены (в результате аварии диска), вы можете при помощи новой переменной реестра DB2 - DB2_NEWLOGPATH2, - задать вторичный путь, чтобы база данных работала с копиями активного журнала, создавая зеркальные тома для записи журналов.

Переменная реестра DB2_NEWLOGPATH2 позволяет базе данных записывать идентичную вторую копию файлов журнала в другой путь. Рекомендуется

разместить вторичный путь на другом физическом диске (желательно с другим контроллером диска). Тогда контроллер диска не будет уязвимой точкой.

Примечание: Поскольку Windows NT и OS/2 не позволяют "монтировать" устройство с произвольным именем пути, на этих платформах невозможно задать вторичный путь на другом устройстве.

DB2_NEWLOGPATH2 может быть включена (задана равной 1) или выключена (задана равной 0). Значение по умолчанию - ноль. Если этой переменной присвоена 1, имя вторичного пути равно текущему значению переменной LOGPATH плюс символ 2. Например, в среде SMP, если LOGPATH равен /u/dbuser/sqllogdir/logpath, вторичный путь журнала будет /u/dbuser/sqllogdir/logpath2. В среде MPP, если LOGPATH равен /u/dbuser/sqllogdir/logpath, DB2 присоединит к пути указатель узла, и в качестве первичного пути журнала будет использовать /u/dbuser/sqllogdir/logpath/NODE0000. В этом случае вторичный путь журнала будет /u/dbuser/sqllogdir/logpath2/NODE0000.

При первом включении DB2_NEWLOGPATH2 вторичный путь журнала не используется, пока не будет завершен текущий файл журнала при следующем запуске базы данных. Это аналогично тому, как сейчас используется NEWLOGPATH.

Если возникает ошибка при записи в первичный или вторичный путь журнала, база данных пометит отказавший путь как "плохой", запишет сообщение в файл db2diag.log и будет помещать следующие записи журнала только в оставшийся "хороший" путь журнала. DB2 не будет пытаться еще раз использовать "плохой" путь, пока не будет завершен текущий файл журнала. Когда DB2 потребует открыть следующий файл журнала, она проверит, допустим ли этот путь, и если да, начнет использовать его. Если нет, DB2 не будет пытаться еще раз использовать путь до первого обращения к следующему файлу журнала. Попыток синхронизировать пути журналов не производится, но DB2 сохраняет информацию о происходящих ошибках доступа, чтобы использовать правильные пути при архивации файлов журнала. Если сбой произойдет при записи в оставшийся "хороший" путь, база данных прервет работу.

Поддержка кроссплатформенного резервного копирования и восстановления на Sun Solaris и HP

Теперь доступна поддержка for кроссплатформенного резервного копирования и восстановления между Sun Solaris и HP. Передача резервной копии между системами должна происходить в двоичном режиме. В системе назначения база данных должна быть создана с той же кодовой страницей/территорией, что и система, в которой была создана исходная база данных.

Особенности менеджера связей данных DB2/Особенности утилиты резервного копирования

Второй абзац в этом разделе замените на следующий текст:

Когда на файлы есть ссылки, серверы связей данных планируют их асинхронное копирование на архивный сервер, например, ADSM, или на диск. При работе утилиты резервного копирования DB2 проверяет, что скопированы все внесенные в расписание копирования файлы. В начале резервного копирования DB2 соединяется со всеми серверами связей данных, указанными в файле конфигурации DB2. Если у сервера связей данных есть один или несколько связанных файлов, и он не запущен или остановился во время операции резервного копирования, в резервной копии не будет полной информации DATALINK. Операция резервного копирования завершится успешно. Прежде чем сервер связей данных снова сможет быть помечен как доступный для базы данных, обработка всех ожидающих процессов резервного копирования должна успешно завершиться. Если резервное копирование будет инициировано в то время,

когда число резервных копирований, уже ожидающих завершения на сервере связей данных, вдвое превысит num_db_backups (см. ниже), операция резервного копирования завершается неудачно. Этот сервер связей данных должен быть перезапущен, а ожидающие операции резервного копирования завершены, прежде чем будут разрешены дополнительные резервные копирования.

Особенности менеджера связей данных DB2/Особенности утилиты восстановления и повтора транзакций

Абзацы, начинающиеся словами:

Если вы восстанавливаете из резервной копии базу данных или табличное пространство и не указали опцию WITHOUT DATALINK...

и

Когда вы восстанавливаете из резервной копии базу данных или табличное пространство и указали опцию WITHOUT DATALINK...

, надо заменить на:

Когда вы восстанавливаете базу данных или табличное пространство, для успешного завершения операции восстановления необходимы следующие условия:

- o Если не работает любой из серверов связей данных, записанных в файле резервной копии, операция восстановления все равно завершится успешно.

Таблицы с информацией о столбце DATALINK, пострадавшие из-за отсутствия сервера связей данных, после операции восстановления будут переведены в состояние отложенного согласования связей данных (или отложенного повтора транзакций - если выполнялся повтор транзакций).

Прежде чем серверы связей данных смогут быть снова помечены как доступные для базы данных, этот процесс восстановления должен успешно завершиться.

- o Если любой из серверов связей данных, записанных в файле резервной копии, остановится во время операции восстановления, операция восстановления завершится неудачно. Восстановление можно перезапустить при неработающем сервере связей данных (смотри выше).
- o Если предыдущая операция восстановления остается незавершенной на каких-либо серверах связей данных, последующие операции восстановления базы данных или табличных пространств будут завершаться неуспешно до тех пор, пока не будут перезапущены эти серверы связей данных и не будет завершено незавершенное восстановление.
- o Информация обо всех столбцах DATALINK, записанных в файле резервной копии, должна существовать в соответствующих регистрационных таблицах серверов связей данных.

Если в регистрационных таблицах не записано всей информации о столбцах DATALINK, после завершения операции восстановления (или отложенного повтора транзакций - если выполнялся повтор транзакций) таблица с отсутствующим столбцом DATALINK переводится в состояние невозможности согласовать связи данных.

Если резервная копия не записана в регистрационные таблицы, это может означать, что предоставленный файл резервной копии более ранний, чем значение для num_db_backups, и уже был помечен для "чистки мусора". Это означает, что архивированные файлы из этой более ранней резервной копии были удалены и не могут быть восстановлены. Все таблицы, у которых есть столбцы DATALINK, переводятся в состояние отложенного согласования связей данных.

Если резервная копия не записана в регистрационные таблицы, это может означать, что процесс резервного копирования до сих пор не завершен, поскольку не работает сервер связей данных. Все таблицы, у которых есть столбцы DATALINK, переводятся в состояние отложенного согласования связей данных. Когда сервер связей данных будет перезапущен, процесс резервного копирования будет завершен до процесса восстановления.

Таблица остается доступной для пользователей, но ссылки в столбцах DATALINK

могут быть неверными (например, файл, заданный значением столбца DATALINK, может быть не найден). Если вы не хотите этого, можно перевести таблицу в состояние ожидания проверки, введя оператор "SET CONSTRAINTS for имя_таблицы TO DATALINK RECONCILE PENDING".

Если после операции восстановления из резервной копии таблица находится в состоянии невозможности согласования связей данных, данные столбца DATALINK можно исправить одним из способов, предлагаемых в разделе "Вывод таблицы из состояния невозможности согласования связей данных (Datalink_Reconcile_Not_Possible State)".

Примечание в конце первого абзаца остается прежним.

В конце этого раздела добавьте следующую информацию:

Настоятельно

рекомендуется архивировать файл datalink.cfg, чтобы предусмотреть некоторые необычные случаи восстановления, поскольку файл datalink.cfg в резервной копии базы данных лишь отражает файл datalink.cfg, каким он был на момент резервного копирования. В некоторых случаях восстановления требуется иметь наиболее поздний файл datalink.cfg. После каждого вызова команды ADD DATALINKS MANAGER или DROP DATALINKS MANAGER. Это поможет получить наиболее поздний файл datalink.cfg, если самый свежий файл datalink.cfg на диске недоступен.

Таким обр

Если наиболее поздний файл datalink.cfg на диске недоступен, замените существующий файл datalink.cfg (восстановленный из резервной копии) наиболее поздним файлом datalink.cfg, архивированным до запуска восстановления повтором транзакций. Сделайте это после восстановления базы данных.

Восстановление баз данных из резервных копий, сделанных в автономном режиме, без повтора транзакций

Восстановление без повтора транзакций можно выполнять только на уровне базы данных, но не на уровне табличного пространства. Восстановить базу данных без повтора транзакций можно, либо задав базу данных, для которой невозможен повтор транзакций (то есть использующую циклическую запись журнала), либо указав параметр WITHOUT ROLLING FORWARD в команде RESTORE DATABASE.

Если вы используете утилиту восстановления с опцией WITHOUT DATALINK, все таблицы со столбцами DATALINK переводятся в состояние отложенного согласования связей (DRP), а во время операции восстановления никакого согласования с серверами связей данных не выполняется.

Если вы не используете опцию WITHOUT DATALINK, а сервер связей данных, записанный в файл резервной копии, больше не определен для базы данных (то есть был отброшен при помощи команды DROP DATALINKS MANAGER), те таблицы, что содержали ссылки DATALINK на отброшенный сервер связей данных, утилитой восстановления переводятся в состояние DRP.

Если вы не используете опцию WITHOUT DATALINK, все серверы связей данных доступны и вся информация о столбцах DATALINK полностью записана в регистрационные таблицы, то для всех серверов связей данных, записанных в файл резервной копии, происходит следующее:

- Все файлы, которые были связаны после образа резервной копии, использованной для операции восстановления базы данных, помечаются как несвязанные (поскольку они не записаны в образ резервной копии как связанные).
- Все файлы, ссылки на которые были удалены после снятия образа резервной копии, но имелись до снятия образа резервной копии, помечаются как связанные (поскольку они записаны в образ резервной копии как связанные). Если файл

впоследствии был связан с другой таблицей в другой базе данных, восстановленная таблица переводится в состояние отложенного согласования связей данных.

Примечание: Описанные шаги невозможны, если резервная копия, использованная для операции восстановления базы данных, была снята в момент, когда не работал хотя бы один сервер связей данных, поскольку информация DATALINK в резервной копии не полна. Описанные шаги также невыполнимы, если резервная копия, использованная для операции восстановления базы данных, была снята после восстановления базы данных с повтором транзакций или без него. В обоих случаях все таблицы со столбцами DATALINK переводятся в состояние отложенного согласования связей, а во время операции восстановления никакого согласования с серверами связей данных не выполняется.

Восстановление баз данных и табличных пространств и повтор транзакций до конца файлов журнала

При восстановлении базы данных или табличного пространства с последующим повтором транзакций до конца файлов журнала (при условии, что все файлы доступны) проверка синхронизации не требуется, если только во время операции восстановления нет ни одного неработающего сервера связей данных, записанного в файле резервной копии. Если вы не уверены, что для операции повтора транзакций есть все файлы журнала, или полагаете, что может потребоваться синхронизация значений DATALINK, сделайте следующее:

1. Введите оператор SQL для затронутых таблиц:

```
SET CONSTRAINTS FOR имя_таблицы  
TO DATALINK RECONCILE PENDING
```

Это переведет таблицу в состояние отложенного согласования связей данных и в состояние отложенной проверки.

2. Если вы не хотите, чтобы таблица находилась в состоянии отложенной проверки, введите следующий оператор SQL:

```
SET CONSTRAINTS FOR имя_таблицы IMMEDIATE CHECKED
```

Это выведет таблицу из состояния отложенной проверки, но оставит ее в состоянии отложенного согласования связей данных. Для вывода таблицы из этого состояния необходимо использовать утилиту согласования.

Может оказаться, что файл резервной копии содержит ссылки DATALINK на менеджер связей DB2 (то есть менеджер связей DB2 был зарегистрирован в базе данных во время резервного копирования), который в дальнейшем был отброшен от базы данных. Для каждого табличного пространства, для которого выполняется восстановление с повтором транзакций и которое содержит хотя бы одну таблицу со ссылками DATALINK на отброшенный менеджер связей DB2, все таблицы утилитой повтора транзакций переводятся в состояние DRP.

Взаимодействие менеджера связей данных и восстановления

В следующей таблице показаны разные типы восстановления, которые можно выполнить; действия менеджера связей данных DB2 во время восстановления и повтора транзакций, и необходимо ли запускать утилиту согласования после завершения восстановления:

Тип восстановления	Действия менеджера связей во время восстановления	Действия менеджера связей во время повтора транзакций	Согласование
Невосстановимая база данных (<i>logretain=NO</i>)			
Восстановление базы данных из полной резервной копии при работе всех серверов связей данных	Выполняется быстрое согласование	Нет	Может быть запущена дополнительно при подозрении на ошибки связывания файлов
Восстановление базы данных с использованием опции WITHOUT DATALINK	Таблицы переводятся в состояние <i>Datalink_Reconcile_Pending</i> (ожидание согласования связей данных)	Нет	Обязательно
Восстановление базы данных из полной резервной копии, когда хотя бы один из серверов связей данных не работает	Быстрое согласование выполняется лишь на тех таблицах в табличных пространствах, у которых нет связей с неработающим сервером связей данных, тогда как остальные таблицы переводятся в состояние <i>Datalink_Reconcile_Pending</i> (ожидание согласования связей данных)	Нет	Требуется для таблиц в табличных пространствах со связями с неработающим сервером связей данных
Восстановление базы данных из неполной резервной копии при работе всех серверов связей данных	Быстрое согласование не производится, все таблицы со столбцами DATALINK переводятся в состояние <i>Datalink_Reconcile_Pending state</i> (ожидание согласования связей данных)	Нет	Обязательно
Восстановимая база данных (<i>logretain=YES</i>)			
Восстановление базы данных с использованием опции WITHOUT ROLLING FORWARD из полной резервной копии при работе всех серверов связей данных	Выполняется быстрое согласование	Нет	Необязательно
Восстановление базы данных с использованием опций WITHOUT ROLLING FORWARD и WITHOUT DATALINK из полной или неполной резервной копии при работающих или неработающих серверах связей данных	Таблицы переводятся в состояние <i>Datalink_Reconcile_Pending</i> (ожидание согласования связей данных)	Нет	Обязательно
Восстановление базы данных с использованием опции WITHOUT ROLLING FORWARD из полной резервной копии, когда хотя бы один из серверов связей данных не работает	Быстрое согласование выполняется лишь на тех таблицах в табличных пространствах, у которых нет связей с неработающими серверами связей данных, тогда как остальные таблицы переводятся в состояние <i>Datalink_Reconcile_Pending</i> (ожидание согласования связей данных)	Нет	Требуется на таблицах в табличных пространствах со связями с неработающими серверами связей данных
Восстановление базы данных с использованием опции WITHOUT ROLLING FORWARD из неполной резервной копии при работающих или неработающих серверах связей данных	Быстрое согласование не производится, все таблицы со столбцами DATALINK переводятся в состояние <i>Datalink_Reconcile_Pending state</i> (ожидание согласования связей данных)	Нет	Обязательно
Восстановление базы данных и повтор транзакций до конца журналов из полной резервной копии при работе всех серверов связей данных	Никаких действий	Никаких действий	Необязательно
Восстановление базы данных и повтор транзакций до конца журналов из полной резервной копии, когда при выполнении повтора транзакций хотя бы один из серверов связей данных не работает	Никаких действий	Никаких действий	Необязательно

Тип восстановления	Действия менеджера связей во время восстановления	Действия менеджера связей во время повтора транзакций	Согласование
Восстановление базы данных и повтор транзакций до конца журналов из полной или неполной резервной копии, когда во время восстановления любой из серверов связей данных не работает	Никаких действий	Все таблицы со столбцами DATALINK переводятся в состояние Datalink_Reconcile_Pending state (ожидание согласования связей данных)	Требуется для всех таблиц со столбцами DATALINK
Восстановление базы данных и повтор транзакций до конца журналов из неполной резервной копии при работе всех серверов связей данных во время восстановления	Никаких действий	Никаких действий	Необязательно
Восстановление базы данных и повтор транзакций до конца журналов из полной или неполной резервной копии при работе серверов связей данных, когда на любом из серверов связей данных резервная копия неизвестна	Никаких действий	Все таблицы в табличных пространствах со связями с сервером связей данных, на котором резервная копия неизвестна, переводятся в состояние Datalink_Reconcile_Pending (ожидание согласования связей данных)	Обязательно
Восстановление табличного пространства и повтор транзакций до конца журналов из полной резервной копии при работе всех серверов связей данных	Никаких действий	Никаких действий	Необязательно
Восстановление табличного пространства и повтор транзакций до конца журналов из полной резервной копии, когда при выполнении повтора транзакций хотя бы один из серверов связей данных не работает	Никаких действий	Никаких действий	Необязательно
Восстановление табличного пространства и повтор транзакций до конца журналов из полной или неполной резервной копии, когда во время восстановления любой из серверов связей данных не работает	Никаких действий	Все таблицы в табличных пространствах со связями с любым неработающим сервером связей данных переводятся в состояние Datalink_Reconcile_Pending (ожидание согласования связей данных)	Требуется для таблиц в табличных пространствах со связями с любым неработающим сервером связей данных
Восстановление табличного пространства и повтор транзакций до конца журналов из неполной резервной копии при работе всех серверов связей данных	Никаких действий	Никаких действий	Необязательно
Восстановление базы данных и повтор транзакций до определенного момента времени из полной или неполной резервной копии при работающих или не работающих серверах связей данных во время процесса восстановления и/или повтора транзакций	Никаких действий	Таблицы переводятся в состояние Datalink_Reconcile_Pending (ожидание согласования связей данных)	Обязательно
Восстановление табличного пространства и повтор транзакций до определенного момента времени из полной или неполной резервной копии при работающих или не работающих серверах связей данных во время процесса восстановления и/или повтора транзакций	Никаких действий	Таблицы переводятся в состояние Datalink_Reconcile_Pending (ожидание согласования связей данных)	Обязательно
Восстановление базы данных с другим именем базы данных, алиасом, именем хоста или экземпляра без повтора транзакций ¹ на стр. 72	Таблицы переводятся в состояние Datalink_Reconcile_Not_Possible (согласование связей данных невозможно)	Нет	Необязательно, но таблицы в состоянии Datalink_Reconcile_Not_Possible (согласование связей данных невозможно) нужно исправить вручную

Тип восстановления	Действия менеджера связей во время восстановления	Действия менеджера связей во время повтор транзакций	Согласование
Восстановление базы данных с другим именем базы данных, алиасом, именем хоста или экземпляра и повтор транзакций	Никаких действий	Таблицы переводятся в состояние <i>Datalink_Reconcile_Not_Possible</i> (согласование связей данных невозможно)	Необязательно, но таблицы в состоянии <i>Datalink_Reconcile_Not_Possible</i> (согласование связей данных невозможно) нужно исправить вручную
Восстановление базы данных из непригодной к использованию резервной копии (образ был удален при чистке мусора на сервере связей данных) без повтор транзакций 1, с опцией WITHOUT DATALINK или без нее	Таблицы переводятся в состояние <i>Datalink_Reconcile_Pending</i> (ожидание согласования связей данных)	Никаких действий	Обязательно
Восстановление базы данных из непригодной к использованию резервной копии (образ был удален при чистке мусора на сервере связей данных) и повтор транзакций, с опцией WITHOUT DATALINK или без нее	Никаких действий	Таблицы переводятся в состояние <i>Datalink_Reconcile_Pending</i> (ожидание согласования связей данных)	Обязательно
Восстановление табличного пространства из непригодной к использованию резервной копии (образ был удален при чистке мусора на сервере связей данных) и повтор транзакций	Никаких действий	Таблицы переводятся в состояние <i>Datalink_Reconcile_Pending</i> (ожидание согласования связей данных)	Обязательно

Примечания:

1. Восстановление с использованием резервной копии, сделанной в автономном режиме, и опции WITHOUT ROLLING FORWARD (*logretain* включен), или восстановление с использованием резервной копии, сделанной в автономном режиме (*logretain* выключен).
2. *Полная* резервная копия - это резервная копия, снятая при всех работающих серверах связей данных. *Неполная* резервная копия - это резервная копия, снятая, когда хотя бы один из серверов связей данных не работал.
3. Процесс быстрого согласования не производится, если резервная копия, использованная для операции восстановления базы данных, была снята после восстановления базы данных - с повтором транзакций или без него. В этом случае все таблицы со столбцами DATALINK переводятся в состояние *Datalink_Reconcile_Pending* (ожидание согласования связей данных)

Обнаружение ситуации, требующей согласования

Далее приводятся несколько ситуаций, в которых может потребоваться запуск утилиты согласования:

- Вся база данных восстановлена, и для нее произведен повтор транзакций до определенного момента времени. Поскольку для всей базы данных был выполнен повтор транзакций до принятой транзакции, ни одна из таблиц не будет находиться в состоянии ожидания проверки (из-за реляционных или проверочных ограничений). Все данные в базе данных приведены в согласованное состояние. Тем не менее, столбцы DATALINK могут быть не синхронизированы с метаданными в Менеджере связей данных DB2, и требуется согласование.

В этом случае таблицы с данными столбцов DATALINK уже будут в состоянии DRP (ожидание согласования связей данных). Для каждой из этих таблиц необходимо вызвать утилиту согласования.

- Один из серверов связей данных, на котором запущен менеджер связей DB2, теряет след своих метаданных. Это может произойти по нескольким причинам. Например:
 - После холодного старта сервера связей данных.

- После восстановления метаданных сервера связей данных до более раннего состояния.

В некоторых ситуациях, например, при выполнении операторов SQL UPDATE и DELETE, DB2 может быть в состоянии обнаружить ошибку метаданных на сервере связей. В таких ситуациях DB2 откажется обработать оператор SQL. Вам придется перевести таблицу в состояние DRP (ожидание согласования связей данных) при помощи оператора SET CONSTRAINTS, а затем запустить утилиту согласования для этой таблицы.

- Файловая система недоступна (например, из-за отказа диска) и не восстанавливается до текущего состояния. В этой ситуации могут отсутствовать отдельные файлы.
- Менеджер связей DB2 отброшен в базе данных, но в столбце DATALINK FILE LINK CONTROL есть ссылки на этот менеджер связей DB2. Для таких таблиц следует запустить утилиту согласования.

Приложение С. Обработчик пользователя для восстановления базы данных

В разделе “Особенности архивирования и получения” следующий абзац теперь неверен, и его надо удалить:

Программа обработчика пользователя может быть прервана, если удаленный клиент потеряет свое соединение с сервером DB2. Это происходит, если пока обработчик пользователя обрабатывает архивирование журналов, один из подсоединенных клиентов SNA становится неработоспособным или выключается, в результате чего на сервер посылается сигнал (SIGUSR1). Сервер передает этот сигнал обработчику пользователя, что приводит к прерыванию. Программу обработчика пользователя можно изменить так, чтобы она проверила прерывание и продолжила обработку.

В разделе Обработка ошибок есть список Примечания, в котором содержание Примечания 3 должно быть изменено на следующее:

- Требования программы обработчика пользователя приостанавливаются на пять минут. В течение этого времени все требования игнорируются, включая требование файла журнала, из-за которого поступил этот код возврата.

После пятиминутной приостановки обработки требований обрабатывается следующее требование. Если при обработке этого запроса не возникли ошибки, продолжается обработка новых запросов программы обработчика пользователя и DB2 повторит запрос архивирования для файлов журналов, при архивировании которых ранее возникла ошибка или архивирование которых было приостановлено. Если во время повтора генерируется код возврата больше 8, требования приостанавливаются еще на пять минут. Пятиминутные приостановки продолжаются, пока не будет исправлена ошибка или не будет остановлена и перезапущена база данных.

Когда все прикладные программы отключены от базы данных и база данных повторно открыта, DB2 выдает запрос архивирования для всех файлов журналов, которые, возможно, не были успешно архивированы при предыдущем использовании базы данных.

Если программа обработчика пользователя не смогла заархивировать файлы журналов, диск может быть заполнен файлами журналов и дополнительные затраты на форматирование этих файлов журналов могут ухудшить производительность. Когда диск будет переполнен, менеджер баз данных не будет более принимать запросы прикладных программ на изменение баз данных.

Если программа обработчика пользователя вызывалась для получения файлов журнала, восстановление с повтором транзакций будет отложено, но не

остановлено, если только остановка не была задана в утилите ROLLFORWARD DATABASE. Если остановка не задана, вы можете исправить ошибку и продолжить восстановление.

Приложение I. Высокоскоростная межузловая связь

Следующий раздел был обновлен:

Подготовка DB2 к работе с использованием VI

Подробную информацию по установке смотрите в руководстве *DB2 Enterprise - Extended Edition for Windows Quick Beginnings*.

По завершении установки DB2 согласно описанию *DB2 Enterprise - Extended Edition for Windows Quick Beginnings* задайте следующие переменные реестра DB2 и выполните следующие задачи на каждом сервере раздела базы данных примера:

- Задайте DB2_VI_ENABLE=ON

Используйте команду **db2set**, чтобы изменить значение этой переменной реестра. Используйте команду **db2_all**, чтобы выполнить команду **db2set** на всех серверах разделов базы данных в этом экземпляре. Для выполнения команды **db2_all** вы должны зарегистрироваться с именем пользователя, входящим в группу администраторов.

В следующем примере внутрь двойных кавычек помещен символ ;, чтобы этот запрос мог выполняться одновременно на всех серверах разделов базы данных в этом экземпляре:

```
db2_all ";db2set DB2_VI_ENABLE=ON"
```

Дополнительную информацию о команде **db2_all** смотрите в разделе "Выполнение команд на нескольких серверах разделов базы данных" руководства *Administration Guide: Implementation*.

- Задайте DB2_VI_DEVICE=nic0

Например:

```
db2_all ";db2set DB2_VI_DEVICE=nic0"
```

Примечание: Для соединения Synfinity значение этой переменной должно быть DB2_VI_DEVICE=VINIC. Имя устройства (VINIC) должно быть введено в верхнем регистре.

- Задайте DB2_VI_VIPL=vipl.dll

Например:

```
db2_all ";db2set DB2_VI_VIPL=vipl.dll"
```

Примечание: В этом примере используется значение по умолчанию для этой переменной реестра. Дополнительную информацию об этих переменных реестра смотрите в книге *Administration Guide: Performance*.

- Введите db2start на экземпляре MPP.
- Просмотрите файл db2diag.log. Для каждого раздела должно быть одно сообщение, что "VI включен".
- Может потребоваться обновление параметров конфигурации FCM (Fast Communications Manager - менеджер быстрой связи). Если проблема заключается в ограниченности ресурсов, касающихся FCM, следует увеличить значения параметров настройки FCM. Если вы переходите из другой среды высокоскоростного соединения, для которой вы увеличили значения параметров

настройки FCM, может понадобиться уменьшить эти значения. Кроме того, в Windows NT может потребоваться перезадать переменную реестра DB2NTMEMSIZE, изменив заданное DB2 значение по умолчанию. Дополнительную информацию об этих переменных реестра смотрите в книге *Administration Guide: Performance*.

Глава 8. Руководство администратора: Производительность

Глава 3. Особенности прикладных программ

Задание уровня изоляции

В раздел “Задание уровня изоляции” добавьте следующие абзацы:

Помимо задания уровня изоляции на уровне пакетов при подготовке или связывании программы вы можете задать изоляцию на уровне операторов. Уровень изоляции для оператора задается при помощи условия WITH.

Изоляцию на уровне операторов поддерживают следующие операторы SQL:

- Оператор SELECT
- SELECT INTO
- DELETE с поиском
- INSERT
- UPDATE с поиском
- DECLARE CURSOR

При использовании изоляции уровня операторов существуют некоторые ограничения:

- Условие WITH нельзя использовать в подзапросах
- Опция WITH UR применяется только к операциям только для чтения. Если она используется в других ситуациях, уровень изоляции автоматически изменяется с “UR” на “CS”.
- Уровнем изоляции по умолчанию для оператора является уровень пакета, в котором этот оператор связан.
- Уровень изоляции оператора заменяет уровень изоляции для пакета, в котором появляется данный оператор.

Настройка класса оптимизации

В разделе “Настройка класса оптимизации” замените имеющуюся информацию для уровня 2 на следующую:

- 2 - Этот класс указывает программе оптимизации использовать степень оптимизации, которая существенно превосходит оптимизацию класса 1, но в то же время сохраняет стоимость компилирования сложных запросов на существенно более низком уровне, чем в классах 3 и старше. В частности:
- Используется вся доступная статистика, включая статистику частот и квантилей неравномерного распределения.
 - Применяются все правила перезаписи запросов, включая перенаправление запросов к таблицам сводок, за исключением правил с большим объемом вычислений, которые применяются только в очень редких случаях.
 - Используется “жадный” перебор объединений.
 - Рассматривается широкий диапазон методов доступа, включая предварительную выборку списка и перенаправление таблиц сводок.
 - Рассматривается, где это допустимо, стратегия объединения типа звезда.

Класс оптимизации 2 во всем аналогичен классу 5, за исключением того, что использует "жадный" перебор методов объединения вместо динамического программирования. В этом классе достигается максимальная оптимизация среди всех классов оптимизации, использующих алгоритм "жадного" перебора методов объединения, при котором рассматривается меньше вариантов для сложных запросов и, следовательно, тратится меньше времени на компиляцию, чем в классе 3 и выше. Поэтому он рекомендуется для очень сложных запросов в среде поддержки решений или среде OLAP (online analytic processing - оперативный анализ данных). В этих случаях велика вероятность, что один и тот же запрос будет повторяться нечасто, и его план доступа вряд ли останется в кэше до очередного повторения запроса.

Динамические составные операторы

Этот новый раздел следует добавить после раздела "Составной оператор SQL":

Динамический составной оператор группирует другие операторы SQL в выполняемый блок. В динамическом составном операторе вы можете объявлять переменные SQL, объявлять условия, относящиеся к соответствующим SQLSTATE; он может содержать один или несколько процедурных операторов SQL. Если в динамическом составном операторе происходит ошибка, выполняется откат всех предшествующих операторов SQL, а оставшиеся операторы SQL в динамическом составном операторе не обрабатываются.

Динамический составной оператор можно встроить в триггер, функцию SQL или выполнить как динамический оператор SQL. Этот выполняемый оператор может быть подготовлен динамически. Для вызова этого оператора не требуется никаких привилегий, но относящийся к нему ID авторизации должен иметь необходимые привилегии для вызова встроженных в этот составной оператор операторов SQL.

Переменные вводятся в подоператорах в объявлении переменной. Условия вводятся в подоператорах на основе значений SQLSTATE объявления условия. Динамические составные операторы компилируются DB2 как простые операторы. Такой оператор может эффективно использоваться для коротких сценариев с простыми алгоритмами управления потоками, но значительным объемом потоков данных. Для более сложных конструкций со встроженным потоком управления лучше подходят процедуры SQL.

В динамическом составном операторе можно использовать некоторые операторы управления потоком. К ним относятся: оператор FOR, оператор IF, оператор ITERATE и оператор WHILE. Подробные сведения об этих операторах смотрите в справочнике *SQL Reference*.

Глава 4. Особенности среды

После раздела "Использование советчика по индексам" добавьте следующий раздел:

Использование индексных ключей большего размера

Есть возможность разрешить задавать в качестве части ключа индекса столбцы длиной более 255 байт. Переменная реестра DB2_INDEX_2BYTEVARLEN допускает использование для хранения длины ключа индекса 2 байта вместо 1.

Изменения переменной реестра влияют на несколько операторов SQL. Это:

- CREATE TABLE. Части первичного, внешнего ключа и ключа уникальности могут иметь размер более 255 байтов.

- CREATE INDEX. Размер, превышающий 255 байтов, могут иметь все индексы, включая индексы уникальности и включая столбцы с переменными частями ключей.
- ALTER TABLE. Части первичного, внешнего ключа и ключа уникальности могут иметь размер более 255 байтов. Размер больше 255 байт могут иметь все индексы, включая индексы уникальности и включая столбцы с переменными частями ключей.

Для внешнего ключа ограничение 255 байтов снимается независимо от значения переменной реестра. Соответствие первичного ключа внешнему ключу снимает все ограничения и пределы.

Чтобы пользоваться ключами индексов больших размеров, преобразуйте существующие индексы: отбросьте индексы, задайте для переменной реестра DB2_INDEX_2BYTEVARLEN значение ON, и создайте индексы вновь (со столбцами больших размеров).

Дополнительную информацию об операторах SQL и описание их синтаксиса смотрите в справочнике *SQL Reference*.

Глава 5. Статистики системного каталога

Сбор и использование статистики распределения

В подразделе “Пример влияния на предикаты равенства” обсуждается предикат $C \leq 10$. Указано, что ошибка равна -86%. Это неверно. Предложение в конце этого абзаца должно выглядеть так:

Если исходить из предположения о равномерном распределении данных и использовать формулу (1), с

В подразделе “Пример влияния на предикаты равенства” обсуждается предикат $C > 8,5 \text{ AND } C \leq 10$. Оценка значения r_2 с использованием линейной интерполяции должна быть изменена на:

$$r_2 * = \frac{10 - 8,5}{100 - 8,5} \times (\text{число строк, значение в которых } > 8,5 \text{ и } \leq 100,0)$$

$$r_2 * = \frac{10 - 8,5}{100 - 8,5} \times (10 - 7)$$

$$r_2 * = \frac{1,5}{91,5} \times (3)$$

$$r_2 * = 0$$

Следующий за этим новым примером абзац также нужно изменить:

Итоговая оценка - $r_1 + r_2 * = 7$, и абсолютная ошибка равна только -12,5%.

Правила обновления статистики каталога

В разделе “Правила обновления статистики столбцов” последний абзац в первом пункте списка надо заменить следующим:

HIGH2KEY должно быть больше, чем LOW2KEY, если в соответствующем столбце есть более 3 различных значений. Если число различных значений в столбце равно или меньше 3, HIGH2KEY может равняться LOW2KEY.

Статистика подэлементов

В FixPak 1 есть опция для сбора и использования статистики подэлементов. Это статистика о содержимом данных в столбцах, когда структура данных имеет вид последовательностей подполей и подэлементов, разделенных пробелами.

Например, предположим, что база данных содержит таблицу DOCUMENTS, в которой каждая строка описывает документ, и в DOCUMENTS есть столбец под названием KEYWORDS, содержащий список соответствующих ключевых слов, описывающих текст этого документа. Значения в KEYWORDS могут выглядеть так:

```
'база моделирование аналитика бизнес интеллект'  
'моделирование модель дрозофила размножение температура'  
'лес хвойный почва эрозия дождевой'  
'лес температура почва осадки пожар'
```

В этом примере каждое значение столбца состоит из 5 подэлементов, каждый из которых является словом (ключевым словом), отделенным от других одним пробелом.

Для запросов, содержащих предикаты LIKE для таких столбцов с использованием символа %, который означает соответствие всем символам:

```
SELECT .... FROM DOCUMENTS WHERE KEYWORDS LIKE '%моделирование%'
```

оптимизатору часто полезно знать основные статистические характеристики структуры подэлементов этого столбца, а именно:

SUB_COUNT

Среднее число подэлементов.

SUB_DELIM_LENGTH

Средняя длина ограничителя, отделяющего подэлемент, где ограничитель в данном контексте - один или несколько последовательных символов пробела.

В приведенном примере SUB_COUNT равняется 5, а SUB_DELIM_LENGTH - 1, поскольку все ограничители состоят из одного символа пробела.

В FixPak 1 системный администратор управляет сбором и использованием такой статистики при помощи расширения переменной реестра DB2_LIKE_VARCHAR. Эта переменная реестра определяет, как оптимизатор DB2 UDB работает с предикатами вида:

```
COLUMN LIKE '%xxxxxx'
```

где xxxxxx - любая строка символов; то есть с любыми предикатами LIKE, поисковое значение которых начинается с символа %. (Оно может и заканчиваться символом %, но это не обязательно). Далее мы будем называть их "предикаты LIKE с подстановкой". Для каждого предиката оптимизатор должен оценить, какое число строк отвечает этому предикату. Для предикатов LIKE с подстановкой оптимизатор полагает, что у COLUMN, для которого найдено совпадение, имеется структура из последовательности сочлененных элементов, образующих собственно столбец, и оценивает длину каждого из этих элементов на основе длины строки символов, исключая из нее начальные и конечные символы %. Новый синтаксис:

```
db2set DB2_LIKE_VARCHAR=[Y|N|S|num1][,Y|N|num2]
```

где

- первый член (перед запятой) означает следующее (только для столбцов, не имеющих позитивной статистики подэлементов)

S

Использовать алгоритм, как в DB2 версии 2.

N	Использовать алгоритм подэлементов с фиксированной длиной.
Y (по умолчанию)	Использовать алгоритм подэлементов с переменной длиной с используемым по умолчанию значением для параметра алгоритма.
num1	Использовать алгоритм подэлементов с переменной длиной со значением num1 для параметра алгоритма.
- второй член (после запятой) означает:	
N (по умолчанию)	Не собирать или не использовать статистики подэлементов.
Y	Собирать статистики подэлементов. Использовать алгоритм подэлементов с переменной длиной, использующий эти статистики, а также устанавливаемое по умолчанию значение для параметра алгоритма в случае столбцов с позитивными статистиками подэлементов.
num2	Собирать статистики подэлементов. Использовать алгоритм подэлементов с переменной длиной, использующий эти статистики, а также значение num2 для параметра алгоритма в случае столбцов с позитивными статистиками подэлементов.

Если значение DB2_LIKE_VARCHAR содержит только первый член, статистика для подэлементов не собирается, а вся ранее собранная статистика игнорируется. Указанное значение влияет подсчет оптимизатором селективности предикатов LIKE с подстановкой так же, как и раньше, то есть:

- Если указано значение S, оптимизатор использует такой же алгоритм, как использовался в DB2 версии 2, в которой не предполагалось подэлементной модели.
- Если указано значение N, оптимизатор использует алгоритм, предполагающий подэлементную модель, и считает, что COLUMN имеет фиксированную длину даже в том случае, когда он определен как имеющий переменную длину.
- Если указано значение Y (по умолчанию) или константа с плавающей запятой, оптимизатор использует алгоритм, предполагающий подэлементную модель, и распознает, что COLUMN имеет переменную длину, если он так определен. Это также подразумевает получение статистики подэлементов непосредственно из запроса, а не из данных. Этот алгоритм содержит параметр ("параметр алгоритма"), указывающий, насколько элемент длиннее, чем строка символов, заключенная в символы %.
- Если указано Y, для параметра алгоритма оптимизатор по умолчанию использует значение 1,9.
- Если указана константа с плавающей запятой, для параметра алгоритма оптимизатор использует это значение. Константа должна быть в диапазоне от 0 до 6,2.

Если значение DB2_LIKE_VARCHAR содержит два члена и второй из них - Y или константа с плавающей запятой, статистика подэлементов собирается во время операции RUNSTATS по столбцам типа CHAR, VARCHAR, GRAPHIC или VARGRAPHIC со строками наборов однобайтных символов и используются во время компиляции запросов, включающих в себя предикаты LIKE с подстановкой. Оптимизатор использует алгоритм, предполагающий подэлементную модель, и вычисляет селективность предиката по показателям SUB_COUNT и SUB_DELIM_LENGTH, а также по параметру алгоритма. Параметр алгоритма указывается так же, как описано выше, то есть:

- Если указано Y, для параметра алгоритма оптимизатор по умолчанию использует значение 1,9.
- Если указана константа с плавающей запятой, для параметра алгоритма оптимизатор использует это значение. Константа должна быть в диапазоне от 0 до 6,2.

Если во время компиляции оптимизатор обнаружит, что статистики подэлементов для включенного в запрос столбца не собраны, он будет использовать "дедуктивный" алгоритм подэлементов, то есть алгоритм, используемый, если указан только первый член DB2_LIKE_VARCHAR. Таким образом, чтобы статистики подэлементов были использованы оптимизатором, второй член DB2_LIKE_VARCHAR должен быть установлен и во время RUNSTATS, и во время компиляции.

Значения статистик подэлементов можно посмотреть, запросив SYSIBM.SYSCOLUMNS. Например:

```
select substr(NAME,1,16), SUB_COUNT, SUB_DELIM_LENGTH
from sysibm.syscolumns where tname = 'DOCUMENTS'
```

Столбцов SUB_COUNT и SUB_DELIM_LENGTH нет в производной таблице статистики SYSSTAT.COLUMNS и поэтому изменить их нельзя.

Примечание: При использовании этой опции RUNSTATS может работать дольше. Например, RUNSTATS может работать на 15 - 40% дольше для таблицы со столбцами из пяти символов, если не используются опции DETAILED и DISTRIBUTION. Если указана опция DETAILED или DISTRIBUTION, процентное увеличение будет меньше, несмотря на то, что абсолютное значение увеличения останется таким же. Если вы собираетесь использовать эту опцию, надо сопоставить эти дополнительные расходы и повышение производительности запроса.

Глава 6. Компилятор SQL

Нужно изменить следующие разделы:

Реплицируемые сводные таблицы

Этот раздел должен содержать следующую информацию:

Реплицируемые сводные таблицы могут использоваться для обеспечения локального выполнения объединений. Например, если есть схема типа "звезда", в которой большая таблица фактов распределена по двенадцати узлам, объединения между таблицей фактов и таблицами ассоциаций будут более эффективны, если эти таблицы размещены совместно.

Если все эти таблицы размещены в одной группе узлов, для локально выполняемого объединения корректно может быть распределена только одна таблица ассоциации. Все другие таблицы ассоциаций нельзя будет использовать в локально выполняемом объединении, так как столбцы объединения таблицы фактов не будут соответствовать ключу разделения таблицы фактов.

Например, вы можете иметь таблицу фактов FACT (C1, C2, C3, ...), для разделения которой используется столбец C1, таблицу ассоциации DIM1 (C1, dim1a, dim1b, ...), для разделения которой используется столбец C1, таблицу ассоциации DIM2 (C2, dim2a, dim2b, ...), для разделения которой используется столбец C2, и т.д.

В этом примере можно увидеть, что объединение между таблицами FACT и DIM1 будет выполняться лучше всего, так как для предиката DIM1.C1 = FACT.C1 будет выполняться условие совместного размещения. Обе эти таблицы используют для разделения столбец C1.

Для объединения с DIM2 с предикатом WHERE DIM2.C2 = FACT.C2 не может использоваться локально выполняемое объединение, так как для разделения таблицы FACT используется столбец C1, а не C2.

В этом случае хорошим решением может быть репликация таблицы DIM2 в группу узлов таблицы фактов. При этом объединение может быть выполнено локально на каждом разделе.

Примечание: Реплицируемые сводные таблицы используются для репликации внутри одной базы данных. При репликации между базами данных используются регистрации и управляющие таблицы, а данные располагаются в различных базах данных и в различных операционных системах. Дополнительную информацию о репликации между базами данных смотрите в руководстве *Replication Guide and Reference*.

При создании реплицируемой сводной таблицы исходная таблица может располагаться в одноузловой группе узлов или в многоузловой группе узлов. В большинстве случаев эта таблица имеет небольшой размер и ее можно поместить в одноузловую группу узлов. Объем реплицируемых данных можно ограничить, задав только часть столбцов этой таблицы, ограничив число строк при помощи предикатов или используя при создании реплицируемой сводной таблицы оба эти метода.

Примечание: Для работы реплицируемой сводной таблицы не требуется опция захвата данных.

Реплицируемую сводную таблицу можно также создать в многоузловой группе узлов. Это та же группа узлов, в которой размещены таблицы большого объема. В этом случае копии исходной таблицы создаются на всех разделах этой группы узлов. В такой среде объединения между большой таблицей фактов и таблицами ассоциаций будут с большей вероятностью выполняться локально, и не будет выполняться передача данных исходной таблицы на все разделы.

Индексы для реплицируемых таблиц не создаются автоматически. Созданные индексы могут отличаться от индексов исходной таблицы.

Примечание: Для реплицируемых таблиц нельзя создать индексы уникальности (или задать какие-либо ограничения). Этим предотвращаются нарушения ограничений, которые не определены для исходных таблиц. Эти ограничения не разрешены, даже если такие же ограничения определены для исходной таблицы.

После использования оператора REFRESH следует выполнить утилиту RUNSTATS для реплицируемой таблицы, так же как для любой другой таблицы.

Реплицируемые таблицы можно использовать прямо в запросе. Однако нельзя использовать с реплицируемой таблицей предикат NODENUMBER(), чтобы узнать данные таблицы на конкретном разделе.

Чтобы узнать, используется ли созданная реплицируемая сводная таблица для запроса, в котором указана исходная таблица, можно использовать средство EXPLAIN. Сначала убедитесь, что существуют таблицы EXPLAIN. Затем создайте план объяснения для интересующего вас оператора SELECT. Наконец, используйте утилиту db2exfmt для форматирования выходных данных EXPLAIN.

В зависимости от того, какие данные будут использоваться в объединении, выбранный оптимизатором план доступа может использовать или не использовать

реплицируемую сводную таблицу. Если оптимизатор определит, что выгоднее передать исходную таблицу на другие разделы этой группы узлов, реплицируемая сводная таблица не будет использоваться.

Концепции доступа к данным и оптимизация

Изменен раздел “Обращение к нескольким индексам” в разделе “Принципы просмотра индексов”.

Перед примечанием в конце этого раздела добавьте следующую информацию:

Чтобы реализовать возможности улучшения производительности от использования динамических битовых образов при просмотре нескольких индексов, может понадобиться изменение значений параметра конфигурации базы данных *sorthheap* (размер кучи сортировки) и параметра конфигурации менеджера баз данных *sheapthres* (порог кучи сортировки).

Если в плане доступа используются динамические битовые образы, требуется дополнительный объем кучи сортировки. Если заданное значение *sheapthres* относительно близко к значению *sorthheap* (то есть для одновременных запросов они различаются менее чем в два или три раза), для работы с динамическими битовыми образами будет использоваться гораздо меньше памяти, чем предполагал оптимизатор.

Чтобы исправить эту ситуацию, увеличьте значение *sheapthres* по отношению к значению *sorthheap*.

Изменен раздел “Стратегии поиска для объединения типа “звезда”” в разделе “Терминология предикатов”.

В конце этого раздела добавьте следующую информацию:

Для динамических битовых образов, создаваемых и используемых в технологии объединения типа “звезда”, используется память кучи сортировки. Дополнительную информацию о параметре конфигурации базы данных *sorthheap* (размер кучи сортировки) смотрите в главе 13, “Конфигурирование DB2” книги *Administration Guide: Performance*.

Глава 8. Производительность работы

Управление пулом буферов базы данных

В разделе “Управление пулом буферов базы данных” после абзаца, начинающегося фразой “При создании пула буферов размер страницы по умолчанию составляет 4 Кбайта”, добавьте следующую информацию:

При работе с Windows 2000 поддерживаются размеры пула буферов до 64 Гбайт минус размер DB2 и размер операционной системы. (Предполагается, что DB2 - основная программа в системе.) Такая поддержка доступна благодаря Microsoft Address Windowing Extensions (AWE).

AWE может использоваться с пулами буферов любого размера, но если вам нужно использовать AWE на пулах буферов большего размера, существуют другие рекомендуемые продукты Windows. Windows 2000 Advanced Server обеспечивает поддержку до 8 Гбайт памяти. Сервер Windows 2000 Data Center обеспечивает поддержку до 64 Гбайт памяти.

Для поддержки пулов буферов AWE нужно правильно сконфигурировать DB2 и Windows 2000. Пул буферов, который будет использовать преимущества AWE, должен существовать в базе данных.

Для выделения 3 Гбайт пользовательского пространства используйте опцию загрузки /3GB Windows 2000. Она допускает использование окна AWE больших размеров. Для доступа к объему памяти более 4 Гбайт через интерфейс памяти AWE используйте опцию загрузки /PAE Windows 2000. Чтобы убедиться, что опция загрузки выбрана правильно, выберите Управление -> Система а затем “Запуск и восстановление”. В выпадающем списке можно увидеть доступные опции загрузки. Если нужная вам опция загрузки (/3GB или /PAE) выбрана, можно приступить к следующей задаче по установке поддержки AWE. Если желаемая вами опция загрузки недоступна для выбора, надо добавить эту опцию в файл boot.ini на системном диске. Файл boot.ini содержит список действий, которые нужно выполнить при запуске операционной системы. Добавьте /3GB или /PAE или обе опции (через пробел) в конце списка существующих параметров. Сохранив измененный файл, можно проверить и выбрать правильную опцию загрузки, как сказано выше.

Кроме того, в Windows 2000 следует задать для пользователя, под именем которого установлена DB2, опцию “lock pages in memory”-right. Для этого надо зарегистрироваться в Windows 2000 как пользователь, установивший DB2, в меню Start (Пуск) в Windows 2000 выбрать папку “Administrative Tools” (Инструменты управления), а затем программу “Local Security Policy” (Локальная политика защиты). В этой программе можно выбрать назначение прав пользователей, установив “lock pages in memory”-right.

Для DB2 требуется задать переменную реестра DB2_AWE. Чтобы правильно задать эту переменную реестра, нужно знать ID пула буферов, для которого вы хотите разрешить поддержку AWE. Кроме того, надо знать выделяемое число физических страниц и число страниц окна адресации. Число выделяемых физических страниц должно быть несколько меньше общего числа физических страниц. Фактическое выбранное число будет зависеть от рабочей среды. Например, для среды, где в системе используются только DB2 и программы баз данных, в качестве значения, используемого с переменной DB2_AWE, можно выбрать объем на 0,5 - 1 Гбайт меньше общего объема физических страниц. В случае среды, где системой используются программы, не относящиеся к базе данных, следует увеличить значение, вычитаемое из общего объема, предоставив большее количество физических страниц для этих других программ. Число, используемое в переменной реестра DB2_AWE - это число физических страниц, которые будут использоваться для поддержки AWE и DB2. Верхний предел для страниц окна адресации - 1 Гбайт.

Информацию о задании переменной реестра DB2 DB2_AWE смотрите ниже в этом разделе в “Приложении А. Переменные среды и реестра DB2”, в таблице новых и измененных переменных реестра.

Управление несколькими пулами буферов базы данных

В раздел “Управление несколькими пулами буферов базы данных” после абзаца, начинающегося с фразы “Разрабатывая базу данных, вы могли предпочесть таблицы с размером страниц 8 Кбайт”, добавьте следующий абзац:

При работе с Windows 2000 переменную реестра DB2_AWE можно использоваться для переопределения параметров пула буферов в файлах каталога и конфигурации. Использование этой переменной реестра позволяет увеличить размеры пула буферов примерно до 64 Гбайт.

В этом же разделе замените абзац непосредственно перед этим примечанием на следующий:

Запустить менеджер баз данных с минимальными значениями размеров можно для того, чтобы соединиться с базой данных. После этого можно изменить конфигурацию размеров пула буферов или выполнить другие срочные задачи, чтобы перезапустить базу данных с правильными размерами пула буферов. Не стоит длительное время использовать базу данных в таком состоянии.

В разделе “Реорганизация каталогов и таблиц пользователя” последнее предложение (с коротким списком) абзаца, начинающегося с фразы “Утилита REORG позволяет задать временное табличное пространство...”, можно заменить следующим:

Использовать то же табличное пространство быстрее, однако это приводит к большему объему записи в журнал и требует достаточного места для реорганизованной таблицы. Если вы задаете временное табличное пространство, обычно рекомендуется задать временное табличное пространство SMS. Временное табличное пространство DMS не рекомендуется, поскольку при использовании этого типа табличного пространства может выполняться только программа REORG.

В раздел “Расширение памяти” добавьте после третьего абзаца следующий абзац:

При выделении пула буферов Windows 2000 Address Windowing Extensions (AWE) с использованием переменной реестра DB2_AWE кэш расширенной памяти использоваться не может.

Глава 9. Использование утилиты ограничения ресурсов

В разделе “Создание файла конфигурации распорядителя” первое предложение в первом абзаце после обсуждения действия планирования следует заменить следующим:

Если к программе применимы несколько правил, применяются все правила. В зависимости от правила и устанавливаемых пределов, первым применяется действие, связанное с первым встречным пределом.

Глава 13. Конфигурирование DB2

Нужно изменить описания следующих параметров:

Размер кучи сортировки (sortheap)

Изменен раздел “Рекомендации”. Он должен содержать следующую информацию:

При работе с кучей сортировки нужно учитывать следующее:

- Выбор подходящих индексов может минимизировать использование кучи сортировки.
- Буферы хеш-объединений и динамические битовые образы (используемые для операций AND для индексов и для объединений типа “звезда”) используют память кучи сортировки. Если используются эти техники, увеличьте размер этого параметра.
- Увеличьте размер этого параметра, если часто требуются сортировки большого объема данных.
- ... (остальной текст остался без изменений)

Порог кучи сортировки (sheapthres)

Изменен второй с конца абзац в описании этого параметра. Этот абзац должен выглядеть так:

Примеры операций, использующих кучу сортировки: сортировки, динамические битовые образы (используемые для операций AND для индексов и для объединений типа "звезда") и операции, в которых таблицы находятся в памяти.

К описанию этого параметра надо добавить следующую информацию:

Не нужно увеличивать значение этого параметра при переходе из одноузловой среды в многоузловую. Если параметры конфигурации базы данных и менеджера баз данных настроены в одноузловой среде (в DB2 EE), в большинстве случаев эти же параметры дадут хорошие результаты в многоузловой среде (в DB2 EEE).

Поскольку параметр порога кучи сортировки является параметром конфигурации менеджера баз данных, он применяется для всего экземпляра DB2. Единственный способ задать для этого параметра различные значения на разных узлах или разделах - создать несколько экземпляров DB2. При этом на разных группах узлов должны быть разные базы данных DB2. Такое решение делает бессмысленными многие преимущества среды многораздельных баз данных.

Максимальный процент списка блокировок перед расширением (maxlocks)

Следующее изменение надо внести в раздел Рекомендация описания параметра конфигурации базы данных "Максимальный процент списка блокировок перед расширением (maxlocks)".

Рекомендация: Следующая формула позволяет настроить параметр *maxlocks*, чтобы разрешить программе удерживать число блокировок в два раза больше среднего:

$$\text{maxlocks} = 2 * 100 / \text{maxapps}$$

Где 2 увеличивает среднее значение в два раза, а 100 - наибольший допустимый процент. Если параллельно работают всего несколько программ, вместо первой формулы можно использовать такую:

$$\text{maxlocks} = 2 * 100 / (\text{среднее число программ, работающих параллельно})$$

Одна из особенностей настройки параметра *maxlocks*: его следует конфигурировать совместно с размером списка блокировок (*locklist*). Фактическое предельное число блокировок, удерживаемых программой перед расширением блокировок:

$$\text{maxlocks} * \text{locklist} * 4096 / (100 * 36)$$

Где 4096 - число байтов на странице, 100 - наибольший процент, допустимый для *maxlocks* и 36 - число байтов на блокировку. Если вы знаете, что одна из программ требует 1000 блокировок, и не хотите, чтобы произошло расширение блокировок, выберите в этой формуле значения для *maxlocks* и *locklist* так, чтобы результат был больше 1000. (Задав в этой формуле для *maxlocks* 10, а для *locklist* 100, в результате вы получите больше 1000 необходимых блокировок.)

Если для *maxlocks* установить слишком низкое значение, расширение блокировок произойдет, когда для других параллельных программ все еще будет оставаться достаточно пространства блокировок. Если для *maxlocks* установить слишком высокое значение, большую часть пространства блокировок могут задействовать

несколько программ, и расширение блокировок придется выполнять другим программам. Такая потребность в расширении блокировок приводит в этом случае к плохому параллелизму.

Для отслеживания и настройки этого параметра конфигурации можно использовать системный монитор баз данных.

Конфигурирование для DB2/Менеджера связей данных DB2 интервала истечения срока маркера доступа к связям данных `dl_expirint`

Вопреки документации, если для `dl_expirint` устанавливается значение "-1", срок маркера управления доступом истекает. Обходной прием - задать для `dl_expirint` максимальное значение 31536000 (секунд). Это соответствует сроку истечения 1 год, чего должно хватить для всех программ.

Параметр конфигурации базы данных `MIN_DEC_DIV_3`

Добавление параметра конфигурации базы данных `MIN_DEC_DIV_3` дает быстрый способ изменить вычисление масштаба десятичного деления в SQL. Для `MIN_DEC_DIV_3` можно задать значения YES или NO. Значение по умолчанию для `MIN_DEC_DIV_3` - NO.

Параметр конфигурации базы данных `MIN_DEC_DIV_3` изменяет масштаб результата десятичной арифметической операции, где используется деление. Если задано значение NO, масштаб вычисляется как $31-p+s-s'$. Дополнительную информацию смотрите в справочнике *SQL Reference*, Глава 3, "Decimal Arithmetic in SQL". Если задано значение YES, масштаб вычисляется как $MAX(3, 31-p+s-s')$. Это приводит к тому, что результат десятичного деления всегда имеет масштаб как минимум 3. Точность всегда равна 31.

Изменение этого параметра конфигурации базы данных может привести к изменениям в программах для существующих баз данных. Это может произойти, если изменение этого параметра конфигурации базы данных повлияет на полученный при десятичном делении масштаб. Ниже приведены некоторые возможные сценарии, которые могут влиять на программы. Эти сценарии следует учесть перед изменением параметра `MIN_DEC_DIV_3` на сервере баз данных с действующими базами данных.

- При изменении масштаба результата одного из столбцов производной таблицы, которая определяется в среде с такой конфигурацией, при обращении к этой производной таблице после изменения данного параметра конфигурации базы данных может случиться ошибка с SQLCODE -344. Сообщение SQL0344N относится к рекурсивным общим табличным выражениям, однако если именем объекта (первый маркер) является производная таблица, чтобы избежать этой ошибки, вам необходимо отбросить эту производную таблицу и создать ее заново.
- Поведение статического пакета не изменится, пока не свяжут его повторно, неявно или явно. Например, после изменения значения с NO на YES дополнительные разряды масштаба не будут включены в результаты, пока не будет выполнено повторное связывание. Чтобы провести повторное связывание принудительно для любых измененных статических пакетов, можно воспользоваться явной командой повторного связывания.
- Проверочное ограничение, в котором используется десятичное деление, может отклонить некоторые значения, допустимые ранее. Такие строки будут теперь нарушать ограничение, но они не будут обнаружены, пока не изменится один из столбцов, участвующих в строке проверочного ограничения, или не будет выполнена команда **SET INTEGRITY** с опцией **IMMEDIATE CHECKED**. Для принудительной проверки такого ограничения выполните команду **ALTER**

TABLE, чтобы отбросить проверочное ограничение, а затем снова добавьте это ограничение, выполнив команду **ALTER TABLE** еще раз.

Примечание: В DB2 Версии 7 действуют также следующие ограничения:

1. Параметр **MIN_DEC_DIV_3** не выводится командой **GET DB CFG FOR DBNAME**. Лучший способ узнать текущее значение - посмотреть побочный эффект результата десятичного деления. Например, рассмотрим следующий оператор:
`VALUES (DEC(1,31,0)/DEC(1,31,5))`

Если этот оператор возвращает sqlcode SQL0419N, база данных не поддерживает параметр **MIN_DEC_DIV_3**, или для него установлено значение OFF. Если данный оператор возвращает 1,000, для параметра **MIN_DEC_DIV_3** установлено значение ON.

2. Если ввести приведенную ниже команду, параметр **MIN_DEC_DIV_3** в списке ключевых слов конфигурации не выводится: **? UPDATE DB CFG**

Размер кучи управления программами (**app_ctl_heap_sz**)

Текст для этого параметра теперь следует читать так:

Для многораздельных и одnorаздельных баз данных с включенным внутренним параллелизмом (**intra_parallel=ON**) это размер совместно используемой области памяти, выделенной для кучи управления программами. Для одnorаздельных баз данных, где внутренний параллелизм отключен (**intra_parallel=OFF**), это максимальная собственная память, которая будет выделена для кучи. На каждое соединение раздела существует одна куча управления программами.

Куча управления программами требуется, прежде всего, для совместного использования информации между агентами, работающими для одного запроса, а в среде многораздельных баз данных - для хранения выполняемых разделов, представляющих операторы SQL. Использование этой кучи минимально в одnorаздельных базах данных, где выполняются запросы со степенью параллелизма меньше либо равной 1.

Эта куча служит также для хранения информации дескриптора для объявленных временных таблиц. Информация дескриптора для всех объявленных временных таблиц, которые не были явным образом отброшены, сохраняется в памяти этой кучи и не может быть отброшена, пока не отброшена объявленная временная таблица.

Часть "Рекомендация" осталась без изменений.

Размер кучи системного монитора базы данных (**mon_heap_sz**)

Значение по умолчанию для сервера баз данных OS/2 и Windows NT с локальными и удаленными клиентами и для сервера баз данных Satellite с локальными клиентами изменилось с 24 на 32. Диапазон значений остался прежним.

Максимальное число активных прикладных программ (**maxappls**)

Верхний предел диапазона для всех платформ изменился с 64000 на 60000. Значение по умолчанию осталось прежним.

Диапазон восстановления и интервал мягких контрольных точек (softmax)

Единицей измерения теперь служат проценты от размера одного первичного файла журнала.

Разрешить отслеживание измененных страниц (trackmod)

Тип конфигурации: База данных

Тип параметра: Конфигурируемый

Значение по умолчанию [Диапазон]: Off [On; Off]

Если для этого параметра установить значение ON, менеджер баз данных будет отслеживать страницы в базе данных, измененные со времени создания самой последней полной резервной копии. Это позволяет утилите резервного копирования определить, какие страницы следует включить в инкрементную резервную копию, не проверяя каждую страницу отдельно. Для табличных пространств SMS уровень такого отслеживания - табличное пространство. Для табличных пространств DMS уровень отслеживания - экстенд для страниц данных и страниц индексов, и табличное пространство - для других типов страниц. После установки для этого параметра значения ON нужно создать резервную копию базы данных, чтобы иметь основу для инкрементных резервных копий.

Изменить путь журнала базы данных (newlogpath)

Тип конфигурации: База данных

Тип параметра: Конфигурируемый

Значение по умолчанию [Диапазон]: Null [любой допустимый путь или устройство]

Параметры, связанные с данным: Положение файлов журнала (logpath);
Согласованность базы данных (database_consistent)

В этом параметре можно задать строку длиной до 242 байтов, чтобы изменить место хранения файлов журнала. Строка может задавать как путь, так и непосредственное устройство. Если в строке задается путь, он должен быть полным, а не относительным.

Примечание: В среде многораздельных баз данных к пути автоматически добавляется номер узла. Это делается для сохранения единого пути в нескольких конфигурациях логических узлов.

Чтобы задать устройство, введите строку, которую операционная система распознает в качестве устройства. Например, в Windows NT -

\\.\d: или \\.\PhysicalDisk5

Примечание: Чтобы была возможность записи журналов на устройство, надо использовать Windows NT Версии 4.0 с Service Pack 3.

Для платформ на основе UNIX -
/dev/rdblog8

Примечание: Устройство можно задать только на платформах AIX, Windows 2000, Windows NT, Solaris, HP-UX, NUMA-Q и Linux.

Новая установка не станет значением для logpath, пока не выполнены два условия:

- База данных должна находиться в согласованном состоянии, что указывается параметром database_consistent.
- Все пользователи должны отсоединиться от базы данных.

После установки с базой данных первого нового соединения менеджер баз данных переместит журналы в новое место, заданное параметром logpath.

На прежнем месте могут остаться файлы журнала. Эти файлы могут быть еще не заархивированы. Их может понадобиться архивировать вручную. Кроме того, если для этой базы данных вы запускаете репликацию, для нее могут понадобиться файлы журнала, записанные до изменения пути. Если при конфигурировании базы данных для ее параметра конфигурации Включить обработчик пользователя (userexit) было установлено значение "Yes", а все файлы журнала архивировались либо автоматически DB2, либо вами вручную, у DB2 есть возможность получить файлы журнала для процесса репликации. В противном случае можно скопировать эти файлы из старого пути в новый.

Рекомендация: Желательно поместить файлы журнала на физический диск, с которым не идут интенсивные операции ввода-вывода. Например, не помещайте журналы на диск, где расположена операционная система или базы данных большого объема. Это позволит эффективно вести запись при минимальных расходах, таких как ожидание ввода-вывода.

Чтобы отследить число операций ввода-вывода, относящихся к журналам базы данных, можно воспользоваться системным монитором базы данных.

Дополнительную информацию смотрите в справочном руководстве System Monitor Guide and Reference, в описаниях для следующих элементов монитора:

- log_reads (число прочитанных страниц журнала)
- log_writes (число записанных страниц журнала)

Эти элементы данных возвращают показатели активности ввода-вывода, связанной с регистрацией в журналах базы данных. Можно воспользоваться средствами монитора операционной системы для сбора информации о других дисковых операциях ввода-вывода, а затем сравнить результаты, полученные для этих видов операций.

Положение файлов журналов (logpath)

Тип конфигурации: База данных

Тип параметра: Информационный

Параметры, связанные с данным: Изменить путь журнала для базы данных (newlogpath)

Этот параметр содержит текущий путь для журналов регистрации. Этот параметр нельзя изменить непосредственно, так как он задается менеджером баз данных после того, как вступает в действие изменение для параметра newlogpath. После создания базы данных для нее создается файл журнала восстановления в подкаталоге каталога, содержащего эту базу данных. По умолчанию используется подкаталог SQLOGDIR в каталоге, созданном для этой базы данных.

Максимальная память для списка блокировок (locklist)

Максимальное значение увеличено с 60000 до 524288.

Приложение А. Переменные реестра DB2 и переменные среды

Следующие переменные реестра являются новыми или требуют изменения:

Таблица новых и измененных переменных реестра

Таблица 3. Переменные реестра

Имя переменной	Операционная система	Значения
Описание		
DB2MAXFSCRSEARCH	Все	По умолчанию=5 Значения: -1, от 1 до 33554
Задаёт число записей управления свободным пространством, которые ищутся при добавлении записи в таблицу. Значение по умолчанию - искать пять записей управления свободным пространством. Изменяя это значение, можно улучшить скорость операций вставки за счёт повторного использования пространства или наоборот. Используйте большие значения, чтобы улучшить повторное использование пространства. Используйте небольшие значения, чтобы улучшить скорость операций вставки. Если задано значение -1, менеджер баз данных будет искать все записи управления свободным пространством.		
DLFM_TSM_MGMTCLASS	AIX, Windows NT, Solaris	По умолчанию: класс управления TSM по умолчанию Значения: любой допустимый класс управления TSM
Задаёт класс управления TSM, который должен использоваться для архивирования и получения связанных файлов. Если значение этой переменной не задано, используется класс управления TSM по умолчанию.		
DB2_CORRELATED_PREDICATES	Все	По умолчанию: YES Значения: YES или NO
Значение по умолчанию для этой переменной - YES. Если по соответствующим столбцам в объединении существуют индексы уникальности, и для этой переменной реестра задано значение YES, оптимизатор пытается выявить и компенсировать корреляцию предикатов объединения. Если для этой переменной реестра задано значение YES, оптимизатор использует информацию KEYCARD статистики индексов уникальности, чтобы выявить случаи корреляции, и динамически настраивает совместную избирательность коррелирующих предикатов, таким образом получая более точную оценку размера и стоимости объединения.		
DB2_VI_DEVICE	Windows NT	По умолчанию=null Значения: nic0 или VINIC
Задаёт символическое имя устройства или Virtual Interface Provider Instance, связанного с платой сетевого интерфейса Network Interface Card (NIC). Все независимые производители оборудования (Independent hardware vendors, IHV) производят собственные NIC. На одном компьютере Windows NT можно установить только одну (1) NIC; несколько логических узлов на одном физическом компьютере будут совместно использовать одну и ту же NIC. Символическое имя устройства "VINIC" должно задаваться в верхнем регистре; его можно использовать только для соединений Synfinity. Для всех других поддерживаемых в настоящее время реализаций используйте в качестве символического имени устройства имя "nic0".		
DB2_SELECTIVITY	Все	По умолчанию=NO Значения: YES или NO
Эта переменная реестра управляет тем, где может использоваться условие SELECTIVITY. Подробности об условии SELECTIVITY смотрите в разделе Language Elements, Search Conditions справочника <i>SQL Reference</i> .		
Если эта переменная реестра имеет значение YES, условие SELECTIVITY можно указывать для базового предиката, по крайней мере одно из выражений которого содержит переменные хоста.		

Таблица 3. Переменные реестра (продолжение)

Имя переменной	Операционная система	Значения
<p>Описание</p> <p>DB2_UPDATE_PART_KEY</p> <p>Все</p> <p>По умолчанию: YES</p> <p>Значения: YES или NO</p> <p>Для пакета исправлений FixPak 3 и новее значение по умолчанию - YES. Эта переменная реестра задает, разрешено или нет изменение ключа разделения.</p>		
<p>DB2_BLOCK_ON_LOG_DISK_FULL</p> <p>Все</p> <p>По умолчанию=NO</p> <p>Значения: YES или NO</p> <p>С помощью этой переменной реестра DB2 можно предотвратить генерирование сообщений об ошибках заполнения диска ("disk full"), когда DB2 не может создать новый файл журнала по действующему пути журналов.</p> <p>Вместо этого DB2 будет пытаться создавать файл журнала каждые 5 минут, пока операция не завершится успешно. После каждой попытки DB2 записывает сообщение в файл db2diag.log. Единственный способ убедиться, что программа зависает из-за переполнения диска - это следить за файлом db2diag.log.</p> <p>Пока файл журнала не будет создан успешно, ни одна пользовательская программа, пытающаяся изменить табличные данные, не сможет осуществить принятие транзакций. На запросы только для чтения нельзя воздействовать непосредственно, однако если запросу нужен доступ к данным, которые блокируются требованием изменения или к странице данных которая исправляется в пуле буферов изменяющей программой, запросы только для чтения, скорее всего, также зависнут.</p>		
<p>DB2_INDEX_2BYTEVARLEN</p> <p>Все</p> <p>По умолчанию=NO</p> <p>Значения: YES или NO</p> <p>Эта переменная реестра позволяет задавать в качестве части ключа индекса столбцы длиной более 255 байт. У индексов, созданных до задания для этой переменной реестра значения YES, для ключей по-прежнему останется предельное ограничение 255. Индексы, созданные после задания для данной переменной реестра значения YES, будут вести себя как двухбайтные индексы, если для этой переменной реестра вновь задать значение NO.</p> <p>Изменения для этой переменной реестра влияют на несколько операторов SQL, в том числе CREATE TABLE, CREATE INDEX и ALTER TABLE. Дополнительную информацию об этих операторах смотрите в изменениях для справочника <i>SQL Reference</i>.</p>		
<p>DB2_FORCE_FCM_BP</p> <p>AIX</p> <p>По умолчанию=NO</p> <p>Значения: YES или NO</p> <p>Задаёт, откуда выделяются ресурсы менеджера FCM. Его ресурсы могут выделяться либо из сегмента совместно используемой памяти менеджера баз данных, либо из отдельного сегмента. Эту переменную реестра следует использовать, если на одном компьютере есть несколько логических узлов. В системе многораздельной базы данных с разрешенной симметричной многопроцессорной обработкой (SMP) значение этой переменной среды реестра не влияет на установление связи. В этом случае связь всегда осуществляется через совместно используемую память. Однако от этой переменной зависит число сегментов совместно используемой памяти, которое будет использовать DB2.</p>		
<p>DB2_AWE</p> <p>Windows 2000</p> <p>По умолчанию: Null</p> <p>Значения: <запись>; <запись>;..., где <запись>=<ID пула буферов>, <число физических страниц>, <число окон адресации></p>		

Таблица 3. Переменные реестра (продолжение)

Имя переменной	Операционная система	Значения
Описание		
<p>Позволяет DB2 UDB в Windows 2000 выделять пулы буферов, где используется до 64 Гбайт памяти. Для поддержки пулов буферов Address Windowing Extensions (AWE) нужно правильно сконфигурировать Windows 2000. Надо установить для данного пользователя в Windows 2000 “lock pages in memory”-right, и задать эту переменную реестра в DB2. При задании этой переменной нужно знать ID пула буферов, который будет использоваться для поддержки AWE. Кроме того, надо определить выделяемое число физических страниц и число окон адресации.</p> <p>Информацию об определении числа выделяемых физических страниц и числа окон адресации смотрите выше в данном документе: “Глава 8. Производительность работы”, раздел “Управление пулом буферов базы данных”.</p> <p>Примечание: Если поддержка AWE включена, расширенная память (ESTORE) не может использоваться ни для каких пулов буферов в базе данных. Пулы буферов, на которые действует эта переменная, должны уже существовать в SYSIBM.SYSBUFFERPOOLS.</p>		
DB2_STPROC_LOCKUP_FIRST	Все	По умолчанию=NO Значения: YES или NO
Эта переменная реестра раньше называлась DB2_DARI_LOOKUP_ALL.		
DB2MEMDISCLAIM	AIX	По умолчанию: YES Значения: YES или NO
<p>В AIX память, используемая процессами DB2, может иметь некоторое связанное пространство подкачки. Это пространство подкачки может оставаться зарезервированным, даже если была освобождена связанная с ним память. Резервирование пространства подкачки зависит от правил управления выделением настраиваемой виртуальной памяти системы AIX. Эта переменная реестра задает, будут ли агенты DB2 явно требовать, чтобы AIX отсоединяла зарезервированное пространство подкачки от освобожденной памяти.</p> <p>Значение “YES” предполагает меньшие требования к пространству подкачки и возможно меньшую активность диска, связанную с подкачкой. Значение “NO” предполагает большие требования к пространству подкачки и возможно большую активность диска, связанную с подкачкой. В некоторых ситуациях, например, при избыточном пространстве подкачки и избыточной реальной памяти, где подкачки никогда не происходит, значение NO обеспечит небольшое улучшение производительности.</p>		
DB2MEMMAXFREE	Все	По умолчанию: 8388608 байт Значения: от 0 до 2 ³² -1 байт
Эта переменная реестра управляет максимальным объемом неиспользуемой памяти в байтах, сохраняемой процессами DB2.		
DB2_ANTIJOIN	Все	По умолчанию: NO - в среде EEE По умолчанию: YES - в других средах (не EEE) Значения: YES или NO
<p>Для сред DB2 Universal Database EEE: Если задать YES, оптимизатор будет искать возможности преобразования подзапросов NOT EXISTS в антиобъединения, которые DB2 может обрабатывать более эффективно. Для других сред (не EEE): Если задать NO, оптимизатор ограничит возможности преобразования подзапросов NOT EXISTS в антиобъединения.</p>		
NEWLOGPATH2	UNIX	По умолчанию=NO Значения: YES или NO
<p>Этот параметр позволяет задать, будет ли использоваться второй путь для реализации двойной регистрации в журналах. Путь, который будет использоваться, генерируется добавлением символа '2' к текущему значению 'LOGPATH'.</p>		

DB2DOMAINLIST	Windows NT	По умолчанию: Null Значения: один или несколько допустимых доменов Windows NT (через запятую)
<p>Определяет один или несколько доменов Windows NT. Требования на соединение или подключение будут приниматься только от пользователей, принадлежащих этим доменам.</p> <p>Эту переменную реестра следует использовать только в чистой среде доменов Windows NT с серверами DB2 и клиентами, где запускается DB2 Universal Database Версии 7.1 (или новее).</p>		
DB2_LIKE_VARCHAR	Все	По умолчанию: Y,N Значения: Y, N, S, константа с плавающей запятой от 0 до 6,2
<p>Управляет сборанием и использованием статистики подэлементов. Это статистика о содержимом данных в столбцах, когда структура данных имеет вид последовательностей подполей и подэлементов, разделенных пробелами.</p> <p>От этой переменной реестра зависит, как оптимизатор поступает с предикатом данной формы: <code>COLUMN LIKE '%xxxxxx%'</code></p> <p>где xxxxxx - любая строка символов.</p> <p>Следующий синтаксис показывает, как используется эта переменная реестра: <code>db2set DB2_LIKE_VARCHAR=[Y N S num1] [,Y N S num2]</code></p> <p>где</p> <ul style="list-style-type: none"> • Элемент перед запятой или единственный элемент справа от предиката означает следующее (только для столбцов без позитивной статистики подэлементов): <ul style="list-style-type: none"> – S – Оптимизатор оценивает длину каждого элемента в сериях элементов, сочлененных в форме столбца на основе длины строки, заключенной в символы %. – Y – Значение по умолчанию. Использовать для параметра алгоритма значение по умолчанию 1,9. Использовать с параметром алгоритма алгоритм подэлементов с переменной длиной. – N – Использовать алгоритм подэлементов с постоянной длиной. – num1 – Использовать num1 в качестве параметра алгоритма с алгоритмом подэлементов с переменной длиной. • Элемент после запятой означает следующее: <ul style="list-style-type: none"> – N – Значение по умолчанию. Не собирать статистику использования подэлементов. – Y – Собирать статистику подэлементов. Использовать алгоритм подэлементов с переменной длиной, использующий собранную статистику совместно со значением по умолчанию 1,9 для параметра алгоритма в случае столбцов с позитивной статистикой подэлементов. – num2 – Собирать статистику подэлементов. Использовать алгоритм подэлементов с переменной длиной, использующий собранную статистику совместно со значением num2 в качестве параметра алгоритма для столбцов с позитивной статистикой подэлементов. 		
DB2_PINNED_BP	AIX, HP-UX	По умолчанию=NO Значения: YES или NO

Эта переменная используется для хранения глобальной памяти базы данных (включая пулы буферов), связанной с базой данных, в главной памяти на нескольких операционных системах AIX. Хранение этой глобальной памяти базы данных в главной памяти системы позволяет улучшить производительность базы данных.

Если бы, например, пул буферов сбрасывался из главной памяти системы на диск, производительность базы данных бы снизилась. Сокращение дискового ввода-вывода, когда пулы буферов размещаются в памяти системы, улучшает производительность базы данных. Если есть другие программы, которым требуется больший объем главной памяти, можно разрешить программам использование глобальной памяти базы данных с подкачкой из главной памяти в зависимости от требований к главной памяти системы.

При работе с HP-UX в 64-битной среде, кроме изменения этой переменной реестра, нужно также предоставить привилегию MLOCK группе экземпляра DB2. Для этого пользователь с правами доступа ROOT должен выполнить следующее:

1. Добавить группу экземпляра DB2 в файл /etc/privgroup. Например, если группа экземпляра DB2 принадлежит группе db2iadm1, в файл /etc/privgroup надо добавить следующую строку:
db2iadm1 MLOCK
2. Выполнить следующую команду:
setprivgrp -f /etc/privgroup

Приложение С. Средства объяснения SQL

Последний абзац в разделе “Выполнение db2expln и dynexpln” нужно заменить на:

Чтобы запустить db2expln, требуется привилегия SELECT для производных таблиц системного каталога и полномочия EXECUTE для пакета db2expln. Для выполнения dynexpln необходимо иметь полномочия BINDADD для базы данных, используемая для соединения с базой данных схема должна существовать или вы должны иметь полномочия EXPLICIT_SCHEMA для этой базы данных, а все привилегии, необходимые для объяснения этих операторов SQL. (Заметим, что все нужные уровни авторизации можно получить вместе с полномочиями SYSADM или DBADM автоматически.)

Глава 9. Command Reference

db2batch - Замер производительности

Последнее предложение в описании параметра PERF_DETAIL должно быть таким:

Значение больше 1 допустимо только на серверах DB2 Версии 2 и DB2 UDB и не поддерживается в на

db2sarp (новая команда)

db2cap - Утилита связывания статического пакета CLI/ODBC

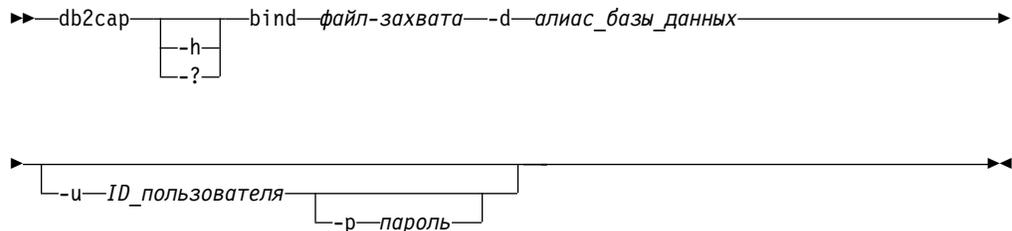
Связывает файл захвата, чтобы сгенерировать один или несколько статических пакетов. Файл захвата генерируется во время сеанса статического профилирования программы CLI/ODBC/JDBC и содержит операторы SQL, захваченные во время работы этой программы. Эта утилита обрабатывает файл захвата, чтобы он мог использоваться драйвером CLI/ODBC/JDBC для выполнения для этой программы статического SQL.

Дополнительную информацию об использовании статического SQL в прикладных программах CLI/ODBC/JDBC смотрите в описании статического профилирования в справочном руководстве *CLI Guide and Reference*.

Авторизация

- Привилегии доступа для всех объектов базы данных, используемых в операторах SQL, записанных в файл захвата.
- Достаточные полномочия для задания опций связывания, таких как OWNER и QUALIFIER, если они отличаются от ID соединения, использованного для вызова команды **db2cap**.
- Полномочия BINDADD, если этот пакет связывается первый раз; иначе требуются полномочия BIND.

Синтаксис команды



Параметры команды

-h/-? Выводит текст справки по синтаксису этой команды.

bind *файл-захвата*

Связывает операторы из этого файла захвата и создает один или несколько пакетов.

-d *алиас_базы_данных*

Задает алиас базы данных, в которой будут находиться эти один или несколько пакетов.

-u *ID_пользователя*

Задает ID пользователя, который будет использоваться для соединения с источником данных.

Примечание: Если ID пользователя не задан, проверенный ID авторизации получается от системы.

-r *пароль*

Задает пароль, который будет использоваться для соединения с источником данных.

Замечания по использованию

На платформах UNIX эта команда должна вводиться в нижнем регистре, но в операционных системах Windows и OS/2 ее можно вводить как в нижнем, так и в верхнем регистре.

Эта утилита поддерживает ряд пользовательских опций связывания, которые могут быть заданы в файле захвата. Для улучшения производительности и обеспечения защиты можно проверить и отредактировать содержимое этого файла, используя текстовый редактор для изменения этих опций.

Для создания пакета можно использовать опции связывания `SQLERROR(CONTINUE)` и `VALIDATE(RUN)`.

При использовании этой утилиты для создания пакета статическое профилирование должно быть запрещено.

Число создаваемых пакетов зависит от уровней изоляции, использованных для операторов SQL из файла захвата. Имя пакета состоит из максимум семи первых символов ключевого слова пакета их файла захвата, и одного из следующих односимвольных суффиксов:

- 0 - Чтение неприятого (UR)
- 1 - Стабильность на уровне указателя (CS)
- 2 - Стабильность чтения (RS)
- 3 - Многократное чтение (RR)
- 4 - Без принятия (NC)

Чтобы получить конкретную информацию о пакетах, можно:

- Запросить соответствующие таблицы каталога SYSIBM, указав ключевые слова `COLLECTION` и `PACKAGE` из файла захвата.
- Посмотреть файл захвата.

db2ckrst - Проверка последовательности резервных копий для инкрементного восстановления (новая команда)

Просматривает хронологию базы данных и создает список отметок времени для резервных копий, необходимых для инкрементного восстановления. Создает также упрощенный синтаксис восстановления для инкрементного восстановления вручную.

Авторизация

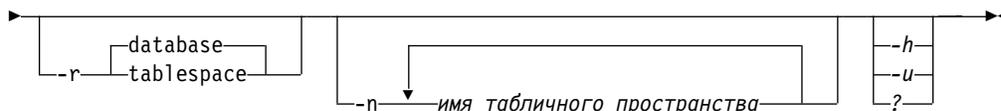
Нет

Необходимое соединение

Нет

Синтаксис команды

►► db2ckrst **-d** *имя базы данных* **-t** *отметка времени* **-r** *database* **-n** *имя табличного пространства* **-h** **-u** **?**



Параметры команды

-d *имя-базы-данных* *имя-файла*

Задаёт алиас для восстанавливаемой базы данных.

-t *отметка-времени*

Задаёт отметку времени для резервной копии, подлежащей инкрементному восстановлению.

-r

Задаёт тип восстановления, которое нужно выполнить. По умолчанию восстанавливается база данных.

Примечание: Если выбрано восстановление табличного пространства, но не указано имен табличных пространств, утилита использует для восстановления имена табличных пространств, перечисленные в записи хронологии заданной резервной копии.

-n *имя-табличного-пространства*

Задаёт имя одного или нескольких табличных пространств для восстановления.

Примечание: Если выбрано восстановление базы данных, но задан список имен табличных пространств, утилита будет выполнять восстановление указанных табличных пространств.

-h/-u/?

Выводит справку. Если указана эта опция, прочие опции игнорируются и выводится только информация справки.

Примера

```
db2ckrst -d mr -t 20001015193455 -r database
db2ckrst -d mr -t 20001015193455 -r tablespace
db2ckrst -d mr -t 20001015193455 -r tablespace -n tbsp1 tbsp2
```

```
> db2 backup db mr
```

```
Backup successful. The timestamp for this backup image is : 20001016001426
```

```
> db2 backup db mr incremental
```

```
Backup successful. The timestamp for this backup image is : 20001016001445
```

```

> db2ckrst -d mr -t 20001016001445

Suggested restore order of images using timestamp 20001016001445 for database mr.
=====
db2 restore db mr incremental taken at 20001016001445
db2 restore db mr incremental taken at 20001016001426
db2 restore db mr incremental taken at 20001016001445
=====

> db2ckrst -d mr -t 20001016001445 -r tablespace -n userspace1
Suggested restore order of images using timestamp 20001016001445 for database mr.
=====
db2 restore db mr tablespace ( USERSPACE1 ) incremental taken at 20001016001445
db2 restore db mr tablespace ( USERSPACE1 ) incremental taken at 20001016001426
db2 restore db mr tablespace ( USERSPACE1 ) incremental taken at 20001016001445
=====

```

Замечания по использованию

Для использования этой утилиты необходимо наличие хронологии базы данных. Если хронологии базы данных нет, следует перед использованием утилиты задать опцию HISTORY FILE в команде RESTORE.

Если используется опция FORCE команды PRUNE HISTORY, есть вероятность удаления записей, требуемых для восстановления из самой поздней полной резервной копии базы данных. По умолчанию команда PRUNE HISTORY не удаляет нужные записи. Не рекомендуется использовать опцию FORCE команды PRUNE HISTORY.

Рекомендуется хранить подробные записи резервных копий, используя для ориентации в них эту утилиту.

db2gncol (новая команда)

db2gncol - Изменить значения генерируемых столбцов

Изменяет значения генерируемых столбцов в таблицах, находящихся в состоянии отложенной проверки и имеющих ограниченное пространство журналов. Эта утилита используется для подготовки оператора SET INTEGRITY для таблицы, содержащей столбцы, для генерации которых используются выражения.

Авторизация

Одна из следующих привилегий:

- *sysadm*
- *dbadm*

Синтаксис команды

```
► db2gncol -d база_данных -s имя_схемы -t имя_таблицы  
► -c счетчик_принятий  
└─┬─u ID_пользователя -p password ─┬─h ─┘
```

Параметры команды

- d *база_данных*
Задаёт алиас базы данных, в которой находится эта таблица.
- s *имя_схемы*
Задаёт имя схемы для этой таблицы. Имя схемы регистрозависимо.
- t *имя_таблицы*
Задаёт таблицу, для столбцов которой нужно вычислить по формулам новые значения генерируемых столбцов. Имя таблицы регистрозависимо.
- c *счетчик_принятий*
Задаёт число строк, обновляемых между операциями принятия. Этот параметр влияет на размер пространства журналов, необходимого для генерации значений столбцов.
- u *ID_пользователя*
Задаёт ID пользователя, обладающего привилегиями администратора системы или администратора базы данных. Если эта опция не задана, используется текущий ID пользователя.
- p *пароль*
Задаёт пароль для указанного ID пользователя.
- h
Выводит справку. Если указана эта опция, прочие опции игнорируются, и выводится только информация справки.

Замечания по использованию

Использование этой утилиты вместо опции FORCE GENERATED оператора SET INTEGRITY может быть необходимым, если таблица имеет большой размер и при этом:

- Необходимо заново сгенерировать все значения столбца, так как было изменено выражение генерации этого столбца.
- Используемая в генерируемом столбце внешняя пользовательская функция была изменена и поэтому нужно изменить многие значения столбца.
- В таблицу был добавлен генерируемый столбец.

- Была выполнена операция загрузки (или загрузки с добавлением) большого объема данных, во время которой не были заданы значения для генерируемых столбцов.
- Слишком мало пространство журналов (из-за выполнения длительных одновременных транзакций или из-за большого размера таблицы).

Эта утилита сгенерирует значения всех столбцов, созданных на основе выражений генерации. В процессе обновления таблицы регулярно выполняются операции принятия, чтобы избежать переполнения пространства журналов. После выполнения утилиты **db2gncol** таблицу можно вывести из состояния отложенной проверки, используя оператор SET INTEGRITY.

db2inidb - инициализировать зеркальную копию базы данных

В среде с отделенной зеркальной копией **db2inidb** используется для инициализации зеркальной копии базы данных с различными целями.

Авторизация

Надо иметь одни из следующих полномочий:

- *sysadm*
- *sysctrl*
- *sysmaint*

Необходимое соединение

Нет

Синтаксис команды



Параметры команды

алиас_базы_данных

Указывает алиас инициализируемой базы данных.

SNAPSHOT

Эта опция используется для инициализации зеркальной копии базы данных как клона (или снимка) первичной базы данных. Эта база данных предназначена только для чтения.

STANDBY

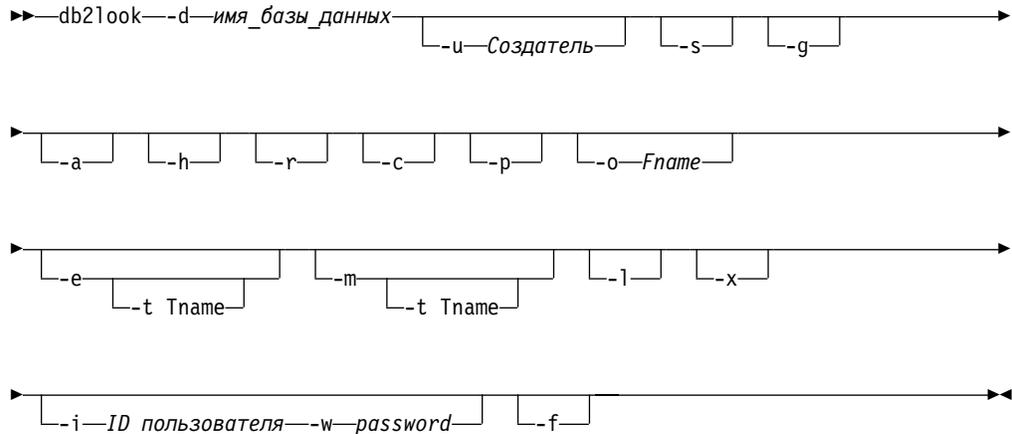
Эта опция позволяет постоянно повторять транзакции для зеркальной копии базы данных при помощи файлов журнала. Новые файлы журнала первичной базы данных считываются и применяются к этой резервной базе данных. Поэтому она может быть использована в качестве подменяющей базы данных при выходе из строя первичной базы данных.

MIRROR

Эта опция позволяет использовать зеркальную копию базы данных в качестве образа резервной копии, который можно восстановить вместо первичной базы данных.

db2look - Утилита извлечения статистики DB2

Синтаксическая диаграмма должна выглядеть так:



После параметра -c добавлен параметр -td x. Его определение:

Задаёт разделитель для операторов SQL, генерируемых db2look. Если опция не задана, по умолчанию используется точка с запятой ';'. Рекомендуется использовать эту опцию, если задана опция -e. В этом случае в извлекаемых объектах могут содержаться триггеры или процедуры SQL.

Кроме того, добавлен следующий пример:

Генерация операторов DDL для объектов, созданных всеми пользователями в базе данных DEPARTMENT. Вывод db2look направляется в файл db2look.sql:

```
db2look -d department -a -e -td % -o db2look.sql  
db2 -td% -f db2look.sql
```

db2updv7 - обновление базы данных до уровня текущего пакета FixPak Версии 7

Эта команда обновляет системные каталоги в базе данных для поддержки текущего пакета FixPak:

- Позволяет использовать новые встроенные функции (ABS, DECRYPT_BIN, DECRYPT_CHAR, ENCRYPT, ROUND и MULTIPLY_ALT).
- Позволяет использовать новые встроенные процедуры (GET_ROUTINE_SAR и PUT_ROUTINE_SAR).
- Добавляет функции WEEK_ISO и DAYOFWEEK_ISO или применяет изменения для них в базах данных Windows и OS/2.
- Применяет изменения к упакованным описателям таблиц для таблиц, перенастроенных из Версии 2 в Версию 6.
- Создает производную таблицу SYSCAT.SEQUENCES.

Авторизация

sysadm

Необходимое соединение

База данных. Эта команда автоматически устанавливает соединение с указанной базой данных.

Синтаксис команды

```
► db2upd7 -d имя_базы_данных [-h] ►
```

Параметры команды

-d имя_базы_данных

Имя обновляемой базы данных.

-h

Вывести справочную информацию. Если указана эта опция, прочие опции игнорируются и выводится только информация справки.

Пример

После установки FixPak обновите системный каталог в базе данных примера при помощи команды:

```
db2upd7 -d sample
```

Замечания по использованию

Этот инструмент можно использовать для базы данных DB2 Версии 7.1, только если установлен по крайней мере FixPak 2. Если эта команда запускается несколько раз, сообщений об ошибках не выводится, но все изменения применяются к каталогам только один раз.

Чтобы ввести в действие новые встроенные функции, надо отключить от этой базы данных все прикладные программы, а если база данных была активирована, ее следует деактивировать.

Новая опция процессора командной строки (-x, подавление вывода заголовков столбцов)

Новая опция, -x, указывает процессору командной строки, что он должен возвращать данные без заголовков, в том числе без имен столбцов. Значение по умолчанию этой опции команды - OFF.

Для процессора командной строки DB2 необходим шрифт True Type

Чтобы национальные символы языков с однобайтным набором символов (SBCS) правильно выводились в окне процессора командной строки DB2, измените шрифт на True Type.

ADD DATALINKS MANAGER

Для этой команды требуется следующий уровень авторизации:

- *sysadm*
- *sysctrl*
- *sysmaint*

Следует добавить следующее замечание по использованию:

| Эта команда начинает действовать только после того, как от базы данных
| отключатся все прикладные программы. Для успешного выполнения команды
| добавляемый менеджер связей данных DB2 должен быть полностью
| сконфигурирован и запущен. Кроме того, база данных должна быть
| зарегистрирована в менеджере связей баз данных DB2 при помощи команды d1 fm
| add_db. Максимальное число менеджеров связей баз данных DB2, которое можно
| добавить в одну базу данных - 16.

| **ARCHIVE LOG (новая команда)**

|

Archive Log

Закрывает и усекает активный файл журнала для восстановимой базы данных. Если включен обработчик пользователя, посылает требование на архивирование.

Авторизация

Одна из следующих привилегий:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

Синтаксис команды

```
➤ ARCHIVE LOG FOR DATABASE алиас-базы-данных
                   DB
➤ USER имя-пользователя USING пароль
                   Условие узла
```

Условие узла:

```
ON Условие списка узлов
ALL NODES EXCEPT Условие списка узлов
```

Условие списка узлов:

```
NODE (номер узла TO номер узла)
NODES
```

Параметры команды

DATABASE алиас-базы-данных

Задаёт алиас базы данных, для которой архивируется активный журнал.

USER имя-пользователя

Задаёт имя пользователя для попытки соединения.

USING пароль

Задаёт пароль для имени пользователя.

ON ALL NODES

Указывает, что команду следует выполнить для всех узлов в файле db2nodes.cfg. Это значение используется по умолчанию, если не задано условие узла.

EXCEPT

Указывает, что команду следует выполнить для всех узлов в файле db2nodes.cfg, кроме приведенных в списке узлов.

ON NODE/ON NODES

Указывает, что следует архивировать журналы для заданной базы данных на наборе узлов.

номер узла

Задаёт номер узла в списке узлов.

ТО номер узла

Используется при задании диапазона узлов, для которого следует архивировать журналы. В список включаются все узлы от первого указанного номера узла до второго указанного номера узла включительно.

Замечания по использованию

Основное назначение этой команды - собрать полный набор файлов журнала до некоторой точки. Затем файлы журнала можно использовать для обновления резервной базы данных.

Эта функция работает, только когда с заданной базой данных нет соединений. Таким образом, пользователь не может выполнить команду с непринятыми транзакциями. Команда ARCHIVE LOG не вызывает принудительного приема незаконченных пользовательских транзакций. Если при выполнении команды уже есть соединение с заданной базой данных, она прекращает работу и возвращает ошибку. Если при выполнении команды другая программа производит транзакции с заданной базой данных, возможно небольшое ухудшение производительности, так как команда сбрасывает буфер журнала на диск. Всем остальным транзакциям, пытающимся сохранять в буфер журнальные записи, придется ожидать окончания записи на диск.

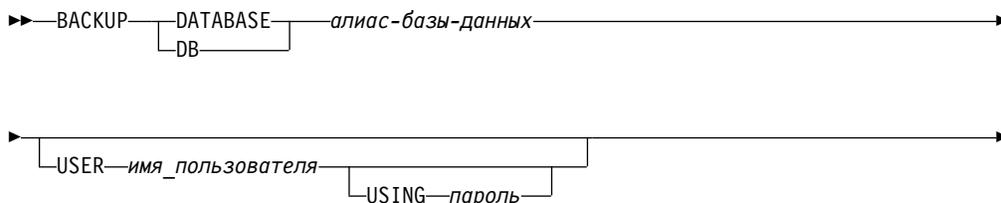
При использовании в среде MPP можно задать подмножество узлов при помощи условия узлов. Если условие узлов не задано, по умолчанию эта команда закрывает и архивирует активный журнал на всех узлах.

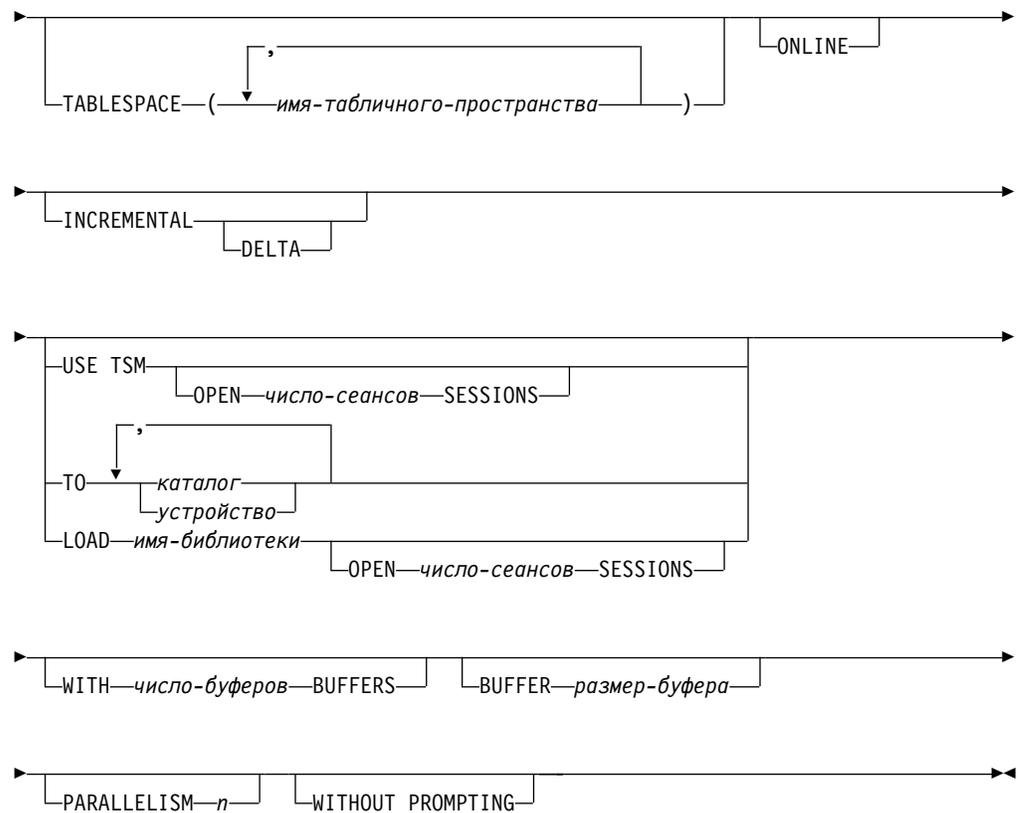
Использование этой команды вызывает потерю базой данных части пространства LSN (последовательных номеров записей журнала) и, как следствие, исчерпание диапазона возможных LSN.

BACKUP DATABASE

Синтаксическая диаграмма

Синтаксическая диаграмма для BACKUP DATABASE обновлена, чтобы отразить новый параметр INCREMENTAL и необязательный аргумент DELTA. Задание одной опции INCREMENTAL вызовет создание инкрементной (кумулятивной) резервной копии. Необязательный аргумент DELTA задаёт создание разностной резервной копии.





Особенности менеджера связей данных DB2

Если для базы данных сконфигурирован один или несколько серверов связей данных, операция завершится успешно, даже если ни один сервер связей данных не доступен. После перезапуска сервера связей данных до того, как он станет доступным базе данных, на нем будет завершено резервное копирование.

Примечание: Если операций резервного копирования, ожидающих недоступный сервер связей данных, вдвое больше, чем сохраняется в файле хронологии базы данных (параметр конфигурации базы данных num_db_backups), операция резервного копирования завершится неудачно.

BIND

В синтаксис этой команды DB2 надо включить параметр federated:

```
FEDERATED---NO---
          '-YES--'
```

FEDERATED

Указывает, ссылается ли статический оператор SQL в пакете на псевдоним или на производную таблицу объединения. Если эта опция не указана и статический оператор SQL в пакете ссылается на псевдоним или производную таблицу объединения, пакет создается, но возвращается предупреждение.

NO В статических операторах SQL пакета нет ссылок на псевдонимы или производные таблицы объединения. Если во время подготовки или

связывания этого пакета в статическом операторе SQL встретится псевдоним или производная таблица объединения, возвращается ошибка, и пакет *не* создается.

YES В статических операторах SQL пакета могут быть ссылки на псевдонимы или производные таблицы объединения. Если во время подготовки или связывания этого пакета в статическом операторе SQL не встретятся псевдонимы или производные таблицы объединения, не возвращается ни ошибок, ни предупреждений, а пакет создается.

Примечание: В Версии 7 FixPak 2 при прекомпиляции исходного файла или связывании файла связывания без указания значения для опции FEDERATED сервер генерирует предупреждение SQL1179W. Такое же сообщение генерируется, когда исходный файл или файл связывания содержит ссылку статического SQL на псевдоним. Есть два исключения:

- Для клиентов более ранних уровней, чем Версия 7 FixPak 2 или для клиентов предыдущих версий API sqlprep() не записывает это предупреждение SQL1179W в файл сообщений. В этом случае команда PRECOMPILE процессора командной строки также не выдает это предупреждение.
- Для клиентов более ранних уровней, чем Версия 7 FixPak 2 или для клиентов предыдущих версий API sqlabndx не записывает это предупреждение SQL1179W в файл сообщений. Однако в файл сообщений ошибочно включается сообщение SQL0092N, указывающее на то, что пакет не был создан. Это неправильно, так как на самом деле пакет создается. Команда BIND процессора командной строки возвращает то же самое ошибочное предупреждение.

CALL

Синтаксис команды CALL должен выглядеть так:



Описание параметра *аргумент* должно быть изменено на следующее:

Задаёт один или несколько аргументов для хранимой процедуры. Все входные и выходные аргументы должны быть заданы в порядке, определенном для этой процедуры. Вместо выходных аргументов задается символ "?". Например, хранимую процедуру foo с одним входным параметром типа целое число и одним выходным параметром надо вызывать как "call foo (4; ?)".

Примечания:

1. При вызове этой утилиты из командной строки операционной системы команду нужно задавать в кавычках:

```
"call DEPT_MEDIAN (51)"
```

Можно также использовать одиночные кавычки (').

2. Вызываемая хранимая процедура должна иметь уникальное имя в базе данных.

3. Эта хранимая процедура должна быть внесена в каталог. При вызове процедуры, не внесенной в каталог, выдается сообщение об ошибке DB21036.
4. Сообщение DB21101E выдается, если в командной строке задано недостаточное число параметров или порядок, в котором заданы эти параметры (входные, выходные), не совпадает с их порядком в определении этой хранимой процедуры.
5. Столбец результатов может содержать не более 1023 символов.
6. Большие объекты или двоичные данные (FOR BIT DATA, VARBINARY, LONGVARBINARY, GRAPHIC, VARGAPHIC или LONGVARGRAPHIC) не поддерживаются.
7. Утилита CALL поддерживает наборы результатов.
8. Если в вызове используется SP с выходной переменной (OUTPUT) неподдерживаемого типа, возникает ошибка CALL и выдается сообщение DB21036.
9. Максимальная длина входного параметра (INPUT) для CALL равна 1024.

DROP DATALINKS MANAGER (новая команда)

DROP DATALINKS MANAGER

Отбрасывает менеджер связей данных DB2 из списка зарегистрированных менеджеров связей данных DB2 для заданной базы данных.

Авторизация

Одни из следующих:

- *sysadm*
- *sysctrl*
- *sysmaint*

Синтаксис команды

```
►► DROP DATALINKS MANAGER FOR DATABASE имя_базы_данных USING имя ◀◀
```

└──┬──┘
DB

Параметры команды

DATABASE имя_базы_данных

Задаёт имя базы данных.

USING имя

Задаёт имя сервера связей данных DB2, как его выводит команда LIST DATALINKS MANAGER.

Примера

Пример 1

Отбрасывание менеджера связей данных DB2 micky.almaden.ibm.com из базы данных test в экземпляре validate, находящейся на хосте bramha.almaden.ibm.com, в то время как в некоторых таблицах базы есть связи с micky.almaden.ibm.com. *При отбрасывании менеджера связей данных DB2 крайне важно выполнить следующие действия.*

1. Сделайте резервную копию базы данных test.
2. Если есть какие-либо связи с micky.almaden.ibm.com, отмените их:
 - a. Войдите в систему с ID пользователя из SYSADM_GROUP и соединитесь с базой данных test в монопольном режиме.

```
connect to test in exclusive mode
```

Это должно быть единственное соединение с test под этим ID пользователя. Это гарантирует, что не будут созданы новые связи.
 - b. Получите список всех столбцов FILE LINK CONTROL DATALINK и таблиц, содержащих эти столбцы, в базе данных.

```
select tabname, colname from syscat.columns where substr(dl_features, 2, 1) = 'F'
```
 - c. Для каждого столбца FILE LINK CONTROL DATALINK в списке выполните SQL SELECT, чтобы выяснить, есть ли связи с micky.almaden.ibm.com. Например, для столбца DATALINK c в таблице t оператор SELECT будет выглядеть как:

```
select count(*) from t where dlurlserver(t.c) = 'MICKY.ALMADEN.IBM.COM'
```
 - d. Для каждого столбца FILE LINK CONTROL DATALINK, содержащего такие связи, выполните SQL UPDATE, чтобы удалить связи с micky.almaden.ibm.com. Например, для столбца DATALINK c в таблице t оператор UPDATE будет выглядеть как:

```
update t set t.c = null where dlurlserver(t.c) = 'MICKY.ALMADEN.IBM.COM'
```

Если t.c не допускает пустых значений, можно использовать:

```
update t set t.c = dvalue('') where dlurlserver(t.c) = 'MICKY.ALMA DEN.IBM.COM'
```

e. Выполните принятие для этого оператора SQL UPDATE:

```
commit
```

3. Выполните команду DROP DATALINKS MANAGER:

```
drop datalinks manager for db test using node micky.almaden.ibm.com
```

4. Прервите соединение в монопольном режиме, чтобы изменения вступили в силу и другие могли соединяться с этой базой данных:

```
terminate
```

5. Запустите удаление связей и чистку мусора информации резервного копирования для базы test на сервере micky.almaden.ibm.com. В качестве администратора менеджера связей данных DB2 выполните на сервере micky.almaden.ibm.com следующую команду:

```
dlfm drop_dlm test validate bramha.almaden.ibm.com
```

Это удалит связи всех файлов, которые все еще связаны с базой данных test, на случай, если пользователь забыл удалить их до перехода к шагу 3. Если на micky.almaden.ibm.com есть информация резервного копирования (например, файлы архивов, метаданные) для файлов, ранее связанных с базой test, эта команда запустит чистку мусора для этой информации. Фактическое удаление связей и чистка мусора будут выполняться асинхронно.

Пример 2

Удаление значений DATALINK, являющихся связями с файлами менеджера связей данных DB2 micky.almaden.ibm.com, в то время как менеджер уже отброшен из базы данных test. Это может потребоваться, если при отбрасывании micky.almaden.ibm.com не выполнялись шаги из Примера 1. Операторы SQL DELETE, SELECT и UPDATE не сработают для таких значений DATALINK (SQL0368). Пользователь должен выполнить операцию согласования для всех таблиц, содержащих такие значения DATALINK. Все значения DATALINK, которые были связями с micky.almaden.ibm.com, будут заменены на NULL или значение DATALINK нулевой длины. Все строки, содержащие такие значения, будут вставлены в таблицу исключений (если она была задана). Однако значение DATALINK не будет содержать имени-префикса. Имя-префикс в оригинальном значении DATALINK больше не может быть получено системой, поскольку micky.almaden.ibm.com отброшен. Например, если оригинальное значение DATALINK было 'http://host.com/dlfs/x/y/a.b', а '/dlfs' - имя-префикс, значение DATALINK в таблице исключений будет 'http://host.com/x/y/a.b'. Файлы, на которые ссылаются эти значения DATALINK, остаются в менеджере связей данных DB2 в связанном состоянии. Чтобы запустить удаление связей для этих файлов, следует ввести на сервере micky.almaden.ibm.com команду dlfm drop_dlm. Если на micky.almaden.ibm.com есть информация резервного копирования (например, файлы архивов, метаданные) для файлов, ранее связанных с базой test, эта команда запустит чистку мусора этой информации. Фактическое удаление связей и чистка мусора будут выполняться асинхронно.

Пример 3

Несколько экземпляров менеджера связей данных DB2 micky.almaden.ibm.com для базы данных test. В этом сценарии показано, как можно перерегистрировать менеджер связей данных DB2 после его отбрасывания, и как он воспринимается в качестве абсолютно нового менеджера связей данных DB2. Нижеприведенные шаги - лишь пример возможного сценария. Если, как рекомендовалось выше, при

отбрасывании micky.almaden.ibm.com выполнялись шаги из Примера 1, связей со старым воплощением micky.almaden.ibm.com не осталось, и ниже вы не увидите на шаге 7 ошибку SQL0368.

1. Зарегистрируйте micky.almaden.ibm.com в базе данных test:

```
add datalinks manager for db test using node micky.almaden.ibm.com port 14578
```

2. Создайте связи с файлами на micky.almaden.ibm.com:

```
connect to test
create table t(c1 int, c2 datalink linktype url file link control mode dboptions)
insert into t values(1, dvalue('file://micky.almaden.ibm.com/pictures/yosemite.jpg'))
commit
terminate
```

3. Отбросьте micky.almaden.ibm.com из базы данных test:

```
drop datalinks manager for db test using micky.almaden.ibm.com
```

4. Выберите значения DATALINK:

```
connect to test
select * from t
terminate
```

Пользователь увидит следующее:

```
SQL0368 Менеджер связей данных DB2 "MICKY.ALMA DEN.IBM.COM" не зарегистрирован для этой базы да
```

5. Снова зарегистрируйте micky.almaden.ibm.com для базы данных test:

```
add datalinks manager for db test using node micky.almaden.ibm.com port 14578
```

6. Вставьте новые значения DATALINK:

```
connect to test
insert into t values(2, dvalue('file://micky.almaden.ibm.com/pictures/tahoe.jpg'))
commit
```

7. Оператор SELECT для значений DATALINK:

```
select c2 from t where c1 = 2
```

завершится успешно, так как выбираемое значение - связь с зарегистрированным в данный момент экземпляром micky.almaden.ibm.com.

```
select c2 from t where c1 = 1
```

возвратит:

```
SQL0368 Менеджер связей данных DB2 "MICKY.ALMA DEN.IBM.COM" не зарегистрирован для этой базы да
```

так как выбираемое значение - связь с экземпляром micky.almaden.ibm.com, отброшенным ранее на шаге 3.

Замечания по использованию

Последствия выполнения команды DROP DATALINKS MANAGER нельзя отменить. При использовании команды DROP DATALINKS MANAGER чрезвычайно важно следовать шагам из Примера 1.

Эта команда действует только после того, как от базы данных отключатся все прикладные программы.

После успешного завершения команды пользователю сообщается (DB210201I), что Менеджер связей данных DB2 не выполняет операций для этих связей. До того, как отбросить менеджер связей данных DB2, пользователь должен убедиться, что в базе данных нет связей с файлами на этом менеджере связей данных DB2. Если после отбрасывания менеджера связей данных DB2 в базе данных остаются связи, пользователь должен удалить их при помощи утилиты согласования. Утилита согласования заменит эти связи на NULL (если столбец DATALINK допускает пустые значения) или на значение DATALINK нулевой длины.

Файлы, соответствующие связям между базой данных и отброшенным менеджером связей данных DB2, остаются в связанном состоянии. Это значит, что для них недоступны такие операции, как чтение, запись, переименование, удаление, изменение разрешений или владельцев.

При очистке мусора эта команда не удаляет архивированные копии файлов удаленных связей на менеджере связей данных DB2. Однако пользователи могут в явном виде запускать операцию удаления связей и очистку мусора при помощи команды `dlfm drop_dlm` на менеджере связей данных DB2.

Перед отбрасыванием менеджера связей данных DB2 рекомендуется сделать резервную копию базы данных. Кроме того, убедитесь, что для всех зарегистрированных репликаций реплицированы все изменения, затрагивающие этот менеджер связей данных DB2.

Если до отбрасывания менеджера связей данных DB2 из базы данных была сделана резервная копия, и эта копия используется для восстановления после отбрасывания этого менеджера связей данных DB2, операция восстановления или повтора транзакций может перевести некоторые таблицы в состояние отложенного согласования связей данных (DRP).

EXPORT

В разделе "DB2 Data Links Manager Considerations" описание шага 3 процедуры обеспечения согласованного копирования для экспорта таблицы и соответствующих файлов, указанных в столбцах `DATALINK`, должно выглядеть так:

3. Выполните утилиту `dlfm_export` на каждом сервере связей данных. В качестве входного параметра утилиты `dlfm_export` задается имя управляющего файла, сгенерированного утилитой экспорта. Будет создан файл архива `tar` (или эквивалентный), содержащий файлы, перечисленные в заданном управляющем файле. В распределенных файловых системах (DFS) утилита `dlfm_export` получит корневые полномочия сети DCE перед архивированием файлов, перечисленные в управляющем файле. Утилита `dlfm_export` не сохраняет для архивируемых файлов информацию ACL.

В том же разделе текст после фразы "Successful execution of EXPORT results in the generation of the following files" (При успешном выполнении утилиты EXPORT генерируются следующие файлы) должен быть изменен:

Второе предложение в первом абзаце должно выглядеть так:

Значение столбца `DATALINK` в этом файле имеет тот же формат, который используется утилитами импорта.

Первое предложение во втором абзаце должно выглядеть так:

Управляющие файлы `имя_сервера`, генерируемые для каждого сервера связей данных. (В операционной системе Windows NT для всех серверов связей данных используется один управляющий файл `ctrlfile.lst`. В DFS используется один управляющий файл для каждой ячейки.)

В абзац перед таблицей 5 нужно добавить следующее предложение:

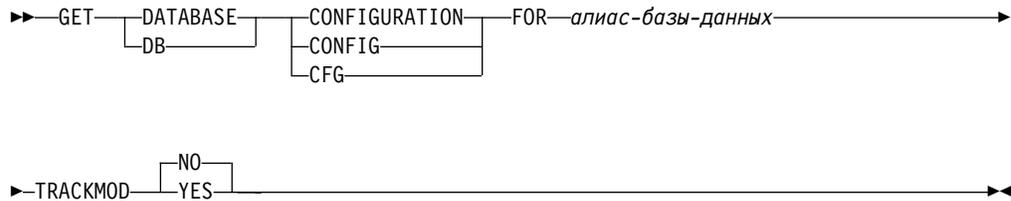
Дополнительную информацию об утилите `dlfm_export` смотрите в разделе "Using Export to move DB2 Data Links Manager Data" справочного руководства "Data Movement Utilities Guide and Reference".

GET DATABASE CONFIGURATION

Описание параметра конфигурации `DL_TIME_DROP` должно быть изменено на следующее:

Применяется только для менеджера связей данных DB2.
Этот параметр задает интервал времени (в днях), в течение которого файлы должны сохраняться на сервере архивов (например, сервере TSM) после выполнения команды DROP DATABASE.

Для команды GET DATABASE CONFIGURATION добавлен новый параметр TRACKMOD. Синтаксис команды теперь выглядит так:



Надо добавить описание этого параметра:

TRACKMOD

Задаёт, должна ли DB2 отслеживать измененные страницы в базе данных для возможности снятия инкрементных копий.

OFF Задаёт, что не нужно отслеживать измененные страницы. Это значение по умолчанию для баз данных, созданных до Версии 7.1, FixPak 3.

ON Указывает, что нужно отслеживать измененные страницы. Когда включен этот параметр, можно выполнять инкрементное резервное копирование базы данных. Это значение по умолчанию для баз данных, созданных Версией 7.1, FixPak 3 и новее.

GET ROUTINE (новая команда)

GET ROUTINE

Получает файл архива SQL (SAR) для заданной подпрограммы SQL.

Авторизация

dbadm

Необходимое соединение

База данных. Если разрешено неявное соединение, устанавливается соединение с базой данных по умолчанию.

Синтаксис команды

```
▶▶ GET ROUTINE INTO имя_файла FROM SPECIFIC PROCEDURE 

▶ имя_процедуры ▶▶
```

Параметры команды

INTO *имя-файла*

Задаёт имя файла, куда записывается архив подпрограмм SQL (SAR).

FROM

Указывает, что надо получить начало спецификации подпрограммы.

SPECIFIC

Заданное *имя-подпрограммы* представляет собой конкретное имя.

PROCEDURE

Подпрограмма является процедурой SQL.

имя-подпрограммы

Имя процедуры. Если задана опция SPECIFIC, это конкретное имя процедуры. Если в имени не указано имя схемы, используется CURRENT SCHEMA. Подпрограмма *имя-подпрограммы* должна быть существующей процедурой, определенной, как процедура SQL.

Примера

```
GET ROUTINE INTO procs/proc1.sar FROM PROCEDURE myapp1.proc1;
```

GET SNAPSHOT

Описание параметра FCM FOR ALL NODES должно выглядеть так:

Предоставляет статистику менеджера FCM по связи между узлом, для которого выполнялась команда GET SNAPSHOT, и другими узлами в экземпляре EEE.

IMPORT

В разделе "DB2 Data Links Manager Considerations" в шаг 3 надо добавить следующее предложение:

Для распределенных файловых систем (DFS) при необходимости обновите информацию об имени ячейки в URL (столбцов DATALINK) из экспортируемых данных для таблицы SQL.

В шаг 4 нужно добавить следующее предложение:

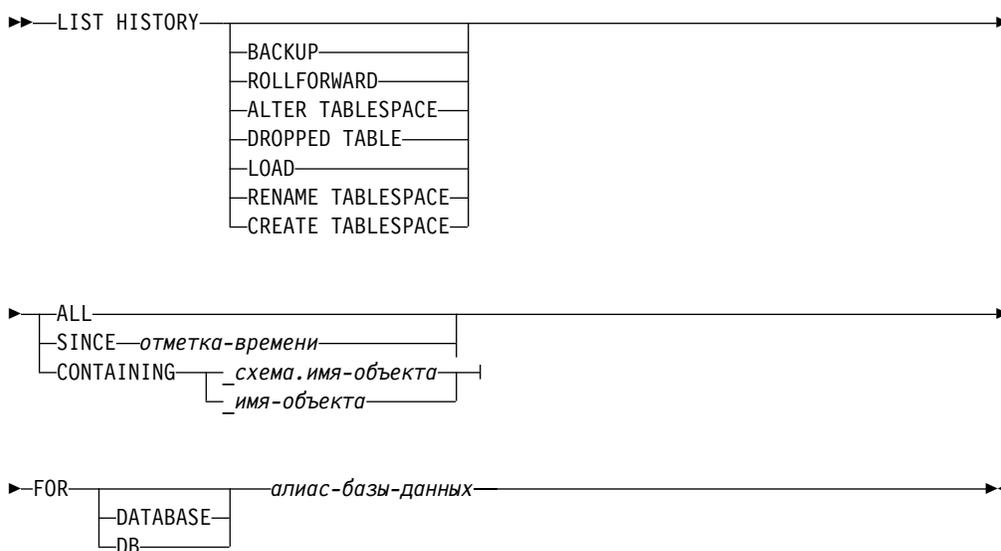
Для DFS в файле конфигурации менеджера связей данных DB2 определите ячейки конфигурации назначения

Абзац после шага 4 должен выглядеть так:

При выполнении утилиты импорта для базы данных назначения файлы, указанные в данных столбцов DATALINK, связываются с соответствующими серверами связей данных.

LIST HISTORY

В команде LIST HISTORY добавлен параметр CREATE TABLESPACE.
Синтаксическая диаграмма будет выглядеть так:



Надо добавить следующее описание параметра:

CREATE TABLESPACE

Выводит список всех операций CREATE TABLESPACE и DROP TABLESPACE.

Замечания по использованию изменены так:

В раздел отчета Операция будут добавлены следующие обозначения:

- А - Создание табличного пространства
- О - Отбрасывание табличного пространства
- U - Выгрузка

Обозначения в разделе отчета Тип теперь будут выглядеть так:

- Типы резервного копирования
 - F - Автономное
 - N - Оперативное
 - I - Инкрементное автономное
 - O - Инкрементное оперативное
 - D - Разностное автономное
 - E - Разностное оперативное

- Типы повтора транзакций
 - E - До конца журнала
 - P - До момента времени
- Типы загрузки
 - I - Вставка
 - R - Замена
- Типы изменения табличного пространства
 - C - Добавление контейнеров
 - R - Перераспределение
- Типы стабилизации
 - S - Стабилизация с общим доступом
 - U - Стабилизация с изменением
 - X - Монопольная стабилизация
 - Z - Сброс стабилизации

LOAD

В разделе "DB2 Data Links Manager Considerations" добавьте следующее предложение в Шаг 1 процедуры, которая должна выполняться перед запуском утилиты загрузки, если данные загружаются в таблицу со столбцом DATALINK, определенным как FILE LINK CONTROL:

Для распределенных файловых систем (DFS) убедитесь, что зарегистрированы менеджеры связей данных

В Шаг 5 нужно добавить следующее предложение:

Для DFS в файле конфигурации менеджера связей данных DB2 зарегистрируйте ячейки конфигурации назначения, на которые указывают загружаемые данные DATALINK.

В разделе "Representation of DATALINK Information in an Input File" первое примечание после описания параметра urlname должно выглядеть так:

В настоящее время в качестве имен схем разрешено использовать "http", "file", "unc" и "dfs".

Первое предложение второго примечания должно выглядеть так:

Префикс (схема, хост и порт) имени URL необязателен.
 Для DFS этот префикс указывает часть имени объединенного файлового пространства, определяющую ячейку схемы.

В примерах данных DATALINK для форматов файлов ASCII с ограничителями (DEL) и ASCII без ограничителей (ASC) нужно удалить третий пример.

Примеры данных DATALINK, в которых для спецификации загрузки или импорта для столбца предполагается, что он DL_URL_DEFAULT_PREFIX, должны быть удалены и заменены на следующие:

Примеры данных DATALINK, в которых для спецификации загрузки или импорта для столбца предполагается, что он DL_URL_REPLACE_PREFIX ("http://qso"):

- <http://www.almaden.ibm.com/mrep/intro.mpeg>

Он хранится со следующими частями:

- scheme = http
- server = qso
- path = /mrep/intro.mpeg
- comment = пустая строка

- /u/me/myfile.ps
- Он хранится со следующими частями:
- scheme = http
 - server = qso
 - path = /u/me/myfile.ps
 - comment = пустая строка

PING (новая команда)

PING

Проверяет время ответа сети для связи между клиентом и сервером базы данных, если для установления соединения используется DB2 Connect.

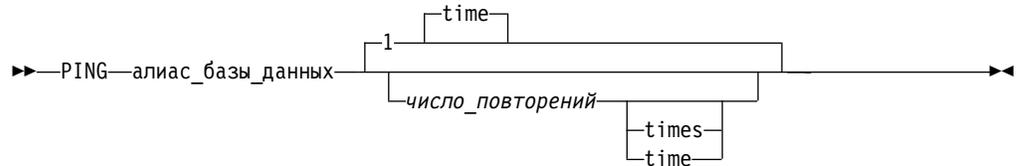
Авторизация

Нет

Необходимое соединение

База данных

Синтаксис команды



Параметры команды

алиас_базы_данных

Указывает алиас для базы данных на сервере DRDA, которому посылается сигнал ping.

Примечание: Этот параметр, хотя он и обязательный, в настоящее время не используется. Он зарезервирован для будущего использования. Может быть указано любое действительное имя алиаса базы данных.

число_повторений

Указывает число повторений этой проверки. Значение должно находиться в диапазоне от 1 до 32767 включительно. Значение по умолчанию - 1. При каждом повторении будет возвращено одно значение времени.

Примера

Для однократной проверки времени ответа сети для соединения с сервером базы данных хоста hostdb:

```
db2 ping hostdb 1  
или:  
db2 ping hostdb
```

Результат выполнения этой команды будет выглядеть примерно так:

```
Истекшее время: 7221 микросекунд
```

Чтобы проверить время ответа сети для соединения с сервером базы данных хоста hostdb пять раз:

```
db2 ping hostdb 5  
или:  
db2 ping hostdb 5 times
```

Результат выполнения этой команды будет выглядеть примерно так:

```
Истекшее время: 8412 микросекунд  
Истекшее время: 11876 микросекунд  
Истекшее время: 7789 микросекунд  
Истекшее время: 10124 микросекунд  
Истекшее время: 10988 микросекунд
```

Замечания по использованию

Соединение базы данных должно существовать до вызова этой команды; в противном случае будет возвращена ошибка.

Истекшее время возвращается для соединения между клиентом и сервером базы данных DRDA через DB2 Connect.

PUT ROUTINE (новая команда)

PUT ROUTINE

При помощи заданного файла архива процедур SQL (SAR) определить подпрограмму в базе данных.

Авторизация

dbadm

Необходимое соединение

База данных. Если разрешено неявное соединение, устанавливается соединение с базой данных по умолчанию.

Синтаксис команды



Параметры команды

FROM имя-файла

Задаёт имя файла, где записан архив подпрограмм SQL (SAR).

OWNER новый-владелец

Задаёт для подпрограммы новое имя авторизации, которое будет использоваться при проверке авторизации подпрограммы. У пользователя новый-владелец должны быть необходимые привилегии для определения подпрограммы. Если не задано условие OWNER, используется имя авторизации, под которым изначально была определена подпрограмма.

USE REGISTERS

Указывает, что для определения подпрограммы используются специальные регистры CURRENT SCHEMA и CURRENT PATH. Если это условие не задано, для схемы и пути SQL по умолчанию используются параметры, использовавшиеся при изначальном определении подпрограммы. CURRENT SCHEMA используется в качестве имени схемы для неспецифицированных имен объектов (включая имя подпрограммы), а CURRENT PATH - для разрешения неспецифицированных подпрограмм и типов данных в определении подпрограммы.

Примера

```
PUT ROUTINE FROM procs/proc1.sar;
```

Замечания по использованию

В заданной схеме можно одновременно установить не более одной процедуры.

RECONCILE

В описание этой команды надо добавить следующее замечание по использованию:

При согласовании, если нет никаких других несоответствий, делаются попытки связать файлы, существующие в соответствии с данными таблиц, но несуществующие в соответствии с данными менеджера связей данных. Требуемый менеджер связей данных DB2 - тот, на который ссылается значение DATALINK в таблице.

Согласование допускает недоступность требуемого менеджера связей данных DB2, равно как и тех, которые сконфигурированы в базе данных, но не упоминаются в данных таблицы.

Если не задана таблица исключений, в файле отчета исключений (*имя-файла.exr*) будет имя хоста, имя файла, ID столбца и код причины для всех значений из столбца DATALINK, для которых не удастся восстановить ссылки на файлы. Если ссылку на файл не удастся восстановить из-за того, что менеджер связей данных DB2 сам был отброшен из этой базы данных при помощи команды DROP DATALINKS MANAGER, в файл отчета исключений записывается не полное имя файла; в нем отсутствует префикс. В таблице исключений для значений DATALINK, с отброшенным или недоступным менеджером связей данных DB2 имя файла в значении DATALINK не будет полным именем файла. В нем отсутствует префикс. Например, если исходное значение DATALINK было 'http://host.com/dlfs/x/y/a.b', в таблицу исключений будет записано значение 'http://host.com/x/y/a.b', то есть префикс '/dlfs' будет исключен. В файл отчета исключений в этом случае будет записано 'x/y/a.b', то есть префикс 'dlfs' будет исключен.

В конце процесса согласования таблица выводится из состояния отложенного согласования связей данных (DRP), только если согласование завершилось на всех требуемых менеджерах связей данных DB2. Если для какого-нибудь из требуемых менеджеров связей данных DB2 согласование отложено (поскольку менеджеры были недоступны), таблица останется в состоянии DRP или будет переведена в это состояние.

В список возможных нарушений надо добавить следующее:

00010-Менеджер связей данных DB2, на который ссылается значение DATALINK, отброшен из базы данных при помощи команды DROP DATALINKS MANAGER. В этом случае в соответствующем значении DATALINK в таблице исключений не будет содержаться префикс. Например, если исходное значение DATALINK было 'http://host.com/dlfs/prfx/x/y/a.b', а '/dlfs/prfx' - префикс, в таблицу исключений будет записано 'http://host.com/x/y/a.b'.

REORGANIZE TABLE

В Замечания по использованию надо добавить следующее предложение:

REORGANIZE TABLE не может использовать индекс на основе расширения индекса.

RESTORE DATABASE

Синтаксис

В синтаксис команды RESTORE DATABASE после опций TABLESPACE/TABLESPACE ONLINE/HISTORY FILE добавлена следующая опция:



Надо добавить описания этих параметров:

INCREMENTAL

Вызывает инкрементное (кумулятивное) восстановление базы данных вручную. Все команды восстановления будет вводить пользователь.

В описание параметра ABORT добавлено следующее:

Этот параметр можно также использовать для остановки инкрементного восстановления прежде его завершения.

Кроме того, добавлены следующие примеры:

Ниже приведен пример стратегии инкрементного резервного копирования восстанавливаемой базы данных в течение недели. Один раз в неделю выполняется полное резервное копирование, каждый день - разностное, в середине недели выполняется инкрементное резервное копирование:

```
(Sun) backup db kdr use adsm
(Mon) backup db kdr online incremental delta use adsm
(Tue) backup db kdr online incremental delta use adsm
(Wed) backup db kdr online incremental use adsm
(Thu) backup db kdr online incremental delta use adsm
(Fri) backup db kdr online incremental delta use adsm
(Sat) backup db kdr online incremental use adsm
```

Для восстановления базы данных вручную из созданных выше резервных копий в пятницу утром введите следующие команды:

```
restore db kdr incremental taken at (Thu)
restore db kdr incremental taken at (Sun)
restore db kdr incremental taken at (Wed)
restore db kdr incremental taken at (Thu)
```

Особенности менеджера связей данных DB2

Второй абзац раздела "DB2 Data Links Manager Considerations" (Особенности менеджера связей данных DB2) нужно заменить на:

Если для базы данных сконфигурирован один или несколько серверов связей данных, операция восстановления завершится успешно, даже если ни один сервер связей данных не доступен. После перезапуска сервера связей данных до того, как он станет доступным базе данных, на нем будет завершено восстановление.
ПРИМЕЧАНИЕ: Если операция восстановления базы данных ожидает недоступный сервер связей данных, ни одна из последующих операций восстановления базы данных или табличного пространства не будет успешной.

ROLLFORWARD DATABASE

Второй абзац раздела "DB2 Data Links Manager Considerations" (Особенности менеджера связей данных DB2) нужно заменить на:

Если для базы данных сконфигурирован один или несколько серверов связей данных, операция повтора транзакций завершится успешно, даже если ни один сервер связей данных не доступен. После перезапуска сервера связей данных до того, как он станет доступным базе данных, на нем будет завершен повтор транзакций.

UPDATE HISTORY FILE

В Замечаниях по использованию надо удалить следующие предложения:

The history file is used for record keeping purposes only.
It is not used during database recovery.

Надо добавить следующее предложение:

Файл хронологии используется администраторами базы данных для хранения записей. Он используется внутри DB2 при автоматическом восстановлении из инкрементных резервных копий.

Ошибка документации в разделе CLP Return Codes (Коды возврата процессора командной строки)

Второй абзац раздела Command Line Processor Return Codes Главы 2 должен выглядеть следующим образом:

Например, следующий сценарий оболочки Bourne выполняет команду GET DATABASE MANAGER CONFIGURATION, а затем проверяет код возврата процессора командной строки:

```
db2 get database manager configuration
if [ "$?" = "0" ]
then echo "OK!"
fi
```

Глава 10. Data Movement Utilities Guide and Reference

Состояния отложенных действий после операции загрузки

Первые два предложения в последнем абзаце в этом разделе изменены на следующие:

Четвертое возможное состояние, связанное с процессом загрузки (состояние отложенной проверки), относится к реляционным и проверочным ограничениям, ограничениям DATALINKS, ограничениям AST или ограничениям генерируемых столбцов. Например, если существующая таблица - это родительская таблица, содержащая первичный ключ, на который ссылается внешний ключ зависимой таблицы, изменение данных в родительской таблице переводит обе таблицы (но не табличное пространство) в состояние отложенной проверки.

Ограничения загрузки

Для генерируемых столбцов и утилиты загрузки применяются следующие ограничения:

- В таблицу, содержащую в уникальном индексе генерируемый столбец, можно загрузить данные, только если этот генерируемый столбец является включенным столбцом индекса или если используется модификатор типа файла `generatedoverride`. Если используется этот модификатор, ожидается, что все значения для этого столбца будут получены из входного файла данных.
- В таблицу, содержащую генерируемый столбец в ключе разделения, можно загрузить данные, только если используется модификатор типа файла `generatedoverride`. Если используется этот модификатор, ожидается, что все значения для этого столбца будут получены из входного файла данных.

Для запуска автозагрузки при значении **Authentication YES** необходим **hexecd**

В разделе Опции автозагрузки к описанию параметров **AUTHENTICATION** и **PASSWORD** надо добавить следующее замечание:

В среде Linux, если вы запускаете автозагрузку при значении опции аутентификации **YES**, на всех компьютерах необходимо разрешить **hexecd**. Если этого не сделать, будет сгенерировано следующее сообщение об ошибке:

```
openbreeze.torolab.ibm.com: Соединение отклонено
SQL6554N Ошибка при попытке удаленного выполнения процесса.
```

В файл `db2diag.log` будут записаны следующие сообщения об ошибках:

```
2000-10-11-13.04.16.832852 Instance:svtdbm Node:000
PID:19612(db2at1d) Appid:
oper_system_services sqloRemoteExec Probe:31
```

Модификатор типа файла **totalfreespace**

Модификатор типа файла `totalfreespace` (**LOAD**) изменен; теперь он допускает значения от 0 до 2 147 483 647.

Использование импорта с буферизованными вставками

Примечание в конце этого раздела должно выглядеть так:

| **Примечание:** Во всех средах, кроме EEE, возможность буферизованных вставок
| отключается при операциях импорта, для которых задан параметр
| INSERT_UPDATE.

Глава 11. Replication Guide and Reference

Репликация в Windows 2000

DB2 DataPropagator Версии 7.1 совместим с операционной системой Windows 2000.

Имена столбцов и таблиц

При репликации пробелы в именах столбцов и таблиц не поддерживаются.

Репликация столбцов DATALINK

Репликация столбцов DATALINK в Solaris возможно, начиная с Версии 7.1 FixPak 1. Для нее требуется, чтобы демон FTP работал на файловых системах источника и назначения DATALINK и поддерживал команду MDTM (modtime), которая выводит время последней модификации данного файла. Если вы используете Версию 2.6 операционной системы Solaris, или любую другую версию, где FTP не поддерживает MDTM, вам понадобится дополнительное программное обеспечение, например, WU-FTPD.

Нельзя выполнить репликацию столбцов DATALINK между базами данных DB2 в AS/400 и базами данных DB2 на других платформах.

На платформе AS/400 не поддерживается репликация атрибута "comment" значений DATALINK.

При работе в AIX 4.2 необходимо перед выполнением программы обработчика пользователя по умолчанию (ASNDLCOPY) установить PTF для APAR IY03101 (AIX 4210-06 RECOMMENDED MAINTENANCE FOR AIX 4.2.1). Этот PTF содержит исправление "проблемы 2000 года" для команды "modtime/MDTM" демона FTP. Чтобы проверить это исправление, посмотрите какое время последнего изменения возвращает команда "modtime <файл>" для файла, измененного после 1 января 2000 года.

Если таблица назначения - это внешняя таблица CCD, DataPropagator DB2 вызывает для репликации файлов DATALINK процедуру ASNDLCOPY. Самую свежую информацию о том, как использовать программы ASNDLCOPY и ASNDLCOPYD, смотрите во вводной части исходного текста этих программ. Применяются следующие ограничения:

- Внешние таблицы CCD могут содержать индикаторы DATALINK, но не значения DATALINK.
- Сжатые внешние таблицы CCD могут содержать значения DATALINK.
- Несжатые таблицы назначения CCD не могут содержать столбцов DATALINK.
- Если сервер источника совпадает с сервером назначения, в наборе регистрации не должно быть членов со столбцами DATALINK.

Ограничения на большие объекты

Сжатые внутренние таблицы CCD не могут содержать ссылок на столбцы больших объектов или индикаторы больших объектов.

Репликация и серверы других типов (не IBM)

Для репликации данных на и с серверов других типов (не IBM), таких как Informix, Microsoft SQL Server, Oracle, Sybase и Sybase SQL Anywhere, необходимо использовать DataJoiner Версии 2 или более поздней. Для этого типа репликации нельзя использовать функцию Relational Connect, поскольку DB2 Relational Connect Версии 7.1 не имеет средств обновления. Кроме того, для управления такой разнородной репликацией на всех платформах (AS/400, OS/2, OS/390, UNIX и Windows) для всех существующих версий DB2 и DataJoiner необходимо использовать DJRA (администратор репликации DataJoiner).

Предварительное требование для повсеместного изменения

Если нужно настроить репликацию типа повсеместного изменения с определением конфликтов и с набором регистрации, содержащим более 150 членов, нужно выполнить следующий оператор DDL для создания таблицы ASN.IBMSNAP_COMPENSATE на сервере управления:

```
CREATE TABLE ASN.IBMSNAP_COMPENSATE (  
    APPLY_QUAL char(18) NOT NULL,  
    MEMBER SMALLINT,  
    INTENTSEQ CHAR(10) FOR BIT DATA,  
    OPERATION CHAR(1));
```

Сценарии репликации

Посмотрите на странице Library Web-сайта DataPropagator (<http://www.ibm.com/software/data/dpropr/>) новый сценарий репликации разнородных данных. Выполните шаги этого сценария по копированию изменений из таблицы источника репликации базы данных Oracle для AIX в таблицу назначения базы данных DB2 для Windows NT. В этом сценарии используется инструмент DB2 DataJoiner Replication Administration (DJRA), триггеры Capture, программа Apply и DB2 DataJoiner.

В этой книге на странице 44 инструкции в Шаге 6 по созданию файла паролей следует читать так:

Шаг 6: Создание файла паролей

Так как для соединения с сервером источника необходима программа Apply, для аутентификации пользователей нужно создать файл паролей. Убедитесь, что этот файл паролей будет доступен для чтения ID пользователя, который будет запускать программу Apply.

Чтобы создать файл паролей:

1. Из окна командной строки Windows NT перейдите в каталог C:\scripts.
2. Создайте в этом каталоге новый файл с именем DEPTQUAL.PWD. Этот файл можно создать, используя любой текстовый редактор, например Notepad. Для имени файла паролей действует соглашение об именах: *стцфapply.pwd*, где *стцфapply* - регистрозависимая строка символов, которая по регистру и значению *должна* совпадать со спецификатором Apply, использовавшимся при создании набора регистрации. В данном сценарии спецификатор Apply - DEPTQUAL.

Примечание: Кроме того, поддерживается соглашение об именах из DB2 DataPropagator, Версии 5.

3. Содержание файла паролей записывается в следующем формате:

SERVER=сервер USER=id_пользователя PWD=пароль

Где:

сервер

Имя сервера источника, назначения или управления, в точности совпадающее с именем в таблице набора регистрации. В этом сценарии используются имена SAMPLE и COPYDB.

id_пользователя

ID пользователя, который планируется использовать при управлении конкретной базой данных. В операционных системах Windows NT и UNIX это значение регистрозависимо.

пароль

Пароль, соответствующий данному ID пользователя. В операционных системах Windows NT и UNIX это значение регистрозависимо.

Не помещайте в этот файл пустых строк или строк комментариев. Добавьте только информацию об имени сервера, ID пользователя и пароле.

4. Содержание файла паролей должно выглядеть примерно так:

```
SERVER=SAMPLE USER=subina PWD=subpw  
SERVER=COPYDB USER=subina PWD=subpw
```

Дополнительную информацию об аутентификации и защите в DB2 смотрите в руководстве *IBM DB2 Administration Guide*.

Планирование репликации

В раздел "Connectivity" на страницу 65 нужно добавить следующую информацию: Если программа Apply не может соединиться с сервером управления, она прекращает работу.

При использовании блокировки данных для AS/400 общий объем данных для репликации за интервал не должен превышать 4 миллионов строк (а не "4 Мбайт", как сказано на странице 69 этой книги).

Настройка среды репликации

В разделе "Customizing CD table, index, and tablespace names" на странице 95 сказано, что файл DPREPL.DFT находится или в каталоге \sqllib\bin, или в каталоге \sqllib\java. Это неверно, на самом деле файл DPREPL.DFT находится в каталоге \sqllib\cc.

Диагностика ошибок

Анализатор репликации работает в 32-битных системах Windows и в AIX. Для работы анализатора в AIX каталог sqllib/bin должен быть задан в переменной среды PATH до каталога /usr/local/bin, чтобы избежать конфликта с /usr/local/bin/analyze.

Для анализатора репликации можно использовать два дополнительных необязательных ключевых слова: ST и AT.

ST=*n* Показывать только те записи из таблицы трассировки захвата, которые не старше *n* дней. Это необязательное ключевое слово. Если это ключевое слово не задано, по умолчанию используется значение 7 дней.

AT=*n* Показывать только те записи из таблицы журнала Apply, которые не старше

n дней. Это необязательное ключевое слово. Если это ключевое слово не задано, по умолчанию используется значение 7 дней.

Пример:

```
analyze mydb1 mydb2 f=mydirectory ct=4 at=2 deepcheck q=applyqual1
```

Изменена следующая информация о ключевых словах анализатора репликации:

deepcheck

Задаёт, что анализатор должен выполнить более сложный анализ, включая следующую информацию: информацию о сокращении таблиц CD и UOW, подробности разделения табличных пространств и сжатия DB2 for OS/390, анализ индексов назначения для ключей регистрации, временную последовательность регистрации и ошибки операторов SQL для набора регистрации. Этот анализ проводится на всех серверах. Это необязательное ключевое слово.

lightcheck

Задаёт, что из отчёта должна быть исключена следующая информация: вся подробная информация о столбцах, полученная из таблицы ASN.IBMSNAP_SUBS_COLS, информация об ошибках, неправильностях и пропусках регистрации, а также информация о неправильных или неэффективных индексах. При этом экономятся ресурсы и получается выходной файл HTML меньшего размера. Это необязательное ключевое слово; его нельзя использовать вместе с ключевым словом deepcheck.

Средства анализатора для репликации на платформах AS/400 доступны в PTF. Эти средства собирают информацию о среде репликации и создают файл HTML, который можно послать в сервисный центр фирмы IBM для помощи в диагностике ошибок. Чтобы получить эти средства для AS/400, загрузите соответствующий PTF (например, для продукта 5769DP2 нужно загрузить PTF SF61798 или его более позднюю замену).

В раздел "Troubleshooting" добавьте следующее описание ошибки и её решения:

Ошибка: Программа Apply закикливается, не выполняя изменений репликации; таблица журнала Apply содержит значение STATUS=2.

Набор регистрации включает несколько исходных таблиц. Для улучшения обработки активных областей для одной исходной таблицы из этого набора определена внутренняя таблица CCD, но она определена в другом наборе регистрации. Изменения были внесены в исходную таблицу, но процесс Apply, заполняющий внутреннюю таблицу CCD, выполняется асинхронно (например, программа Apply могла быть не запущена или событие не произошло и т.п.). Программа Apply, выполняющая репликацию изменений из исходной таблицы в таблицу назначения, закикливается, так как она ожидает обновления внутренней таблицы CCD.

Чтобы остановить закикливание, запустите программу Apply (или иницируйте событие, запускающее репликацию) для внутренней таблицы CCD. Эта программа Apply заполнит внутреннюю таблицу CCD и позволит закиклившейся программе Apply обработать изменения из всех исходных таблиц.

Подобная ситуация может возникнуть для набора регистрации, содержащего исходные таблицы с внутренними таблицами CCD, заполняемыми несколькими программами Apply.

Программы Capture и Apply для AS/400

Замечание "A note on work management" на странице 178 должно выглядеть так:

Можно изменить определения по умолчанию или задать собственные определения. При создании собственного описания подсистемы нужно задать для нее имя QZSNDRP и создать ее в библиотеке. Дополнительную информацию об изменении этих определений смотрите в руководстве "OS/400 Work Management".

В раздел "Verifying and customizing your installation of DB2 DataPropagator for AS/400" на странице добавьте следующее:

Если большой объем транзакций вызывает проблемы с конфликтами блокировок, можно увеличить значение срока ожидания. Можно изменять задание каждый раз при запуске задания Capture или же использовать следующую процедуру для изменения значения срока ожидания по умолчанию для всех заданий, выполняющихся в подсистеме:

1. Выполните следующую команду, чтобы создать новый объект класса, скопировав QGPL/QBATCH:

```
CRTDUPOBJ OBJ(QBATCH) FROMLIB(QGPL) OBJTYPE(*CLS) TOLIB(QDPR) NEWOBJ(QZSNDPR)
```

2. Измените значения срока ожидания для вновь созданного класса (например, на 300):

```
CHGCLS CLS(QDPR/QZSNDPR) DFTWAIT(300)
```

3. Измените запись маршрутизации в описании подсистемы QDPR/QZSNDPR, чтобы использовался этот вновь созданный класс:

```
CHGRTE SBSD(QDPR/QZSNDPR) SEQNBR(9999) CLS(QDPR/QZSNDPR)
```

На странице 195 параметры команды ADDEXITPGM должны выглядеть так:

```
ADDEXITPGM EXITPNT(QIBM_QJO_DLT_JRNRCV)
            FORMAT(DRCV0100)
            PGM(QDPR/QZSNDREP)
            PGMNBR(*LOW)
            CRTEXITPNT(*NO)
            PGMFTA(65535 10 QSYS)
```

Структуры таблиц

На странице 339 добавьте следующее предложение в конец описания столбца STATUS для значения "2":

Если используются внутренние таблицы CCD и столбец STATUS таблицы журнала Apply часто содержит значение "2", посмотрите в главе 8 "Problem Determination" раздел "Problem: The Apply program loops without replicating changes, the Apply trail table shows STATUS=2".

Сообщения программ Capture и Apply

Надо добавить сообщение ASN1027S:

```
ASN1027S
```

Задано слишком много столбцов больших объектов (LOB).

Код ошибки: "<код_ошибки>".

Объяснение: Для члена набора регистрации задано слишком много столбцов больших объектов (BLOB, CLOB).

Максимальное разрешенное число столбцов - 10.

Действия пользователя: Удалите избыточные столбцы больших объектов из этого члена набора регистраций.

Сообщение ASN1048E должно выглядеть так:

```
ASN1048E
```

Ошибка выполнения цикла Apply.

Подробную информацию смотрите в таблице журнала Apply:

```
"<текст>"
```

Объяснение: Ошибка выполнения цикла Apply. В этом сообщении "<текст>"

указывает "<сервер_назначения>", "<владелец_назначения, таблица_назначения, номер_stmt>" и "

Действия пользователя: Проверьте поля APPERRM в таблице журнала аудита, чтобы определить причину с

Запуск программ Capture и Apply из прикладной программы

На странице 399 этой книги в комментариях к примеру процедуры, запускающей программы Capture и Apply, есть несколько ошибок; однако код этой процедуры правилен. Последняя часть этого примера процедуры относится к параметрам Apply, несмотря на то, что в комментарии сказано, что она относится к параметрам Capture.

Примеры API для вызова программ Apply и Capture и соответствующие им make-файлы можно найти в следующих каталогах:

Для NT - sql11b\samples\rep1
Для UNIX - sql11b/samples/rep1

Глава 12. System Monitor Guide and Reference

db2ConvMonStream

В разделе Usage Notes структура для типа потока данных переменной снимка SQLM_ELM_SUBSECTION должна быть sqlm_subsection.

Глава 13. Troubleshooting Guide

Запуск DB2 в Windows 95, Windows 98 и Windows ME, когда пользователь не зарегистрирован в системе

Для успешного выполнения команды `db2start` в среде Windows 95, Windows 98 или Windows Millennium Edition (ME) необходимо:

- Зарегистрироваться в системе с помощью окна регистрации Windows или окна регистрации Microsoft Networking
- Или же выполнить команду `db2logon` (информацию о команде `db2logon` смотрите в примечании 1 на стр. 138).

Кроме этого, ID пользователя, заданный при регистрации или для команды `db2logon` должен удовлетворять требованиям DB2 (смотрите примечание 2 на стр. 139).

При запуске команды `db2start` она сначала проверяет, нет ли в системе зарегистрированного пользователя. Если такой пользователь есть, команда `db2start` использует его ID пользователя. Если зарегистрированных пользователей нет, команда `db2start` проверяет, не запущена ли команда `db2logon`, и если она запущена, команда `db2start` использует ID пользователя, заданный для команды `db2logon`. Если команда `db2start` не может найти правильный ID пользователя, она прекращает свою работу.

При установке DB2 Universal Database Версии 7 в Windows 95, Windows 98 и Windows ME программа установки по умолчанию добавляет в папку Автозагрузка ярлык, запускающий команду `db2start` при загрузке системы (дополнительную информацию смотрите в примечании 1 на стр. 138). Если ни один пользователь не был зарегистрирован в системе и не была запущена команда `db2logon`, команда `db2start` прекратит свою работу.

Для случая, когда в системе не были зарегистрированы пользователи, можно удовлетворить требование запуска команды `db2logon` перед командой `db2start`, выполняя команды из пакетного файла:

1. Создайте пакетный файл, содержащий команду `db2logon` и после нее команду `db2start.exe`. Например:

```
@echo off
db2logon db2local /p:password
db2start
cls
exit
```

2. Назовите этот пакетный файл `db2start.bat` и сохраните его в каталоге `/bin` на диске и в каталоге, где установлена система DB2. Пакетный файл сохраняется в этом каталоге, чтобы операционная система могла найти его.

Диск и каталог установки DB2 записан в переменной реестра DB2 DB2PATH.

Чтобы узнать диск и каталог установки DB2, используйте следующую команду:

```
db2set -g db2path
```

Предположим, команда эта `db2set` вернула значение `c:\sql1ib`. В этом случае сохраните пакетный файл как файл:

```
c:\sql1ib\bin\db2start.bat
```

3. Для запуска DB2 при загрузке системы нужно запускать этот пакетный файл с помощью ярлыка в папке Автозагрузка. Есть две возможности:
- Изменить ярлык, созданный программой установки DB2, чтобы вместо файла **db2start.exe** он запускал этот пакетный файл. Для предыдущего примера этот ярлык должен запускать пакетный файл **db2start.bat**. Ярлык, созданный программой установки DB2, имеет имя **DB2 - DB2.lnk** и в большинстве систем находится по адресу `c:\WINDOWS\Главное меню\Программы\Автозагрузка\DB2 - DB2.lnk` (`c:\WINDOWS\Start Menu\Programs\Start\DB2 - DB2.lnk`).
 - Добавить свой собственный ярлык для запуска этого пакетного файла и удалить ярлык, добавленный программой установки DB2. Для удаления ярлыка DB2 используйте следующую команду:

```
del "C:\WINDOWS\Главное меню\Программы\Автозагрузка\DB2 - DB2.lnk"
```

Если вы решили использовать свой собственный ярлык, для него нужно включить атрибут *Закрывать окно по завершении сеанса работы*. Если этот атрибут не выключен, даже после успешного завершения команды **db2start** в полосе задач останется сеанс командной строки DOS. Чтобы окно DOS не открывалось при выполнении **db2start**, можно при создании этого ярлыка задать для него (и окна DOS, в котором выполняется пакетный файл) режим выполнения в свернутом окне.

Примечание: Вместо запуска DB2 во время загрузки системы можно запускать DB2 перед выполнением прикладных программ, использующих DB2. Подробную информацию смотрите в примечании 5 на стр. 139.

Если для выполнения команды **db2logon** перед командой **db2start** используется пакетный файл и иногда в этот момент в системе зарегистрированы пользователи, команда **db2start** будет продолжать работать; единственное отличие в том, что DB2 будет использовать ID зарегистрированного пользователя. Более подробную информацию смотрите в примечании 1.

Примечания:

1. Команда **db2logon** имитирует регистрацию пользователя. Формат команды **db2logon**:

```
db2logon ID_пользователя /р:пароль
```

Заданный в этой команде ID пользователя должен удовлетворять требованиям к именам DB2 (дополнительную информацию смотрите в примечании 2 на стр. 139). Если в этой команде не заданы ID пользователя и пароль, откроется окно, запрашивающее пользователя ввести ID пользователя и пароль. Если задан только ID пользователя, пароль у пользователя не запрашивается; в некоторых случаях, описанных ниже, пароль обязателен.

Заданные командой **db2logon** ID пользователя и пароль используются, только если в системе не зарегистрировался ни один пользователь (с помощью окна регистрации Windows или окна регистрации Microsoft Networking). Если зарегистрирован пользователь и была запущена команда **db2logon**, для всех действий DB2 используется ID пользователя, заданный в команде **db2logon**, но заданный в команде **db2logon** пароль игнорируется.

Если нет пользователей, зарегистрированных с помощью окна регистрации Windows или окна регистрации Microsoft Networking, заданные в команде **db2logon** ID пользователя и пароль используются следующим образом:

- Команда **db2start** использует этот ID пользователя при своем запуске (пароль ей не требуется).
- При отсутствии спецификатора высокого уровня для таких операций, как создание таблицы, в качестве спецификатора высокого уровня используется этот ID пользователя. Например:
 - a. Если введена команда: `db2logon db2local`
 - b. И затем введена команда: `create table tabl`
Эта таблица будет создана со спецификатором высокого уровня как `db2local.tabl`.

Используйте ID пользователя, совпадающий с именем схемы для таблиц и других объектов.

- Если в системе клиента с типом аутентификации сервер пользователь вводит оператор `CONNECT` без ID пользователя и пароля (например, `CONNECT TO TEST`), для проверки пользователя на удаленном сервере используются ID пользователя и пароль, заданные в команде **db2logon**. Если пользователь явно задает ID пользователя и пароль для соединения (например, `CONNECT TO TEST USER ID_пользователя USING пароль`), используются значения, заданные в операторе `CONNECT`.
2. В Версии 7 ID пользователя, использованный для регистрации или заданный в команде **db2logon** должен удовлетворять следующим требованиям DB2:
 - Он не должен совпадать с один из следующих имен: `USERS`, `ADMINS`, `GUESTS`, `PUBLIC`, `LOCAL` или с каким-либо из зарезервированных слов SQL (которые перечислены в руководстве *SQL Reference*).
 - Он не может начинаться с `SQL`, `SYS` или `IBM`
 - Могут использоваться следующие символы:
 - От A до Z (в Windows 95, Windows 98 и Windows ME поддерживаются регистрозависимые ID пользователей)
 - От 0 до 9
 - `@`, `#` и `$`
 3. Создание ярлыка **db2start** в папке Автозагрузка можно предотвратить, используя режим интерактивной выборочной установки или используя для установки файл ответов, в котором задана опция `DB2.AUTOSTART=NO`. В этих случаях в папке Автозагрузка не будет создан ярлык `db2start` и будет нужно создать свой собственный ярлык для запуска файла `db2start.bat`.
 4. В **Windows 98** и **Windows ME** можно задать ID пользователя, который будет автоматически регистрироваться при запуске Windows 98 или Windows ME. В этом случае окно регистрации Windows не будет появляться. Если используется эта возможность, пользователь будет зарегистрирован и команда **db2start** будет успешно выполнена, если этот ID пользователя удовлетворяет требованиям DB2 (подробную информацию смотрите в примечании 2). Если эта возможность не используется, пользователю всегда будет предлагаться окно регистрации. Если пользователь закроет это окно без регистрации, команда **db2start** не будет выполнена успешно, если перед ней не будет выполнена команда **db2logon** или если она не будет запущена из пакетного файла, как описано выше.
 5. Если система DB2 не запущена во время загрузки системы, система DB2 может быть запущена прикладной программой. Файл `db2start.bat` можно выполнить в процессе инициализации прикладных программ, использующих DB2. При использовании этого метода система DB2 будет запускаться только при запуске прикладной программы, которая будет использовать DB2. После завершения работы этой прикладной программы можно использовать команду **db2stop** для

остановки DB2. Прикладные программы могут запускать DB2 таким способом, если DB2 не запущена при загрузке системы.

Для использования прикладной программы Synchronizer DB2 или вызовов функций API синхронизации из прикладной программы должна быть запущена DB2, если загруженные для выполнения сценарии содержат команды, обращающиеся к локальному экземпляру или локальной базе данных. Эти команды могут находиться в сценариях базы данных, сценариях экземпляра или сценариях операционной системы (OS). Если сценарий OS не содержит команд процессора командной строки или вызовов функций API DB2, использующих экземпляр или базу данных, этот сценарий можно выполнять без запуска DB2. Поскольку может быть сложно заранее определить, будут ли такие команды выполняться в сценариях во время процесса синхронизации, лучше запустить DB2 перед началом процесса синхронизации.

Если из прикладной программы вызывается команда **db2sync** или функции API синхронизации, при инициализации этой прикладной программы следует запустить DB2. Если для запуска синхронизации пользователи будут использовать ярлык Synchronizer DB2 из папки DB2 для Windows, этот ярлык синхронизации DB2 нужно изменить, чтобы он вызывал файл `db2sync.bat`. Этот файл должен содержать следующие команды, обеспечивающие запуск DB2 перед началом синхронизации:

```
@echo off
db2start.bat
db2sync.exe
db2stop.exe
cls
exit
```

В этом примере подразумевается, что файл `db2start.bat` запускает команды **db2logon** и **db2start**, как описано выше.

Если вы решили запускать DB2 при запуске прикладной программы, убедитесь, что программа установки DB2 не добавила в папку Автозагрузка ярлык для запуска DB2. Подробную информацию смотрите в примечании 3 на стр. 139.

Глава 14. Использование DB2 Universal Database на 64-битных платформах

Глава 5. Configuration

shmsys:shminfo_shmmax

Пользователям DB2 в 64-битной операционной системе Solaris следует при необходимости увеличить значение "shmsys:shminfo_shmmax" в /etc/system, чтобы иметь возможность выделять большую совместную память базы данных. В книге *DB2 for UNIX Быстрый старт* рекомендуется задать значение этого параметра "90% физической оперативной памяти компьютера в байтах". Эта рекомендация подходит и для 64-битных реализаций.

Однако есть сложности со следующей рекомендацией в книге *DB2 for UNIX Быстрый старт*: Для 32-битных систем с более чем 4 Гбайтами оперативной памяти (в операционной системе Solaris возможно в общей сложности до 64 Гбайт), если пользователь задает значение shmmax, превышающее 4 Гбайта, и использует 32-битное ядро, ядро воспринимает только первые 32 бита от всего числа, что может привести к очень низкому значению shmmax.

DBHEAP

Последнее предложение в главе 5 "Configuration":

```
If the DBHEAP value is greater than 64 KB, the cast results in a wrapped value...
```

следует изменить на:

```
Если значение DBHEAP превышает 65535, преобразование типов приводит к wrapped value...
```

LOCKLIST

К Таблице 2 надо добавить следующую информацию.

Параметр	Прежний верхний предел	Текущий верхний предел
LOCKLIST	60000	524288

Глава 6. Restrictions

В 64-битных операционных системах в настоящее время отсутствует поддержка LDAP.

32-битные и 64-битные базы данных нельзя создать по одному и тому же пути. Например, если 32-битная база данных существует по адресу <некоторый_путь>, команда:

```
db2 create db <некоторая_бд> on <некоторый_путь>
```

от 64-битного экземпляра приведет к ошибке с сообщением "SQL10004C Ошибка ввода-вывода при обращении к каталогу базы данных."

Глава 15. XML Extender Administration and Programming

Замечания по выпуску (Release Notes) для IBM DB2 XML Extender можно найти на Web-сайте DB2 XML по адресу: <http://www-4.ibm.com/software/data/db2/extenders/xmlxt/library.html>

Часть 5. Инструменты управления

Глава 16. Центр управления	147
Возможность управления серверами DB2 Server for VSE and VM	147
Поддержка Java 1.2 для Центра управления.	147
Ошибка "Неверный ярлык" при использовании электронной справки в операционной системе Windows	148
Центр управления в виде Java в OS/2	148
Ошибка "Доступ к файлу запрещен" при попытке просмотреть завершенное задание в журнале операционной системы Windows	148
Соединение для проверки многоузлового изменения	148
Центр управления для DB2 for OS/390	148
Необходимое исправление для Центра управления для OS/390	150
Изменения в окне Создать географический слой	150
Информация по устранению неисправностей для Центра управления DB2	150
Устранение неисправностей для Центра управления в системах на основе UNIX	150
Возможные проблемы со всплывающими справками в OS/2	150
Запуск нескольких апплетов Центра управления	151
Справка по параметру конфигурации jdk11_path	151
Системная ошибка Solaris (SQL10012N) при использовании Центра сценариев или журнала	151
Справка по файлу DPREPL.DFT	151
Электронная справка по Центру управления, выполняемому как апплет	151
Выполнение Центра управления в режиме апплета (Windows 95).	152
Центр управления DB2 for OS/390	152
Глава 17. Информационный центр	153
Ошибка "Неверный ярлык" в операционной системе Windows	153
Открытие внешних ссылок Web в Netscape Navigator, если Netscape уже открыт (системы на основе UNIX).	153
Проблемы запуска Информационного центра	153
Глава 18. Мастера	155
Задание размера экстенда в мастере по созданию базы данных	155

Глава 16. Центр управления

Возможность управления серверами DB2 Server for VSE and VM

В Центре управления DB2 Universal Database Версии 7.1 улучшена поддержка баз данных DB2 Server for VSE and VM. С помощью Центра управления можно просмотреть все объекты баз данных DB2 Server for VSE and VM. Поддерживаются также операторы CREATE INDEX, REORGANIZE INDEX и UPDATE STATISTICS и команда REBIND. Для REORGANIZE INDEX и REBIND требуется выполнение хранимой процедуры на хостах DB2 Server for VSE and VM. Эта хранимая процедура поставляется средством Центра управления для VSE и VM DB2 для Server for VSE and VM.

Полностью интегрированный Центр управления позволяет пользователю управлять DB2 независимо от того, на какой платформе работает сервер DB2. Объекты DB2 Server for VSE and VM выводятся в главном окне Центра управления вместе с объектами DB2 Universal Database. При выборе объекта вызываются соответствующие действия и утилиты для управления этим объектом. Например, пользователь может вывести список индексов конкретной базы данных, выбрать один из этих индексов и реорганизовать его. Пользователь может также вывести список таблиц базы данных и выполнить сбор статистической информации или определить одну из таблиц в качестве источника репликации.

Информацию о конфигурировании Центра управления для выполнения задач администратора для объектов DB2 Server for VSE and VM смотрите в руководстве *DB2 Connect. Руководство пользователя* или *Дополнение по установке и конфигурированию*.

Поддержка Java 1.2 для Центра управления

Центр управления поддерживает языки с двумя направлениями письма, такие как арабский или иврит, используя средства поддержки таких языков в Java 1.2. Эта поддержка обеспечивается только для платформ Windows NT.

Необходимо установить Java 1.2, чтобы Центр управления мог распознать и использовать этот продукт:

1. JDK 1.2.2 находится на компакт-диске DB2 UDB в каталоге DB2\bid\NT.
ibm-inst-n122p-win32-x86.exe - это программа установки, а
ibm-jdk-n122p-win32-x86.exe - дистрибутив JDK. Скопируйте оба эти файла во временный каталог на жестком диске и затем запустите оттуда программу установки.
2. Установите этот продукт в каталог <DB2PATH>\java\Java12, где <DB2PATH> - путь установки DB2.
3. При установке JDK/JRE не выбирайте JDK/JRE в качестве системной VM.

Когда Java 1.2 успешно установлен, при обычном запуске Центра управления будет использоваться Java 1.2.

Чтобы прекратить использовать Java 1.2, можно или деинсталлировать JDK/JRE из каталога <DB2PATH>\java\Java12, или просто переименовать подкаталог <DB2PATH>\java\Java12, задав для него какое-либо другое имя.

Примечание: Не спутайте <DB2PATH>\java\Java12 с <DB2PATH>\Java12.<DB2PATH>\Java12 - это часть установки DB2, содержащая поддержку JDBC для Java 1.2.

Ошибка "Неверный ярлык" при использовании электронной справки в операционной системе Windows

При использовании электронной справки Центра управления может возникнуть ошибка "Неверный ярлык". Если недавно был установлен новый браузер Web (или новая версия браузера), убедитесь, что документы HTML и HTM связаны с правильным браузером. Посмотрите в Справке Windows раздел "Чтобы выбрать программу, открывающую файл".

Центр управления в виде Java в OS/2

Центр управления должен быть установлен на диске с файловой системой HPFS.

Ошибка "Доступ к файлу запрещен" при попытке просмотреть завершенное задание в журнале операционной системы Windows

В DB2 Universal Database for Windows NT при попытке открыть журнал для просмотра информации о задании, созданном в Центре сценариев возникает ошибка "Доступ к файлу запрещен". Состояние этого задания указывается как завершенное. Это происходит, если задание, созданное в Центре сценариев, содержит команду START. Чтобы избежать такой ошибки, используйте START/WAIT вместо START в пакетном файле и в самом задании.

Соединение для проверки многоузлового изменения

В Центре управления Версии 7.1 работа соединения для проверки многоузлового изменения зависит от версии экземпляра назначения. Для удаленной работы экземпляр назначения должен быть не ниже Версии 7.1. Для работы соединения для проверки многоузлового изменения в Версии 6 необходимо перенести Центр управления в систему экземпляра назначения и выполнять его оттуда.

Центр управления для DB2 for OS/390

Центр управления DB2 UDB for OS/390 позволяет управлять использованием лицензированных утилит DB2 IBM. Функции утилит, входящие в отдельно заказываемые средства DB2 UDB for OS/390, должны быть лицензированы и установлены в вашей среде, прежде чем ими можно будет управлять с помощью Центра управления DB2.

База данных "CC390", создаваемая для Центра управления при конфигурировании подсистемы DB2 for OS/390, используется для внутренней поддержки Центра управления. Не изменяйте эту базу данных.

Хотя DB2 for OS/390 Версии 7.1 специально не упомянута в таблице содержимого для Центра управления или в информации о задачах информационного центра, эта документация поддерживает функции DB2 for OS/390 Версии 7.1. В таблице содержимого не указаны версии для многих функций DB2 for OS/390, появляющихся только в Версии 6 и использующиеся также в Версии 7.1, и некоторых функций DB2 for OS/390, использующихся только в Версии 7.1. Сконфигурировав в Центре управления подсистему DB2 for OS/390 Версии 7.1, вы получите доступ ко всей документации по этой версии.

Чтобы из Центра управления для DB2 for OS/390 можно было использовать функцию DDL Generate, эта функция должна быть установлена:

- Для Версии 5 установите DB2Admin 2.0 с DB2 for OS/390 Версии 5.
- Для Версии 6 установите небольшое программное усовершенствование, которое будет доступно в виде PTF для средства Admin DB2 для DB2 for OS/390 Версии 6.
- Для Версии 7.1 функция DDL Generate входит в отдельно поставляемое средство Admin DB2 для DB2 for OS/390 Версии 7.1.

Из Центра управления можно обращаться к строителю хранимых процедур, но он должен быть уже установлен в момент запуска Центра управления DB2 UDB. Он входит в клиент разработки программ DB2.

Чтобы внести в каталог подсистему DB2 for OS/390 прямо на рабочей станции, используйте Ассистент конфигурирования клиента.

1. На странице Источник выберите радиокнопку **Конфигурировать соединение с базой данных вручную**.
2. На странице Протокол введите соответствующую информацию о связи.
3. На странице Подсистема задайте имя подсистемы в поле **Имя базы данных**.
4. На странице Опции узла включите переключатель **Сконфигурировать опции узла (необязательные)**.
5. Выберите MVS/ESA, OS/390 из списка в поле **Операционные системы**.
6. Нажмите кнопку **Завершить**, чтобы закончить конфигурирование.

Чтобы внести в каталог подсистему DB2 for OS/390 через компьютер шлюза, выполните описанные выше шаги с 1 по 6 на компьютере шлюза и затем:

1. Запустите Центр управления на компьютере клиента.
2. Щелкните правой кнопкой мыши по папке Системы и выберите **Добавить**.
3. В окне Добавить систему введите имя компьютера шлюза в поле **Имя системы**.
4. В поле **Удаленный экземпляр** введите DB2DAS00.
5. Для протокола TCP/IP в параметрах протокола в поле **Имя хоста** задайте имя хоста компьютера шлюза.
6. В поле **Имя службы** введите 523.
7. Нажмите кнопку **ОК**, чтобы добавить систему. Компьютер шлюза будет добавлен в папку **Системы**.
8. Раскройте объект этого компьютера шлюза.
9. Щелкните правой кнопкой мыши по папке Экземпляры и выберите **Добавить**.
10. В окне Добавить экземпляр нажмите кнопку **Обновить**, чтобы вывести список экземпляров, доступных на компьютере шлюза. Если компьютер шлюза - это система Windows NT, подсистема DB2 for OS/390 может быть каталогизирована под экземпляром DB2.
11. Выберите экземпляр. Параметры протокола будут автоматически заполнены для этого экземпляра.
12. Нажмите кнопку **ОК**, чтобы добавить экземпляр.
13. Откройте папку **Экземпляры**, чтобы увидеть только что добавленный экземпляр.
14. Раскройте объект этого экземпляра.
15. Щелкните правой кнопкой мыши по папке **Базы данных** и выберите **Добавить**.
16. Нажмите кнопку **Обновить**, чтобы вывести список локальных баз данных на компьютере шлюза. Если вы добавляете подсистему DB2 в окне Добавить базу данных, введите имя подсистемы в поле **Имя базы данных**. Необязательно: Введите локальный алиас для этой подсистемы (или базы данных).

17. Нажмите кнопку **ОК**.

Теперь эта подсистема успешно добавлена в Центр управления. Когда вы откроете эту базу данных, вы увидите подсистему DB2 for OS/390.

Необходимое исправление для Центра управления для OS/390

Необходимо применить APAR PQ36382 для средства 390 Enablement в DB2 for OS/390 Версии 5 и DB2 for OS/390 Версии 6, чтобы можно было управлять этими подсистемами с помощью Центра управления DB2 UDB для Версии 7.1. Без этого исправления нельзя использовать Центр управления DB2 UDB для Версии 7.1 для запуска утилит для этих подсистем.

Этот APAR нужно применить для следующих FMID:

390 Enablement в DB2 for OS/390 Версия 5: FMID JDB551D
390 Enablement в DB2 for OS/390 Версия 6: FMID JDB661D

Изменения в окне Создать географический слой

Из окна Создать географический слой удалены кнопки "<<" и ">>".

Информация по устранению неисправностей для Центра управления DB2

В главе "Установка и конфигурирование Центра управления" книги "Быстрый старт" сказано, что если возникают ошибки при запуске Центра управления в виде апплета, нужно в командном окне отменить задание браузера клиента в CLASSPATH. В этом разделе сказано также, что нужно запустить браузер из того же командного окна. Однако там не приведена команда запуска браузера. Для запуска Internet Explorer введите `start iexplore` и нажмите клавишу **Enter**. Для запуска Netscape введите `start netscape` и нажмите клавишу **Enter**. В этих командах предполагается, что путь браузера записан в PATH. Если это не так, добавьте путь браузера в PATH или перейдите в каталог установки браузера и повторите команду `start`.

Устранение неисправностей для Центра управления в системах на основе UNIX

Если не удастся запустить Центр управления в системе на основе UNIX, задайте переменную среды `JAVA_HOME`, чтобы она указывала путь установки Java:

- Если Java установлен в каталоге `/usr/jdk118`, задайте для `JAVA_HOME` значение `/usr/jdk118`.
- Для оболочек `sh`, `ksh` или `bash`:
`export JAVA_HOME=/usr/jdk118.`
- Для оболочек `csh` или `tcsh`:
`setenv JAVA_HOME /usr/jdk118`

Возможные проблемы со всплывающими справками в OS/2

Если Центр управления используется в OS/2 в режиме с размером экрана 1024x768 и 256 цветами и с включенной опцией "Знать палитру рабочего стола", всплывающие справки, выходящие за границы текущего окна, могут выводиться черным текстом на черном фоне. Чтобы исправить эту ошибку, или измените режим экрана на более чем 256 цветов, или отключите опцию "Знать палитру рабочего стола".

Запуск нескольких апплетов Центра управления

На одном компьютере можно одновременно запустить только один апплет Центра управления. Это ограничение применяется к апплетам Центра управления, выполняющимся во всех поддерживаемых браузерах.

Справка по параметру конфигурации `jdk11_path`

В справке Центра управления в описании параметра конфигурации `jdk11_path` (путь установки Java Development Kit 1.1) пропущена строка под заголовком **Применяется к**. Полный список под заголовком **Применяется к**:

- Сервер баз данных с локальными и удаленными клиентами
- Клиент
- Сервер баз данных с локальными клиентами
- Сервер многораздельных баз данных с локальными и удаленными клиентами
- Сервер спутниковых баз данных с локальными клиентами

Системная ошибка Solaris (SQL10012N) при использовании Центра сценариев или журнала

При выборе из Центра сценариев или из журнала системы Solaris может возникнуть следующая ошибка:

```
SQL10012N - Непредвиденная ошибка операционной системы
при загрузке библиотеки "/udbprod/db2as/sqllib/function/unfenced/
db2scdar!ScheduleInfoOpenScan". SQLSTATE=42724.
```

Эта ошибка вызвана ошибкой компоновщика времени выполнения системы Solaris. Чтобы исправить эту ошибку, примените следующие программные исправления:

105490-06 (107733 делает 105490 устаревшим) для Solaris 2.6

Справка по файлу `DPREPL.DFT`

В Центре управления в справке по странице Репликация записной книжки Параметры инструментов в шаге 5d сказано:

Сохраните этот файл в рабочем каталоге для Центра управления (например, `SQLLIB\BIN`), чтобы система могла использовать его в качестве файла по умолчанию.

Правильный текст для шага 5d:

Сохраните этот файл в рабочем каталоге для Центра управления (`SQLLIB\CC`), чтобы система могла использовать его в качестве файла по умолчанию.

Электронная справка по Центру управления, выполняемому как апплет

Если Центр управления выполняется как апплет, клавиша F1 работает только в окнах и записных книжках, для которых есть всплывающие справки.

Чтобы получить всплывающие справки, можно нажать клавишу F1 в следующих компонентах:

- DB2 Universal Database for OS/390
- Мастеры

В остальных компонентах Центра управления клавиша F1 не вызывает справку. Для вывода справки по другим компонентам используйте кнопку Справка или выпадающее меню Справка.

Выполнение Центра управления в режиме апплета (Windows 95)

При попытке открыть Центр сценариев может возникнуть ошибка, если заданы неверные ID пользователя и пароль. Убедитесь, что при регистрации в Центре управления введены правильные ID пользователя и пароль.

Центр управления DB2 for OS/390

В первом абзаце раздела “Центр управления 390” говорится:

The DB2 UDB Control Center for OS/390 allows you to manage the use of your licensed IBM DB2 utilities. Utility functions that are elements of separately orderable features of DB2 UDB for OS/390 must be licensed and installed in your environment before being managed by the DB2 Control Center.

Этот раздел должен выглядеть так:

Центр управления DB2 for OS/390 позволяет управлять использованием ваших лицензированных утилит DB2 IBM. Фун

Глава 17. Информационный центр

Ошибка "Неверный ярлык" в операционной системе Windows

При использовании Информационного центра может возникнуть ошибка "Неверный ярлык". Если недавно был установлен новый браузер Web (или новая версия браузера), убедитесь, что документы HTML и HTM связаны с правильным браузером. Посмотрите в Справке Windows раздел "Чтобы выбрать программу, открывающую файл".

Открытие внешних ссылок Web в Netscape Navigator, если Netscape уже открыт (системы на основе UNIX)

Если Netscape Navigator уже открыт и в нем выводится содержимое локального документа HTML DB2 или документа с внешнего сайта Web, при попытке открыть внешний сайт Web из Информационного центра возникнет ошибка Netscape. В сообщении об ошибке будет сказано: "Netscape не может найти файл или каталог с именем <external site>."

Чтобы обойти эту ошибку, закройте открытый браузер Netscape *перед* тем, как открывать внешний сайт Web. Netscape будет перезапущен и покажет этот внешний сайт Web.

Учтите, что эта ошибка не возникает при попытке открыть локальный документ HTML DB2, когда Netscape уже открыт.

Проблемы запуска Информационного центра

В некоторых системах Информационный центр может запускаться с запаздыванием, если он запущен через меню Пуск, программу Первые шаги или командой **db2ic**. Если вы столкнулись с этой проблемой, запустите Центр управления и затем выберите Справка —> Информационный центр.

Глава 18. Мастера

Задание размера экстента в мастере по созданию базы данных

В мастере по созданию базы данных можно задать параметры размер экстента и размер предварительной выборки для пользовательского табличного пространства (но не для таблиц каталога или временных таблиц) новой базы данных. Эта возможность разрешена, только если на странице "Пользовательские таблицы" этого мастера для этого пользовательского табличного пространства задан хотя бы один контейнер.

Часть 6. Возможности для бизнеса

Глава 19. Центр хранилищ данных	159		Задание Excel как источника хранилища	175
Создание шагов SQL с информацией об издании	159		Определение и запуск процессов	175
Изменение источников и назначений в окне Модель процесса	159		Определение значений для программы VWPMSV	175
Добавление описаний к объектам Центра хранилищ данных	159		Изменения в приложении Пример хранилища данных	176
Запросы примера содержания	159		Сообщения Центра хранилищ данных	176
Editing a Create DDL SQL statement.	159		Создание макета и загрузка данных на сервере интеграции OLAP DB2	176
Перенастройка бизнес-просмотров Visual Warehouse	160		Использование Classic Connect с Центром хранилищ данных	176
Генерация таблиц назначения и первичных ключей	160		Структура среды Центра хранилищ данных	177
Использование драйверов ODBC Merant	160		Использование преобразователя инвертирования	177
Новый драйвер ODBC	160		Доступ агента хранилища DB2 Версии 7.1 к данным DB2 Версии 5.	177
Определение источника или потребителя хранилища в базе данных OS/2.	161		Перенастройка серверов DB2 Версии 5	177
Наблюдение за состоянием управляющие базы данных хранилища.	161		Изменение конфигурации агента	177
Использование SQL Assist с базой данных примера TBC_MD	161		Глава 3. Setting up warehouse sources	179
Использование функции FormatDate	161		Отображение поля Memo в Microsoft Access на источник хранилища	179
Изменение параметра языка	161		Глава 10. Maintaining the Warehouse Database	179
Использование преобразователя Генерировать таблицу ключей.	162		Компоновка таблиц в подтип шага для программы DB2 UDB RUNSTATS.	179
Поддержание соединений с базами данных	162		Сообщения Центра хранилищ данных	179
Установка удаленного клиента Центра хранилищ данных	162		Электронная справка Центра хранилищ данных	179
Определение источника хранилища DB2 for VM	162		Определение таблиц или производных таблиц для репликации	179
Определение таблицы назначения DB2 for VM или DB2 for VSE	162		Выполнение VWP Essbase с агентом AS/400.	180
Включение поддержки идентификаторов с ограничителями	162		Использование окна Сделать доступными метаданные Центра хранилищ данных и связанного с ним окна свойств	180
Ошибка Data Joiner, указывающая на проблемы связывания	163		Внешние ключи	180
Настройка и выполнение репликации с помощью Центра хранилищ данных	163		Записные книжки репликации	180
Советы по устранению неисправностей	166		Импорт языка тегов	180
Поддерживаемые источники и потребители.	166		Ссылки для добавления данных	180
Дополнения для поддерживаемых источников других типов (не баз данных IBM).	168		Импорт таблиц	180
Создание источника данных вручную в Центре хранилищ данных	169		Исправления для электронной справки по RUNSTATS и REORGANIZE TABLE	180
Импорт и экспорт метаданных с помощью CWM1 (Common Warehouse Metadata Interchange - общий обмен метаданными хранилищ)	170		Страница Уведомление (записные книжки Свойства хранилища данных и Расписание).	181
Введение	170		Поле Модуль агента в записной книжке Системы агентов	181
Импорт метаданных	170		Исправленный учебник Business Intelligence Tutorial	181
Изменение ваших метаданных после запуска утилиты импорта	172			
Экспорт метаданных	172			
Шаг утилиты Runstats в OS/390	174			
Шаг утилиты Load в OS/390	174			
Глава 20. Публикации по Центру хранилищ данных	175			
Data Warehouse Center Application Integration Guide	175			
Data Warehouse Center Administration Guide	175			
Устранение неисправностей	175			
			Глава 21. Управляющая база данных хранилища	183
			Управляющая база данных хранилища по умолчанию	183
			Окно Менеджер управляющей базы данных хранилища	183
			Изменение активной управляющей базы данных хранилища	183
			Создание и инициализация управляющей базы данных хранилища.	184
			Перенастройка управляющих баз данных IBM Visual Warehouse	184
			Доступ к управляющим базам данных хранилища	185

Глава 22. Начальный комплект OLAP	187
Web-сайт сервера OLAP	187
Установка Начального комплекта OLAP DB2 в AIX и Solaris	187
Регистрация с рабочего стола Сервера интеграции OLAP	187
Пример регистрации для Начального комплекта	188
Создание и конфигурирование вручную баз данных примера для Сервера интеграции OLAP	188
Известные проблемы и ограничения	189
Начальный комплект OLAP DB2	190
Клиент рабочего стола OLAP DB2	190
Клиенты электронных таблиц	191
Сервер интеграции OLAP DB2	192
Для модуля электронных таблиц Начального комплекта OLAP требуется последнее исправление Windows	194
Отсутствие файлов EQD встраиваемого модуля OLAP для электронных таблиц	194
Поддержка ассоциаций атрибутов	195
Книги, в которые были внесены изменения для нового Начального комплекта OLAP DB2	195
Глава 23. Менеджер хранилищ данных DB2	197
"Менеджер хранилищ данных" следует читать как "Менеджер хранилищ данных DB2"	197
Утилита инициализации менеджера каталогов данных	197
Вопросы лицензирования	197
Вопросы установки	198
Менеджер каталогов данных для Web	198
Information Catalog Manager Administration Guide	198
Доступ к каталогам данных DB2 Версии 5 при помощи менеджера каталогов данных DB2 Версии 7.1	198
Конфигурирование каталога данных	199
Обмен метаданными с другими продуктами	199
Обмен метаданными при помощи командами flgnxoln	200
Обмен метаданными при помощи команды MDISDGC	200
Вызов программ	200
Information Catalog Manager Programming Guide and Reference	200
Коды причин Менеджера каталогов данных	200
Information Catalog Manager User's Guide	201
Менеджер каталогов данных: Сообщения	201
Сообщение FLG0260E.	201
Сообщение FLG0051E.	201
Сообщение FLG0003E.	201
Сообщение FLG0372E.	201
Сообщение FLG0615E.	201
Менеджер каталогов данных: Электронная справка	201
Руководство администратора Query Patroller	202
Клиент DB2 Query Patroller - это отдельный компонент	202
Установка Query Patroller вручную в AIX и Solaris	202
Создание схемы Query Patroller и связывание файлов связывания программ	203
Команды установки вручную	204

Перенастройка из Версии 6 DB2 Query Patroller с использованием dqpmigrate	206
Включение управления запросами	207
Запуск QueryAdministrator	207
Управление пользователями	207
Создание очереди заданий	207
Использование интерфейса командной строки	207
Замечания по Query Enabler	208
DB2 Query Patroller Tracker может вернуть пустую страницу столбцов	208
Приложение B. Troubleshooting DB2 Query Patroller Clients (Устранение неисправностей клиентов DB2 Query Patroller)	208

Глава 19. Центр хранилищ данных

Создание шагов SQL с информацией об издании

При создании шагов SQL с информацией об издании на основе использования можно создать индекс (не индекс уникальности) для столбца издания, чтобы ускорить производительность операций удаления изданий. Делайте это только для больших таблиц хранилища данных, поскольку при вставке небольшого числа строк производительность операций вставки может ухудшаться.

Изменение источников и назначений в окне Модель процесса

Если изменить источник или назначение в окне Модель процесса, это изменение будет немедленно автоматически сохранено. Если сделаны другие изменения (например, добавление шага), необходимо явно сохранить изменения, чтобы сделать их постоянными. Чтобы сохранить изменения, выберите Процесс —> Сохранить.

Добавление описаний к объектам Центра хранилищ данных

В поле Описание записной книжки Центр хранилищ данных можно задать до 254 символов. Используется эта максимальная длина, а не та, что указана в электронной справке.

Запросы примера содержания

- Для источника типа плоский файл нельзя успешно выполнить запрос примера содержания, использующий агент AS/400. Хотя можно создать источник типа плоский файл и попытаться использовать агент AS/400 для выдачи запроса примера содержания, этот запрос завершится неудачно.
- Можно получить сообщение об ошибке при выполнении запроса примера содержания для назначения хранилища данных в окне модели процесса. Эта ошибка связана с доступностью общей системы агента для источника хранилища данных, назначения хранилища данных и шага в процессе. Список доступных систем агентов для шага - пересечение множеств систем агентов IR источника хранилища данных, систем агентов IR назначения хранилища данных и систем агентов, доступных для конкретного шага (эти шаги выбраны на последней странице записной книжки свойств систем агентов). Например, вы можете захотеть увидеть пример содержания для процесса, выполняющего программу копирования файла при помощи FTP (VWPRCPY). Используемый в этом процессе шаг должен быть выбран для системы агента в определении этой системы агента. При выполнении запроса примера содержания для файла назначения обычно используется первая система агента в выбранном списке. Однако операции поддержки базы данных могут влиять на порядок, в котором заданы системы агентов в этом списке. Запросы примера содержания завершатся ошибкой, если выбранная система агента находится не на то же системе, что файл назначения и файл источника.

Editing a Create DDL SQL statement

При попытке отредактировать оператор SQL Create DDL для таблицы назначения для шага в режиме разработки вы можете увидеть следующее вводящее в заблуждение сообщение: “Изменения оператора SQL Create DDL не будут отражены в определении таблицы или реальной физической таблице. Продолжить?”

Изменения будут отражены в реальной физической таблице. Игнорируйте это сообщение и продолжайте изменять оператор Create DDL.

Исправленная версия этого сообщения для шагов в режиме разработки выглядит так: "Изменения оператора SQL Create DDL не будут отражены в определении таблицы. Продолжить?"

Для шагов в режиме тестирования или производства это сообщение правильно. Центр хранилищ данных не будет изменять физическую таблицу назначения, созданную перед переводом шага в режим тестирования.

Перенастройка бизнес-просмотров Visual Warehouse

Если нужно перенастроить бизнес-просмотры синхронизации метаданных Visual Warehouse в Центр хранилищ данных, переведите эти бизнес-просмотры в режим производства перед перенастройкой управляющей базы данных хранилища данных. Если бизнес-просмотры находятся в режиме производства, их расписания будут перенастроены в Центр хранилищ данных. Если бизнес-просмотры находятся в другом режиме, они будут перенастроены в режим тестирования без их расписаний. Перенастроенные шаги нельзя перевести в режим производства. Необходимо заново создать шаги синхронизации в Центре хранилищ данных и удалить перенастроенные шаги.

Генерация таблиц назначения и первичных ключей

Когда Центр хранилищ данных генерирует таблицу назначения для шага, он не генерирует первичный ключ для таблицы назначения. Некоторые из преобразований (например, Скользящее среднее) используют сгенерированную таблицу в качестве таблицы источника и также требуют, чтобы таблица источника имела первичный ключ. Прежде, чем использовать с преобразованием сгенерированную таблицу, определите для нее первичный ключ, щелкнув правой кнопкой мыши по этой таблице в Центре управления DB2 и выбрав **Изменить**.

Использование драйверов ODBC Merant

Для обращения к Microsoft SQL Server в Windows NT с использованием драйверов ODBC Merant убедитесь, что системный путь содержит каталог sqllib\odbc32.

Новый драйвер ODBC

Если вы будете использовать агент Центра хранилищ данных AIX или Sun, который связан для обращения к источникам ODBC Merant, а также будет обращаться к базам данных DB2, измените значение атрибута "Driver=" в разделе источников DB2 файла .odbc.ini следующим образом:

AIX: имя драйвера - /usr/lpp/db2_07_01/lib/db2_36.o

Пример записи об источнике ODBC для AIX:

```
[SAMPLE] Driver=/usr/lpp/db2_07_01/lib/db2_36.o
Description=DB2 ODBC Database
Database=SAMPLE
```

Sun: имя драйвера - /opt/IBMDB2/V7.1/lib/libdb2_36.so

Пример записи об источнике ODBC для Sun:

```
[SAMPLE] Driver=/opt/IBMDB2/V7.1/lib/libdb2_36.so
Description=DB2 ODBC Database
Database=SAMPLE
```

Определение источника или потребителя хранилища в базе данных OS/2

При определении источника хранилища данных или назначения хранилища данных для базы данных OS/2 вводите имя базы данных в верхнем регистре.

Наблюдение за состоянием управляющие базы данных хранилища

Центр управления DB2 или процессор командной строки могут указывать, что управляющая база данных хранилища данных находится в несовместимом состоянии. Информация об этом состоянии может появляться, поскольку это состояние указывает, что сервер хранилища данных не передал свое исходное сообщение запуска зарегистрированному в хранилище данных пользователю.

Использование SQL Assist с базой данных примера TBC_MD

Для содержащегося в базе данных TBC_MD примера использования хранилища данных нельзя использовать SQL Assist для изменения оператора SQL в шаге выбора сценария SQL, поскольку этот оператор SQL был отредактирован после его создания в SQL Assist.

Использование функции FormatDate

Чтобы использовать функцию FormatDate, нажмите кнопку **Построить SQL** на странице Оператор SQL записной книжки Свойства для шага SQL.

Выходное значение функции FormatDate имеет тип данных varchar(255). Нельзя изменить этот тип данных, выбрав тип данных дата, время или дата/время из списка **Категория** на странице Параметры функции - FormatDate page.

Изменение параметра языка

В операционных средах AIX и Solaris процесс установки задает язык для публикации каталога данных и экспорта на сервер интеграции OLAP. Если нужно использовать эти функции в языке, отличном от языка, заданном во время установки, создайте следующую программную связь, введя следующую команду в одну строку:

В AIX

```
/usr/bin/ln -sf /usr/lpp/db2_07_01/msg/локаль/flgnxolv.str
/usr/lpp/db2_07_01/bin/flgnxolv.str
```

локаль Обозначение языка в формате xx_yy

В операционной среде Solaris

```
/usr/bin/ln -sf /opt/IBMDB2/V7.1/msg/локаль/flgnxolv.str
/opt/IBMDB2/V7.1/bin/flgnxolv.str
```

локаль Обозначение языка в формате xx_yy

Использование преобразователя Генерировать таблицу ключей

При использовании опции **Изменить значение в ключевом столбце** преобразования Генерировать таблицу ключей это преобразование изменяет только те строки в таблице, в которых нет значений ключей, то есть ключи с пустыми значениями. После вставки в таблицу дополнительных строк они будут содержать пустые значения ключей, пока вновь не будет выполнено это преобразование.

Чтобы избежать этой ситуации, используйте следующий подход:

- При повторном выполнении этого преобразования используйте опцию **Заменить все значения**, чтобы заново сгенерировать ключи для всех строк.

Поддержание соединений с базами данных

Сервер хранилища данных не поддерживает соединений с локальными и удаленными базами данных, если управляющий этими базами данных сервер DB2 остановлен и заново запущен. Если система DB2 была остановлена и заново запущена, остановите и заново запустите также службы хранилищ данных.

Установка удаленного клиента Центра хранилищ данных

Чтобы при установке клиента администратора DB2 и инструментов хранилища данных настроить клиент администратора Центра хранилищ данных на различных рабочих станциях с одной рабочей станции, содержащей сервер хранилища данных, необходимо добавить номер порта TCP/IP, используемого рабочей станцией сервера хранилища данных для получения файла служб для рабочей станции клиента. Добавьте в файл служб следующую запись:

```
vwkernel          11000/tcp
```

Определение источника хранилища DB2 for VM

При определении источника хранилища данных для базы данных DB2 for VM, обращение к которой производится через шлюз DRDA, есть ограничения на использование типов данных CLOB и BLOB:

- Нельзя использовать функцию примера содержания для просмотра данных в типах данных CLOB и BLOB.
- Нельзя использовать столбцы типов данных CLOB и BLOB с шагом SQL.

Это известные ограничения для сервера DB2 for VM Версии 5.2, в которой большие объекты нельзя передавать клиенту DB2 Версии 7.1, используя DRDA.

Определение таблицы назначения DB2 for VM или DB2 for VSE

При определении в Центре хранилищ данных таблицы назначения DB2 for VM или DB2 for VSE не включайте переключатель **Предоставить всем (public)**. Синтаксис команды GRANT, генерируемой Центром хранилищ данных, не поддерживается DB2 for VM и DB2 for VSE.

Включение поддержки идентификаторов с ограничителями

Чтобы включить поддержку идентификаторов с ограничителями для сервера Sybase и Microsoft SQL Server в Windows NT: включите переключатель **Разрешить идентификаторы в кавычках** на странице Дополнительные записной книжки Настройка драйвера ODBC.

Чтобы включить поддержку идентификаторов с ограничителями для Sybase в UNIX, отредактируйте запись для источника данных Sybase в файле .odbc.ini и задайте атрибут соединения EQL=1.

Ошибка Data Joiner, указывающая на проблемы связывания

Пользователи, использующие DataJoiner с DB2 Версии 7.1 + FixPak 2 или новее, могут получать сообщение, указывающее на проблемы со связыванием.

Например, при использовании источника DataJoiner с агентом Центра хранилищ данных V7.1 вы можете получить следующие сообщения об ошибках:

```
DWC07356E Агент не смог выполнить команду типа "importTableNames" для издания "0" шага "?".
SQL0001N Связывание или прекомпиляция не были завершены успешно.
SQL0001N Пакет "NULLID.SQLL6D05" не найден.  SQLSTATE=51002 RC = 7356  RC2 = 8600
```

Чтобы исправить эту ошибку, добавьте в файл db2cli.ini следующие строки:

```
[COMMON]
DYNAMIC=1
```

В системах UNIX файл db2cli.ini находится в каталоге .../sqllib/cfg. В Windows NT файл db2cli.ini находится в каталоге .../sqllib.

Настройка и выполнение репликации с помощью Центра хранилищ данных

1. Для настройки и выполнения репликации с помощью Центра хранилищ данных требуется, чтобы управляющие таблицы репликации существовали как в управляющей базе данных хранилища, так и в базах данных назначения хранилища.

Для репликации необходимо, чтобы управляющие таблицы репликации существовали и управляющей базе и базах данных назначения. Управляющие таблицы репликации находятся в схеме ASN и все их имена начинаются с IBMSNAP. Управляющие таблицы репликации автоматически создаются в базе данных при определении источника репликации с помощью Центра управления, если эти управляющие таблицы еще не существуют. Учтите, что эти управляющие таблицы должны также существовать в базе данных назначения. Чтобы создать набор управляющих таблиц в базе данных назначения можно создать источник репликации с помощью Центра управления и затем удалить этот источник репликации, не удаляя управляющие таблицы. Можно также определить управляющие таблицы, используя продукт DJRA (Data Joiner Replication Administration).

2. Установка и использование DJRA

Чтобы для определения управляющих таблиц использовать DJRA, необходимо сначала установить этот продукт. DJRA поставляется вместе с DB2. Чтобы установить DJRA, перейдите в каталог d:\sqllib\djra (где установлена система DB2) и запустите пакет djra.exe. DJRA будет установлен в системе. После этого для запуска DJRA в Windows NT из меню Пуск выберите DB2 for Windows NT и затем выберите Репликация и затем Инструменты управления репликацией. Интерфейс DJRA немного отличается от обычного интерфейса прикладных программ NT. Для каждой функции, которую он выполняет, он создает набор операторов SQL, но не выполняет их. Пользователь должен вручную сохранить сгенерированные операторы SQL и затем использовать для их выполнения функцию Выполнить SQL.

3. Настройка выполнения программ Capture и Apply

Инструкции по конфигурированию используемой системы для выполнения программ Capture и Apply смотрите в справочном руководстве Replication Guide and Reference Manual. Необходимо связать программы Capture и Apply с каждой базой данных, для которых они будут использоваться. Обратите внимание на то, что НЕ нужно создавать файл паролей. Центр хранилищ данных автоматически создаст файл паролей для регистрации репликации.

4. Определение источника репликации в Центре управления

Для определения источника репликации используйте Центр управления. Центр хранилищ данных поддерживает пять типов репликации: Пользовательская копия, Момент времени, Базовая сводка, Сводка изменений и Промежуточные таблицы (таблицы CSD). Для типов Пользовательская копия, Момент времени и Сжатая промежуточная таблица требуется, чтобы таблица источника репликации имела первичный ключ. Для других типов репликации это не требуется. Имейте это в виду при выборе входной таблицы, определяемой в качестве источника репликации. Источник репликации на самом деле представляет собой определение исходной таблицы источника и созданной таблицы CD (таблицы изменений данных), хранящей информацию об изменениях данных до того, как они будут перемещены в таблицу назначения. При определении источника репликации в Центре управления в ASN.IBMSNAP_REGISTER записывается определение источника и его таблицы CD. Одновременно создается таблица CD, но изначально она не содержит данных. При определении источника репликации можно выбрать, включать только столбцы образа данных после изменения или столбцы образа данных после изменения и образа данных перед изменением. Для этого используйте переключатели на странице Источник репликации Центра управления. Затем для выбранных столбцов образов данных перед изменением и после изменения создаются столбцы в новой таблице CD. В таблице CD столбцы образа данных после изменения имеют те же имена, что столбцы исходной таблицы источника. Имена столбцов образа данных перед изменением начинаются с символа 'X'.

5. Импорт источника репликации в Центр хранилищ данных

Создав источник репликации в Центре управления, можно импортировать его в Центр хранилищ данных. При импорте источника репликации включите переключатель "Таблицы, которые можно реплицировать". Тогда Центр хранилищ данных посмотрит записи в таблице ASN.IBMSNAP_REGISTER, чтобы узнать какие таблицы были определены как источники репликации.

6. Определение шага репликации в Центре хранилищ данных

В окне Модель процесса выберите один из пяти типов репликации: Базовая сводка, Сводка изменений, Момент времени, Промежуточная таблица и Пользовательская копия. Если нужно определить тип репликации Базовая сводка или Сводка изменений, посмотрите идущий ниже раздел о том, как настроить репликацию типа Базовая сводка или Сводка изменений в Центре хранилищ данных. Выберите соответствующий источник репликации для этого типа репликации. Как указано выше, для типов Пользовательская копия, Момент времени и Сжатая промежуточная таблица требуется, чтобы таблица источника репликации имела первичный ключ. Соедините источник репликации с шагом репликации. Откройте свойства шага репликации. Перейдите за страницу Параметры. Выберите нужные столбцы. Включите переключатель для создания таблицы назначения. Выберите потребитель хранилища данных. Перейдите на страницу Опции обработки и заполните поля параметров. Нажмите кнопку ОК.

7. Запуск программы Capture

В окне DOS введите команду: ASNCCP база-данных-источника COLD PRUNE
Параметр COLD задает холодный запуск и вызывает удаление всех существующих данных из таблиц CD. Параметр PRUNE указывает программе Capture, что нужно поддерживать таблицу IBMSNAP_PRUNCNTL. Оставьте

работать программу Capture. В нужное время ее можно будет остановить, нажав клавиши Ctrl-Break в ее окне DOS. Не забудьте, что нужно запустить программу Capture перед запуском программы Apply.

8. Перевод шага репликации в режим тестирования

Вернитесь в Центр хранилищ данных и переведите определенный шаг репликации в режим тестирования. В результате информация регистрации репликации будет записана в управляющие таблицы репликации. В таблицы IBMSNAP_SUBS_SET, IBMSNAP_SUBS_MEMBR, IBMSNAP_SUBS_COLS и IBMSNAP_SUBS_EVENT будут добавлены записи для поддержки этого набора регистрации. Кроме этого, в базе данных назначения будет создана таблица назначения. Если тип репликации - Пользовательская копия, Момент времени или Сжатая промежуточная таблица, то таблица назначения должна иметь первичный ключ. Перейдите в Центр управления и создайте этот первичный ключ. Учтите, что для некоторых таблиц назначения репликации также требуются индексы уникальности для различных столбцов. Центр хранилищ данных создает эти индексы уникальности при создании таблицы, поэтому вам НЕ нужно создавать их самостоятельно. Но имейте в виду, что при определении в Центре управления первичного ключа для таблицы, уже имеющей для этого столбца индекс уникальности, будет выдано предупреждение. Игнорируйте его.

9. Перевод шага репликации в режим производства

При переводе шага репликации в режим производства не выполняются изменения регистрации репликации. Как и для других шагов, эта операция затрагивает только Центр хранилищ данных.

10. Выполнение шага репликации

После перевода шага репликации в режим тестирования его можно выполнить. Перед внесением каких-либо изменений в таблицу источника сначала выполните этот шаг. Перейдите в раздел Выполняемая работа (WIP) и выберите этот шаг репликации. Выполните его. В таблицу IBMSNAP_SUBS_EVENT будет добавлена запись о событии, а запись регистрации в IBMSNAP_SUBS_SET станет активной. Сразу же выполняется набор регистрации. При выполнении набора регистрации агент вызывает программу Apply, обрабатывающую активные регистрации. Если после этого изменить исходную таблицу источника, измененные данные будут записаны в таблицу CD. Если после этого выполнить шаг репликации, снова запустится программа Apply и она переместит измененные данные из таблицы CD в таблицу назначения.

11. Изменение режима шага репликации на режим тестирования

При изменении режима шага репликации на режим тестирования не выполняются изменения регистрации репликации. Как и для других шагов, эта операция затрагивает только Центр хранилищ данных.

12. Понижение режима шага репликации до режима разработки

При изменении режима шага репликации на режим разработки из управляющих таблиц репликации удаляется информация о регистрации. После завершения изменения режима на режим разработки в управляющих таблицах репликации не останется записей для этой конкретной регистрации. Будет также отброшена таблица назначения. Таблица CD будет оставлена, так как она относится к определению источника репликации.

13. Как настроить репликацию типа Базовая сводка или Сводка изменений в Центре хранилищ данных.

- Входная таблица. Выберите входную таблицу, которую можно использовать с оператором GROUP BY. Для нашего примера мы будем использовать входную таблицу со следующими столбцами: SALES, REGION, DISTRICT.
- Шаг репликации. Выберите тип репликации Базовая сводка или Сводка изменений. Откройте свойства шага.

- При выполнении программы Apply она должна выполнять оператор SELECT такого типа: SELECT SUM(SALES), REGION, DISTRICT GROUP BY REGION, DISTRICT. Поэтому в качестве выходных столбцов вам нужно выбрать столбцы REGION, DISTRICT и один вычисляемый столбец SUM(SALES). Используйте кнопку Добавить вычисляемый столбец. Для нашего примера введите SUM(SALES) в поле Выражение. Сохраните его.
- Условие WHERE. Если для шага репликации требуется только условие GROUP BY, необходимо также задать фиктивное условие WHERE (например, 1=1). Не включайте в условие WHERE слово "WHERE". Это связано с тем, что в пользовательском интерфейсе Центра хранилищ данных для репликации типа Базовая сводка есть только поле ввода условия WHERE. Для нашего примера введите в этом поле: 1=1 GROUP BY REGION, DISTRICT. Для репликации типа Сводка изменений есть поле ввода условия WHERE и поле ввода условия GROUP BY: В поле условия WHERE введите: 1=1, а в поле ввода условия GROUP BY введите: GROUP BY REGION, DISTRICT
- Задайте остальные свойства шага так же, как для других типов репликации. Нажмите кнопку ОК, чтобы сохранить этот шаг и создать объект таблицы назначения.
- Откройте объект таблицы назначения. Теперь нужно переименовать выходной столбец для выражения вычисляемого столбца на допустимое имя столбца и задать для этого столбца правильный тип данных. Сохраните объект таблицы назначения.
- Переведите этот шаг репликации в режим тестирования. Будет создана таблица назначения. Для нее НЕ требуется первичный ключ.
- Выполните этот шаг (как любой другой шаг репликации).

Советы по устранению неисправностей

- Чтобы включить трассировку для программы Apply, на панели Свойства хранилища данных задайте для параметра Трассировка агента значение 4. При этом значении агент включит полную трассировку для программы Apply.
Если таблица CD не содержит данных, то скорее всего или не была запущена программа Capture, или не были изменены данные исходной таблицы источника.
- В электронной справке пропущено поле почтового сервера на странице Уведомление записной книжки Расписание.
- Для работы с уведомлениями Центра хранилищ данных почтовый сервер должен поддерживать ESMTP. В справке Открытие окна Выполняемая работа, выбирайте Хранилище данных → Выполняемая работа вместо Центр хранилищ данных → Выполняемая работа.

Поддерживаемые источники и потребители

В следующих таблицах перечислены уровни версий и выпусков источников и потребителей, поддерживаемые Центром хранилищ данных.

Таблица 4. Уровни версий и выпусков поддерживаемых источников хранилища данных IBM

Источник	Версия/выпуск
IMS	5.1
DB2 Universal Database for Windows NT	5.2 - 7.1
DB2 Universal Database Enterprise-Extended Edition	5.2 - 7.1

Таблица 4. Уровни версий и выпусков поддерживаемых источников хранилища данных IBM (продолжение)

Источник	Версия/выпуск	
DB2 Universal Database for OS/2	5.2 - 7.1	
DB2 Universal Database for AS/400	3.7 - 4.5	
DB2 Universal Database for AIX	5.2 - 7.1	
DB2 Universal Database для операционных сред Solaris	5.2 - 7.1	
DB2 Universal Database for OS/390	4.1 - 5.1.6	
DB2 DataJoiner	2.1.1	
DB2 for VM	5.3.4 или более новый	
DB2 for VSE	7.1	

Источник	Windows NT	AIX
Informix	7.2.2 - 8.2.1	7.2.4 - 9.2.0
Oracle	7.3.2 - 8.1.5	8.1.5
Microsoft SQL Server	7.0	
Microsoft Excel	97	
Microsoft Access	97	
Sybase	11.5	11.9.2

Таблица 5. Уровни версий и выпусков поддерживаемых потребителей хранилища данных IBM

Потребитель	Версия/выпуск
DB2 Universal Database for Windows NT	6 - 7
DB2 Universal Database Enterprise-Extended Edition	6 - 7
DB2 Universal Database for OS/2	6 - 7
DB2 Universal Database for AS/400	3.1-4.5
DB2 Universal Database for AIX	6 -7
DB2 Universal Database для операционных сред Solaris	6 -7
DB2 Universal Database for OS/390	4.1 - 6
DB2 DataJoiner	2.1.1
DB2 DataJoiner/Oracle	8
DB2 for VM	3.4 - 5.3.4
DB2 for VSE	3.2, 7.1
CA/400	3.1.2

Дополнения для поддерживаемых источников других типов (не баз данных IBM)

Следующая таблица содержит дополнения для поддерживаемых источников других типов (не баз данных IBM):

База данных	Операционная система	Требования к клиенту базы данных
Informix	AIX	Informix-Connect и ESQL/C версии 9.1.4 или более поздней
Informix	Операционная среда Solaris	Informix-Connect и ESQL/C версии 9.1.3 или более поздней
Informix	Windows NT	Informix-Connect for Windows Platforms 2.x или Informix-Client Software Developer's Kit for Windows Platforms 2.x
Oracle 7	AIX	Oracle7 SQL*Net и совместно используемая библиотека Oracle7 SQL*Net (построенная при помощи сценария genclntsh)
Oracle 7	Операционная среда Solaris	Oracle7 SQL*Net и совместно используемая библиотека Oracle7 SQL*Net (построенная при помощи сценария genclntsh)
Oracle 7	Windows NT	Соответствующие DLL для текущей версии SQL*Net, плюс OCIW32.DLL. Например, для SQL*Net 2.3 требуются ORA73.DLL, CORE35.DLL, NLSRTL32.DLL, CORE350.DLL и OCIW32.DLL.
Oracle 8	AIX	Oracle8 Net8 и совместно используемая библиотека Oracle8 SQL*Net (построенная при помощи сценария genclntsh8)
Oracle 8	Операционная среда Solaris	Oracle8 Net8 и совместно используемая библиотека Oracle8 SQL*Net (построенная при помощи сценария genclntsh8)

База данных	Операционная система	Требования к клиенту базы данных
Oracle 8	Windows NT	Для доступа к удаленным серверам баз данных Oracle8 с уровнем версии 8.0.3 или более новой установите клиент Oracle Net8 Client версии 7.3.4.x, 8.0.4 или более новой. В системах Intel установите в своем пути соответствующие DLL для клиента Oracle Net8 Client (такие как Ora804.DLL, PLS804.DLL и OCI.DLL).
Sybase	AIX	В среде без DCE (драйвер ODBC ibsyb15): Библиотека libct В среде DCE (драйвер ODBC ibsyb1115): Библиотека клиента Sybase 11.1 libct_r
Sybase	Операционная среда Solaris	В среде без DCE (драйвер ODBC ibsyb15): Библиотека libct В среде DCE (драйвер ODBC ibsyb1115): Библиотека клиента Sybase 11.1 libct_r
Sybase	Windows NT	Sybase Open Client-Library 10.0.4 или более поздняя версия и соответствующая Sybase Net-Library.

Создание источника данных вручную в Центре хранилищ данных

Если источник данных создается с помощью Relational Connect и оператора “Create Nickname”, источник данных не будет доступен в функциях, связанных с импортом таблиц в Центр хранилищ данных. Чтобы использовать источник данных как исходную таблицу или таблицу назначения, последовательно выполните следующие действия:

1. Определите источник/потребитель, не импортируя каких-либо таблиц.
2. Разверните дерево Источники/Потребители хранилища из главного окна Центра хранилищ данных и щелкните правой кнопкой мыши по “Таблицы” для нужного источника/потребителя.
3. Выберите **Определить**.
4. Определите источник данных с помощью открывшей записной книжки и проследите, чтобы столбцы были определены для каждого источника данных.

Дополнительную информацию смотрите в разделах “Определение таблицы источника хранилища” или “Определение таблицы потребителя хранилища” в Информационном центре.

Импорт и экспорт метаданных с помощью CWMI (Common Warehouse Metadata Interchange - общий обмен метаданными хранилищ)

Введение

В дополнение к имеющейся поддержке файлов языка тегов Центр хранилищ данных может теперь импортировать и экспортировать метаданные в формате файлов XML, соответствующих стандарту CWM (Common Warehouse Metamodel - общая метамодель хранилища). Импорт и экспорт этих CWM-совместимых файлов XML называется *CWMI (Common Warehouse Metadata Interchange)*.

Метаданные можно импортировать и экспортировать из следующих объектов Центра хранилищ данных:

- Источники хранилища
- Потребители хранилища
- Предметные области, включая процессы, источники, потребители и шаги
- Пользовательские программы

Утилита экспорта и импорта CWMI не поддерживает метаданные некоторых видов, в том числе, планы, схемы хранилища, шаги ярлыков, каскадные связи, пользователи и группы.

Центр хранилищ данных создает файл журнала, который содержит результаты процессов импорта и экспорта. Обычно файл журнала создается в каталоге `x:\program files\sql\lib\logging` (где `x`: - устройство, где вы установили DB2) или в каталоге, заданным переменной среды `VWS_LOGGING`. Файл журнала представляет собой простой текстовый файл, который можно посмотреть в любом текстовом редакторе.

Импорт метаданных

Метаданные можно импортировать как из Центра хранилищ данных, так и из командной строки.

Новые объекты, создаваемые в процессе импорта, назначаются группе защиты Центра хранилищ данных по умолчанию. Дополнительную информацию смотрите в разделе "Изменение защиты после импорта" в данных Замечаниях по выпуску.

Если импортируются метаданные о шаге, с этим шагом может быть связано несколько файлов. Метаданные о шаге хранятся в файле XML, но с шагом могут быть иногда связаны также данные, хранящиеся как двоичные большие объекты. Имя файла метаданных двоичных больших объектов такое же, как у файла XML, но они существуют в виде отдельных файлов с нумерованными расширениями. При импорте все связанные с шагом файлы должны находиться в одном каталоге.

Изменение шагов, находящихся в режиме тестирования или режиме производства

Чтобы Центр хранилищ данных смог изменить метаданные шага, шаг должен находиться в режиме разработки. Если шаг находится в режиме тестирования или производства, переведите его в режим разработки перед импортом метаданных:

1. Зарегистрируйтесь в Центре хранилищ данных.
2. Щелкните правой кнопкой мыши по шагу, который надо перевести в другой режим, и выберите **Режим**.

3. Выберите **Разработка**.

Данный шаг переведен в режим разработки. После завершения импорта метаданных переведите шаг обратно в режим тестирования или производства.

Импорт данных из Центра хранилищ данных

Метаданные можно импортировать из Центра хранилищ данных:

1. Зарегистрируйтесь в Центре хранилищ данных.
 2. Нажмите на левой панели кнопку **Хранилище**.
 3. Выберите **Выбранные** → **Импортировать метаданные**.
 4. В окне Импортировать метаданные задайте имя файла, содержащего метаданные, которые надо импортировать. Имя файла можно ввести самостоятельно или же выбрать в списке.
 - Если положение файла известно, введите полное имя файла, который надо импортировать. Не забудьте указать расширение `.xml`, указывающее, что надо импортировать метаданные в формате XML.
 - Чтобы просмотреть ваши файлы:
 - a. Нажмите кнопку с многоточием (...).
 - b. В окне Файл измените **Файлы типа** на XML.
 - c. Перейдите в правильный каталог и выберите файл, который надо импортировать.
- Примечание:** Файл должен иметь расширение `.xml`.
- d. Нажмите кнопку **ОК**.
5. В окне **Импортировать метаданные** нажмите кнопку **ОК**, чтобы закончить работу. При импорте файла Центром хранилищ данных появляется окно **Ход выполнения**.

Импорт метаданных с помощью командной строки

Для импорта метаданных можно также использовать командную строку. Синтаксис команды импорта:

```
CWMImport файл_XML управляющая_база Id_пользователя пароль [PREFIX = схема]
```

<i>файл_XML</i>	Полное имя (включая диск и каталог) файла XML, который надо импортировать. Это обязательный параметр.
<i>управляющая_база</i>	Имя управляющей базы данных хранилища, в которую вы хотите импортировать метаданные. Это обязательный параметр.
<i>ID_пользователя</i>	ID пользователя, используемый при регистрации в управляющей базе данных хранилища. Это обязательный параметр.
<i>пароль</i>	Пароль, используемый при регистрации в управляющей базе данных хранилища. Это обязательный параметр.
<i>[PREFIX=схема]</i>	Имя схемы базы данных для Центра хранилищ данных, иногда называемое <i>префиксом таблицы</i> . Если значение для PREFIX= не задано, по умолчанию используется имя IWH. Это необязательный параметр.

Изменение ваших метаданных после запуска утилиты импорта

Изменение защиты после импорта

В качестве меры защиты Центр хранилищ данных не импортирует и не экспортирует пароли. Пароли для новых объектов обновляются по мере необходимости. Подробнее об особенностях импорта смотрите в руководстве *Data Warehouse Center Administration Guide*, глава 12, "Exporting and importing Data Warehouse Center metadata".

При импорте метаданных все объекты назначаются *группе защиты по умолчанию*. Группы, имеющие доступ к объекту, можно изменить:

1. Зарегистрируйтесь в Центре хранилищ данных.
2. Щелкните правой кнопкой мыши по папке, содержащей объект, который надо изменить.
3. Выберите **Свойства** и выберите закладку **Защита**.
4. Удалите группы из списка **Выбранные группы хранилища** или добавьте группы в список **Доступные группы хранилища**.
5. Нажмите кнопку **ОК**.

Экспорт метаданных

Метаданные можно экспортировать как из Центра хранилищ данных, так и из командной строки.

Метаданные некоторых шагов хранятся как двоичные большие объекты. Метаданные двоичных больших объектов экспортируются в отдельный файл с тем же именем, что и файл XML шага, но с числовыми расширениями (.1, .2 и так далее).

Экспорт данных из Центра хранилищ данных

Метаданные можно экспортировать из Центра хранилищ данных:

1. Зарегистрируйтесь в Центре хранилищ данных.
 2. Нажмите на левой панели кнопку **Хранилище**.
 3. Выберите **Выбранные** → **Экспорт метаданных** → **Файл обмена**.
 4. В окне Экспорт метаданных задайте имя файла, куда будут записаны экспортированные метаданные. Имя файла можно ввести самостоятельно или выбрать его в списке:
 - Если вы знаете полное имя требуемого файла, введите его в поле ввода **Имя файла**. Не забудьте задать расширение `.xml`, указывающее, что метаданные надо экспортировать в формате XML.
 - Чтобы просмотреть ваши файлы:
 - a. Нажмите кнопку с многоточием (...).
 - b. В окне **Файл** измените **Файлы типа** на XML.
 - c. Перейдите в правильный каталог и выберите файл, в котором должны содержаться экспортированные метаданные.
- Примечание:** Любой существующий файл, выбранный вами, будет перезаписан экспортированными метаданными.
- d. Нажмите кнопку **ОК**.
5. Если в окне Экспорт метаданных выведено имя нужного файла, выберите в списке **Доступные объекты** объект, метаданные которого надо экспортировать.

6. Нажмите кнопку **>**, чтобы переместить выбранный объект из списка **Доступные объекты** в список **Выбранные объекты**. Повторяйте эту операцию, пока все экспортируемые объекты не окажутся в списке **Выбранные объекты**.
7. Нажмите кнопку **ОК**.

Центр хранилищ данных создает *входной* файл, содержащий информацию о выбранных для экспорта объектах, и затем экспортирует метаданные об этих объектах. Во время экспорта метаданных Центром хранилищ данных появляется окно хода выполнения.

По завершении экспорта выдается информационное сообщение о процессе экспорта. Код возврата 0 указывает на успешный экспорт. Подробнее о процессе можно узнать, посмотрев файл журнала.

Экспорт метаданных с помощью командной строки

Прежде, чем экспортировать метаданные из командной строки, необходимо создать *входной* файл. Входной файл - это текстовый файл с расширением `.INP`, в котором перечислены по типу объекта все объекты, которые надо экспортировать. При экспорте из Центра хранилищ данных входной файл создается автоматически, но при экспорте из командной строки, его надо вначале создать самостоятельно. Входной файл можно создать с помощью любого текстового редактора. Введите имена всех объектов, появляющихся в Центре хранилищ данных. Убедитесь, что файл создается в каталоге, для которого есть разрешение на чтение/запись. При запуске утилиты экспорта Центр хранилищ данных записывает файлы XML в тот же каталог, где находится входной файл.

Ниже приводится пример файла ввода:

```
<PROC>
Tutorial Fact Table Process
<IR>
Tutorial file source
Tutorial target
<UDP>
New Program group
```

В разделе `<PROC>` (процессы) перечислите все процессы, которые надо экспортировать. В разделе `<IR>` (информационные ресурсы) перечислите все источники и потребители хранилища, которые надо экспортировать. Центр хранилищ данных автоматически включает все таблицы и столбцы, связанные с этими источниками и потребителями. В разделе `<UDP>` (пользовательские программы) перечислите все группы программ, которые надо экспортировать.

Для экспорта метаданных введите следующую команду в командной строке DOS:

```
CWMEExport входной_файл управляющая_база_данных ID_пользователя пароль
[PREFIX=схема]
```

входной_файл

Полное имя (включая диск и каталог) файла `.INP`, со списком объектов, которые надо экспортировать. Это обязательный параметр.

управляющая_база

Имя управляющей базы данных хранилища, из которой осуществляется экспорт. Это обязательный параметр.

<i>ID_пользователя</i>	ID пользователя, используемый при регистрации в управляющей базе данных хранилища. Это обязательный параметр.
<i>пароль</i>	Пароль, используемый при регистрации в базе данных управления хранилищем. Это обязательный параметр.
<i>[PREFIX=схема]</i>	Имя схемы базы данных для Центра хранилищ данных, иногда называемое <i>префиксом таблицы</i> . Если значение для PREFIX= не задано, по умолчанию используется имя IWH. Это необязательный параметр.

Шаг утилиты Runstats в OS/390

Определяя в Центре хранилищ данных шаг утилиты Runstats в OS/390, надо помнить, что на вкладке Параметры диалогового окна Свойства шага в поле табличного пространства имя надо вводить в верхнем регистре. Если это табличное пространство не находится в базе данных DSND04, к имени табличного пространства надо добавить имя содержащей его базы данных. Например, введите SAMPLE.EMPLOYEE. В справке сейчас это поле не отражено.

Шаг утилиты Load в OS/390

Определяя в Центре хранилищ данных шаг утилиты Runstats в OS/390, надо помнить, что на вкладке Параметры диалогового окна Свойства шага обязательно надо нажать кнопку Дополнительно. В противном случае не будет сгенерировано условие INTO оператора загрузки, и при запуске утилита загрузки завершится неудачно.

Кроме того, в FixPак 3 включено исправление, удаляющее двойные кавычки вокруг имени набора данных загрузки. Без этого исправления загрузка работать не будет.

Глава 20. Публикации по Центру хранилищ данных

Data Warehouse Center Application Integration Guide

В Главе 5 "Metadata Templates" таблица 16 описывает маркеры тегов столбцов. В этом руководстве должно быть сказано, что отсчет "`*ColumnPositionNumber`" должен начинаться с "1". В руководстве ошибочно утверждается, что начальный символ - "0".

В главе 6 "Data Warehouse Center metadata" описание свойства объекта столбца POSNO следует читать так:

Индекс (начиная с 1) столбца или поля в строке в таблице или в файле.

В главе 8 "Типы объектов менеджера каталогов данных" каталог, в котором находятся файлы .TYP, содержащие язык тегов для определения типа объекта, был изменен на `\SQLLIB\DGWIN\TYPES`.

Data Warehouse Center Administration Guide

Устранение неисправностей

Информация об исправлении ошибок для Центра хранилищ данных была перемещена в руководство DB2 Troubleshooting Guide.

Задание Excel как источника хранилища

В Главе 3 "Setting up warehouse sources" в разделе "Setting up non-DB2 database warehouse sources in Windows NT" в описании Microsoft Excel пропущен шаг. Этот шаг приводится ниже как Шаг 3.

Если вы используете для доступа к электронным таблицам Excel драйвер ODBC Microsoft Excel 95/97 ODBC, надо создать именованную таблицу для каждого рабочего листа электронной таблицы. Чтобы создать именованную таблицу для каждого рабочего листа:

1. Выберите нужные столбцы и строки.
2. Выберите Excel → Вставка → Имя → Определить.
3. Убедитесь, что поле ссылок в окне Определить имя содержит ячейки, которые вы выбрали на шаге 1. Если это не так, щелкните по значку справа от поля ссылок для включения всех выбранных ячеек.
4. Введите имя (или используйте имя по умолчанию для маркированных данных).
5. Нажмите кнопку ОК.

Определение и запуск процессов

В главе 5 "Defining and running processes" раздел "Starting a step from outside the Data Warehouse Center" надо указать, что для запуска шага, имя которого содержит двухбайтные символы, на рабочей станции сервера хранилища и на узле агента необходим JDK 1.1.8 или более новый.

Определение значений для программы VWPMVS

На странице 180 в разделе "Defining values for a Submit OS/390 JCL jobstream (VWPMVS) program" в шаге 8 сказано, что необходимо определить файл .netrc в том же каталоге, что и файл JES. На самом деле файл .netrc создается этой программой.

Если этот файл не существует, программа создает его в начальном каталоге. Если в начальном каталоге уже есть файл .netrc, программа переименует его и создаст новый файл. Когда программа завершает свою работу, она удаляет новый созданный ей файл .netrc и переименовывает исходный файл в .netrc.

Изменения в приложении Пример хранилища данных

- В приложении Data warehousing sample в разделе "Viewing and modifying the sample metadata" в список таблиц источника должна быть включена таблица GEOGRAPHIES.
- В этом же приложении в разделе "Promoting the steps" в описании процедуры перевода шага в режим производства следующее предложение неверно, так как таблица назначения создается при переводе шага в режим тестирования:
The Data Warehouse Center starts to create the target table,
and displays a progress window.

Сообщения Центра хранилищ данных

В Microsoft Windows NT и Windows 2000 Центр хранилищ данных записывает информацию о событиях в системный журнал событий. ID события соответствует номеру сообщения Центра хранилищ данных. Информацию о сообщениях Центра хранилищ данных смотрите в руководстве *Справочник по сообщениям*.

Создание макета и загрузка данных на сервере интеграции OLAP DB2

В примере на рисунке 20 на странице 315 есть ошибка. Правильные команды:

```
"C:\IS\bin\olapicmd" < "C:\IS\Batch\my_script.script" >  
"C:\IS\Batch\my_script.log"
```

Двойные кавычки вокруг "C:\IS\bin\olapicmd" необходимы, если имя каталога в пути содержит пробел (например, Program Files).

Использование Classic Connect с Центром хранилищ данных

- В приложении F "Using Classic Connect with the Data Warehouse Center" раздел "Installing the CROSS ACCESS ODBC driver" на странице 388 надо заменить на следующую информацию:

Установите драйвер ODBC CROSS ACCESS, выполнив пользовательскую установку менеджера хранилищ DB2 Версии 7 и выбрав компонент Classic Connect Drivers. При стандартной установке менеджера хранилищ DB2 этот драйвер не устанавливается.

Драйвер ODBC CROSS ACCESS будет установлен в подкаталог ODBC32 каталога SQLLIB. После завершения установки надо вручную добавить путь для драйвера (например, C:\Program Files\SQLLIB\ODBC32) к системной переменной среды PATH. Если у вас уже установлена другая версия драйвера ODBC CROSS ACCESS, путь ... \SQLLIB\ODBC32\ надо поместить перед путем к предыдущей версии. Операционная система будет использовать первый каталог в пути, где найдет драйвер ODBC CROSS ACCESS.

- К приложению F "Using Classic Connect with the Data Warehouse Center" надо добавить следующую процедуру:

Установка драйвера ODBC Classic Connect:

1. Вставьте компакт-диск менеджера хранилищ в дисковод CD-ROM. Откроется панель запуска.
2. Нажмите кнопку **Установить** на панели запуска.

3. В окне Выбор продукта включите переключатель **Менеджер хранилища DB2** и нажмите кнопку **Далее**.
4. В окне Выбор типа установки выберите **Пользовательская** и нажмите кнопку **Далее**.
5. В окне Выбор компонентов выберите **Драйверы Classic Connect** и **Агент хранилища**, выключите остальные переключатели и нажмите **Далее**.
6. В окне Начать копирование файлов посмотрите, что вы выбрали. Если вы хотите изменить выбор, нажмите кнопку **Назад**, чтобы вернуться в окно, где можно изменить сделанный выбор. Нажмите кнопку **Далее**, чтобы начать установку.

Структура среды Центра хранилищ данных

В приложении G "Data Warehouse Center environment structure" на странице 401 в таблице есть неверная запись. Каталог C:\Program Files\SQLLIB\ODBC32 *не* добавляется к переменной среды PATH. К переменной среды PATH добавляется только C:\Program Files\SQLLIB\BIN.

Использование преобразователя инвертирования

В книге говорится, что преобразователь инвертирования может создать таблицу назначения на основе параметров, однако пропущено уточнение, что в сгенерированной таблице назначения не будет требуемых выходных столбцов, которые надо тем самым создавать явно.

Доступ агента хранилища DB2 Версии 7.1 к данным DB2 Версии 5

Агенты хранилища данных DB2 Версии 7.1, сконфигурированные процессом установки DB2 Версии 7.1, будут поддерживать доступ к данным DB2 Версии 6 и DB2 Версии 7.1. Если требуется доступ к данным DB2 Версии 5, необходимо сделать одно из следующих действий:

- Перенастроить серверы DB2 Версии 5 в DB2 Версии 6 или DB2 Версии 7.1.
- Изменить конфигурацию агента (в соответствующей операционной системе), чтобы разрешить доступ к данным DB2 Версии 5.

Агенты хранилища данных DB2 Версии 7.1 не поддерживают доступ к данным из DB2 Версии 2 или более ранних версий.

Перенастройка серверов DB2 Версии 5

Информацию о перенастройке серверов DB2 Версии 5 смотрите в руководстве *DB2 Universal Database: Быстрый старт* для конкретной операционной системы.

Изменение конфигурации агента

Ниже описывается, как изменить конфигурацию агента в каждой операционной системе. При перенастройке серверов DB2 в DB2 Версии 6 или более поздней удалите внесенные в конфигурацию изменения.

Агенты хранилищ UNIX: Чтобы настроить агенты хранилища данных для UNIX для доступа к данным из DB2 Версии 5 с использованием доступа CLI или ODBC:

1. Установите клиент времени выполнения DB2 Версии 6. Его можно получить, выбрав загрузку клиента на странице:
<http://www.ibm.com/software/data/db2/udb/support>
2. Измените файл конфигурации агента хранилища данных, чтобы переменная среды DB2INSTANCE указывала на экземпляр DB2 Версии 6.

3. Внесите в каталог все базы данных в этом экземпляре DB2 Версии 6, к которым будет обращаться агент хранилища данных.
4. Остановите процесс демона агента командой kill с ID процесса демона агента. Демон агента будет перезапущен автоматически. Для остановки процесса требуются полномочия root.

Агенты хранилищ Microsoft Windows NT, Windows 2000 и OS/2: Чтобы настроить агенты хранилища данных для Microsoft NT, Windows 2000 или OS/2 для доступа к данным из DB2 Версии 5:

1. Установите DB2 Connect Enterprise Edition Версии 6 на рабочей станции, отличной от той, где установлен агент хранилища данных DB2 Версии 7.1.

DB2 Connect Enterprise Edition входит в DB2 Universal Database Enterprise Edition и DB2 Universal Database Enterprise - Extended Edition. Если установлена Версия 6 одного из этих продуктов, не нужно отдельно устанавливать DB2 Connect.

Ограничение: На одной рабочей станции Windows NT или OS/2 нельзя установить несколько версий DB2. Можно установить DB2 Connect на другой рабочей станции Windows NT или на рабочей станции OS/2 или UNIX.

2. Сконфигурируйте агент хранилища данных и DB2 Connect Версии 6 для доступа к данным DB2 Версии 5. Дополнительную информацию смотрите в руководстве *Руководство пользователя DB2 Connect*. Далее представлен обзор требуемых шагов:
 - a. В системе DB2 Версии 5 используйте процессор командной строки DB2 для внесения в каталог базы данных Версии 5, к которой будет обращаться агент хранилища данных.
 - b. В системе DB2 Connect используйте процессор командной строки DB2 для внесения в каталог:
 - Узла TCP/IP для системы DB2 Версии 5
 - Базы данных для системы DB2 Версии 5
 - Записи DCS для системы DB2 Версии 5
 - c. На рабочей станции агента хранилища данных используйте процессор командной строки DB2 для внесения в каталог:
 - Узла TCP/IP для системы DB2 Connect
 - Базы данных для системы DB2 Connect

Информацию о внесении в каталог баз данных смотрите в руководстве *Дополнение по установке и конфигурированию DB2 Universal Database*.

3. На рабочей станции агента хранилища данных свяжите пакет CLI DB2 с каждой базой данных, для обращения к которой будет использоваться DB2 Connect. Далее показан пример команд DB2 для связывания с базой данных DB2 Версии 5 с именем v5database. Для ввода этих команд используйте процессор командной строки DB2. Файлы db2cli.lst и db2ajgrt находятся в каталоге \sqllib\bnd.

```
db2 connect to v5database user ID_пользователя using пароль
db2 bind db2ajgrt.bnd
db2 bind @db2cli.lst blocking all grant public
```

где *ID_пользователя* - ID пользователя для базы данных Версии 5, а *пароль* - пароль для этого ID пользователя.

При связывании db2cli.list с базой данных DB2 Версии 5 возникает ошибка. Эта ошибка вызывается тем, что в этой конфигурации не поддерживаются большие объекты. Эта ошибка не влияет на доступ агента хранилища данных к базе данных DB2 Версии 5.

Для обращения к данным DB2 Версии 5 через DB2 Connect требуется FixPak 14 для DB2 Universal Database Версии 5, доступный с июня 2000 года. Номер APAR для этого FixPak - JR14507.

Глава 3. Setting up warehouse sources

Отображение поля Мемо в Microsoft Access на источник хранилища

Поле Мемо базы данных Microsoft Access представляется в источниках Центра хранилищ данных как тип данных LONG VARCHAR с размером столбца больше 1 Гбайта. Для поддержки практических конфигураций систем Центр хранилищ данных усекает значения, превышающие 128 Кбайт. Чтобы значения Мемо-полей в источнике хранилища не усекались, перед использованием таблицы в шаге измените тип данных столбца, который принимает Мемо-поля, с LONG VARCHAR на CLOB. Если не сделать этого, все значения длиннее 128 Кбайт будут усечены.

Для OS/390 и OS/400 необходима поддержка DRDA для типа данных CLOB. Для OS/390 тип данных CLOB поддерживается, начиная с DB2 Версии 6. Для OS/400 тип данных CLOB поддерживается, начиная с Версии 4, Выпуска 4 с DB FixPak 4 или новее (PTF SF99104). Для OS/400 установочный диск Версии 4, Выпуск 4 от февраля 1999 также содержит поддержку типа данных CLOB.

Глава 10. Maintaining the Warehouse Database

Компоновка таблиц в подтип шага для программы DB2 UDB RUNSTATS

Подтип шага для программы RUNSTATS читает из потребителя хранилища и пишет в него. Свяжите потребитель с подтипом шага в окне Модель процесса перед определением значений для шага.

Сообщения Центра хранилищ данных

Сообщение Центра хранилищ данных DWC3778E должно выглядеть так: "Нельзя удалить группу программ по умолчанию Центра хранилищ данных."

Сообщение Центра хранилищ данных DWC3806E должно выглядеть так: "Создаваемый или изменяемый шаг не связан для заполнения с ресурсом источника или программой Центра хранилищ данных."

Сообщение Центра хранилищ данных DWC6119E должно выглядеть так: "Клиент Центра хранилищ данных не смог получить ответа от сервера хранилища."

Электронная справка Центра хранилищ данных

Определение таблиц или производных таблиц для репликации

Прежде, чем таблицу или производную таблицу можно будет использовать в качестве источника репликации в Центре хранилищ данных, ее необходимо определить для репликации, используя Центр управления DB2.

Выполнение VWP Essbase с агентом AS/400

Перед выполнением VWP Essbase с агентом AS/400 нужно задать ARBORLIB и ARBORPATH в качестве переменных среды *sys. Чтобы задать их, ID пользователя должен иметь полномочия *jobctl. Эти переменные среды нужны для задания библиотеки, в которой установлен Essbase.

Использование окна Сделать доступными метаданные Центра хранилищ данных и связанного с ним окна свойств

В шаге 10 справки задания есть пример, в котором сказано, что если для максимального числа уровней объектов в дереве задано значение 1, при опубликовании процесса станет доступным и будет выводиться только шаг 1 этого процесса. Это верно не во всех ситуациях. В шаге 8 во втором абзаце первое предложение неверно. Оно должно выглядеть так: "Выберите опцию На уровне столбцов, чтобы сгенерировать объект преобразования между столбцом источника каталога данных и столбцом назначения."

Внешние ключи

В электронной справке все упоминания внешних ключей относятся ко внешним ключам хранилища данных.

Записные книжки репликации

В электронной справке все упоминания записной книжки Определение репликации относятся к записной книжке шага репликации.

Импорт языка тегов

В электронной справке по импорту языка тегов в списке общих ошибок импорта есть строка "Импортируется файл языка тегов, который не был правильно экспортирован". Эта ошибка не относится к списку общих ошибок импорта.

Ссылки для добавления данных

В теме "Добавление данных" электронной справки не работают ссылки на темы "Добавление к процессу таблиц источника" и "Добавление к процессу таблиц назначения". Эти темы можно найти с помощью указателя справки.

Импорт таблиц

В темах справки "Импорт таблиц и производных таблиц источника" и "Импорт таблиц назначения в потребитель хранилища" есть неверная информация о символах подстановки. Фразу:

Например, если задать XYZ*, будут возвращены таблицы и производные таблицы, имена схем которых начинаются с символов XYZ.

надо читать как:

Например, если задать XYZ%, будут возвращены таблицы и производные таблицы, имена схем которых начинаются с символов XYZ.

Исправления для электронной справки по RUNSTATS и REORGANIZE TABLE

В электронной справке по этим утилитам сказано, что таблица, для которой нужно собрать статистическую информацию или которую нужно реорганизовать, должна быть связана и с источником, и с потребителем. Но поскольку этот шаг работает только с источником, требуется только связь от источника к этому шагу.

Страница Уведомление (записные книжки Свойства хранилища данных и Расписание)

Для страницы Уведомление записной книжки Свойства хранилища данных предложение:

Поле ввода Отправитель изначально содержит строку <ID регистрации текущего пользователя>.

следует изменить на:

Поле ввода Отправитель изначально содержит строку <адрес e-mail текущего зарегистрированного пользователя>.

На странице Уведомление записной книжки Расписание поле Отправитель изначально содержит значение, заданное в записной книжке Свойства хранилища данных. Если там не задано никакое значение, это поле изначально содержит строку адреса e-mail текущего зарегистрированного пользователя. Если с этим пользователем не связан адрес e-mail, полю Отправитель присваивается ID этого пользователя.

Поле Модуль агента в записной книжке Системы агентов

Поле Модуль агента записной книжки Системы агентов содержит имя программы, выполняющейся в момент, когда демон агента хранилища данных порождает процесс агента хранилища данных. Не изменяйте имя в этом поле, если не получите указаний сделать это от IBM.

Исправленный учебник Business Intelligence Tutorial

FixPak 2 содержит исправленный учебник Business Intelligence Tutorial по обработке деловой информации и пример базы данных Центра хранилищ данных, где устранен ряд ошибок Версии 7.1. Чтобы применить исправленный образец базы данных Центра хранилищ данных, необходимо:

Если вы еще не установили образцы баз данных, создайте новые образцы баз данных с помощью панели запуска Первые шаги. Нажмите кнопку **Пуск** и выберите **Программы —> DB2 IBM —> Первые шаги**.

Если у вас уже установлены баз данных примера, отбросьте баз данных примера DWCTBC, TBC_MD и TBC. Если в базах данных надо сохранить ранее добавленную информацию, прежде чем отбросить базы данных, создайте для нее резервные копии. Чтобы отбросить три баз данных примера:

1. Откройте окно команд DB2, нажав кнопку **Пуск** и выбрав **Программы —> DB2 IBM —> Окно команд**.
2. В окне команд DB2 последовательно введите следующие три команды, нажимая после каждой из них клавишу Enter:

```
db2 drop database dwctbc
db2 drop database tbc_md
db2 drop database tbc
```
3. Закройте командное окно DB2.
4. Создайте новые баз данных примера с помощью панели запуска Первые шаги. Нажмите кнопку **Пуск** и выберите **Программы —> DB2 IBM —> Первые шаги**.

Глава 21. Управляющая база данных хранилища

В этом разделе описаны следующие темы, относящиеся работе с управляющими базами данных хранилища:

- Управляющая база данных хранилища по умолчанию
- Окно Менеджер управляющей базы данных хранилища
- Изменение активной управляющей базы данных хранилища
- Создание и инициализация управляющей базы данных хранилища
- Перенастройка управляющих баз данных IBM Visual Warehouse для использования с Центром хранилищ данных DB2 Версии 7.1.

Управляющая база данных хранилища по умолчанию

При типовой установке DB2 в Windows NT или Windows 2000 система DB2 создает и инициализирует управляющую базу данных хранилища по умолчанию для Центра хранилищ данных, если в реестре Windows NT не указана активная управляющая база данных хранилища. *Инициализация* - это процесс, во время которого Центр хранилищ данных создает управляющие таблицы, необходимые для хранения метаданных Центра хранилищ данных.

Управляющая база данных хранилища по умолчанию имеет имя DWCTRLDB. При регистрации в Центре хранилищ данных по умолчанию в качестве управляющей базы данных хранилища задается DWCTRLDB. Чтобы увидеть имя управляющей базы данных хранилища, которая будет использоваться, нажмите кнопку **Дополнительные** в окне **Регистрация Центра хранилищ данных**.

Окно Менеджер управляющей базы данных хранилища

Окно Менеджер управляющей базы данных хранилища устанавливается при типовой установке DB2 в Windows NT или Windows 2000. В этом окне можно изменить активную управляющую базу данных хранилища, создать и инициализировать новые управляющие базы данных хранилища и перенастроить управляющие базы данных хранилища, которые использовались с IBM Visual Warehouse. Эти действия описаны в последующих разделах.

Перед использованием окна Менеджер управляющей базы данных хранилища остановите сервер хранилища данных.

Изменение активной управляющей базы данных хранилища

Если нужно использовать управляющую базу данных хранилища, не совпадающую с активной управляющей базой данных хранилища, зарегистрируйте эту базу данных в качестве активной с помощью окна Менеджер управляющей базы данных хранилища. Если при регистрации в Центре хранилищ данных задать имя, не совпадающее с именем активной управляющей базы данных хранилища, будет выдано сообщение об ошибке, указывающее, что заданная база данных не совпадает с базой данных, заданной сервером хранилища данных.

Чтобы зарегистрировать базу данных:

1. Выберите **Пуск** → **Программы** → **IBM DB2** → **Менеджер управляющей базы данных хранилища**.

2. В поле **Новая управляющая база** введите имя нужной управляющей базы данных.
3. В поле **Схема** введите имя схемы для этой базы данных.
4. В поле **ID пользователя** введите ID пользователя для обращения к базе данных.
5. В поле **Пароль** введите пароль для данного ID пользователя.
6. В поле **Подтверждение пароля** введите этот пароль еще раз.
7. Нажмите кнопку **ОК**.
Окно останется открытым. В поле **Сообщения** будут появляться сообщения о ходе процесса регистрации.
8. Когда процесс будет завершен, закройте окно.

Создание и инициализация управляющей базы данных хранилища

Если нужно создать управляющую базу данных хранилища, отличающуюся от базы данных по умолчанию, ее можно создать в процессе установки или после установки с помощью окна Менеджер управляющей базы данных хранилища. В процессе установки можно создать базу данных на той же рабочей станции, что и сервер хранилища данных, или на другой рабочей станции.

Чтобы изменить имя управляющей базы данных хранилища, создаваемой во время установки, необходимо выполнить выборочную установку и изменить имя в окне Определить локальную управляющую базу данных хранилища. Процесс установки создаст базу данных с заданным именем, инициализирует ее для использования с Центром хранилищ данных и зарегистрирует ее в качестве активной управляющей базы данных хранилища.

Чтобы в процессе установки создать управляющую базу данных хранилища на рабочей станции, отличающейся от рабочей станции сервера хранилища данных, выберите Локальную управляющую базу данных хранилища во время выборочной установки. Процесс установки создаст эту базу данных. После установки необходимо использовать окно Менеджер управляющей базы данных хранилища на рабочей станции сервера хранилища данных и выполнить шаги, описанные в разделе “Изменение активной управляющей базы данных хранилища” на стр. 183. Задайте имя базы данных, заданное при установке. Эта база данных будет инициализирована для использования с Центром хранилищ данных и зарегистрирована в качестве активной управляющей базы данных хранилища.

Чтобы создать и инициализировать управляющую базу данных хранилища после процесса установки, используйте окно Менеджер управляющей базы данных хранилища на рабочей станции сервера хранилища данных. Если эта новая управляющая база данных хранилища находится не на рабочей станции сервера хранилища данных, ее нужно сначала создать и внести в каталог на рабочей станции сервера хранилища данных. Затем выполните шаги, описанные в разделе “Изменение активной управляющей базы данных хранилища” на стр. 183. Задайте имя базы данных, заданное при установке.

При регистрации в Центре хранилищ данных нажмите кнопку **Дополнительные** и введите имя активной управляющей базы данных хранилища.

Перенастройка управляющих баз данных IBM Visual Warehouse

Информация о перенастройке активной управляющей базы данных хранилища при типовой установке DB2 Universal Database Версии 7.1 в Windows NT и Windows 2000 содержится в руководстве *DB2 Universal Database: Быстрый старт для Windows*. Если нужно перенастроить более одной управляющей базы данных хранилища, для

перенастройке дополнительных баз данных необходимо использовать окно Менеджер управляющей базы данных хранилища. В каждый момент времени активной может быть только одна управляющая база данных хранилища. Если база данных, перенастроенная последней - это не та база данных, которую нужно использовать при следующей регистрации в Центре хранилищ данных, с помощью окна Менеджер управляющей базы данных хранилища зарегистрируйте в качестве активной нужную базу данных.

Доступ к управляющим базам данных хранилища

При типовой установке DB2 Версии 7.1 в Windows NT вместе с сервером хранилища данных создается управляющая база данных хранилища DB2 Версии 7. Если установлена управляющая база данных хранилища Visual Warehouse, перед перенастройкой метаданных этой управляющей базы данных хранилища для использования Центром хранилищ данных DB2 Версии 7.1 необходимо обновить содержащий эту базу данных сервер DB2 в DB2 Версии 7.1. Необходимо перенастроить все управляющие базы данных хранилища, которые нужно использовать в Версии 7.1. Метаданные в активной управляющей базе данных хранилища перенастраиваются в Версию 7.1 во время процесса установки DB2 Версии 7.1. Для перенастройки метаданных в других управляющих базах данных хранилища используйте утилиту перенастройки управляющей базы данных хранилища, для вызова которой в Windows NT выберите **Пуск** → **Программы** → **IBM DB2** → **Менеджер управляющей базы данных хранилища**. Информацию о перенастройке управляющих баз данных хранилища смотрите в руководстве *DB2 Universal Database for Windows: Быстрый старт*.

Глава 22. Начальный комплект OLAP

Web-сайт сервера OLAP

Последние замечания по установке и использованию Начального комплекта OLAP DB2 смотрите на странице Library Web-сайта сервера OLAP DB2:

<http://www.ibm.com/software/data/db2/db2olap/library.html>

Установка Начального комплекта OLAP DB2 в AIX и Solaris

Установка Начального комплекта OLAP DB2 выполняется после основных процедур установки DB2 UDB для UNIX. Программа установки размещает файлы продукта в системном каталоге пользователя root (в AIX: /usr/lpp/db2_07_01; в Solaris: /opt/IBMDB2/V7.1).

Затем на фазе создания экземпляра в начальном каталоге владельца экземпляра в каталоге sqllib будут созданы два каталога OLAP DB2 (essbase и is). На компьютере одновременно может быть запущен только один экземпляр сервера OLAP. Чтобы завершить установку, пользователь должен вручную задать каталог is/bin, чтобы он не был связан с системным каталогом is/bin. Он должен быть связан с каталогом, для которого разрешена запись, в начальном каталоге экземпляра.

Чтобы выполнить установку для Solaris, зарегистрируйтесь с ID экземпляра, перейдите в каталог sqllib/is и затем введите следующие команды:

```
rm bin
mkdir bin
cd bin
ln -s /opt/IBMDB2/V7.1/is/bin/ismesg.mdb ismesg.mdb
ln -s /opt/IBMDB2/V7.1/is/bin/olapicmd olapicmd
ln -s /opt/IBMDB2/V7.1/is/bin/olapisvr olapisvr
ln -s /opt/IBMDB2/V7.1/is/bin/essbase.mdb essbase.mdb
ln -s /opt/IBMDB2/V7.1/is/bin/libolapams.so libolapams.so
```

Регистрация с рабочего стола Сервера интеграции OLAP

Чтобы использовать рабочий стол Сервера интеграции OLAP DB2 для создания моделей и метамакетов OLAP, необходимо установить соединения от программы клиента с двумя серверами: Сервером интеграции OLAP DB2 и сервером OLAP DB2. В окне регистрации будет запрошена необходимая информация для соединения рабочего стола с этими двумя серверами. Слева введите информацию о Сервере интеграции OLAP DB2. Справа введите информацию о сервере OLAP DB2.

Для соединения с Сервером интеграции OLAP DB2:

- Сервер: Введите имя хоста или IP-адрес Сервера интеграции. Если Сервер интеграции установлен на той же рабочей станции, что и рабочий стол, введите значение "localhost" или "127.0.0.1".
- Каталог метаданных OLAP: При соединении с Сервером интеграции OLAP необходимо также задать каталог метаданных. Сервер интеграции OLAP хранит информацию о созданных моделях и метамакетах OLAP в реляционной базе данных, которая называется каталогом метаданных. Эта реляционная база данных должны быть зарегистрирована для ODBC. Эта база данных каталога содержит

специальный набор реляционных таблиц, используемых Сервером интеграции OLAP. В окне регистрации можно задать Сервер интеграции и затем раскрыть выпадающее меню для поля Каталог метаданных OLAP, чтобы увидеть список имен источников данных ODBC, известных на этом Сервере интеграции OLAP. Выберите базу данных ODBC, содержащую таблицы каталога метаданных.

- Имя пользователя и пароль: Сервер интеграции OLAP будет соединяться с каталогом метаданных, используя заданные здесь имя пользователя и пароль. Это учетная запись, существующая на сервере (не на клиенте, если только сервер и клиент не работают на одном компьютере). Это должно быть имя пользователя, создавшего каталог метаданных OLAP. В противном случае Сервер интеграции OLAP не сможет найти нужные реляционные таблицы в базе данных каталога, так как будет использоваться другое имя схемы.

Информацию о сервере OLAP DB2 задавать не обязательно, поэтому можно оставить незаполненными поля в правой части окна Регистрация. Однако для некоторых операций в рабочем столе и менеджере администратора требуется соединение с сервером Server OLAP DB2. Если оставить эти поля пустыми, рабочий стол вновь выведет окно Регистрация, если для выполнения запрошенной операции Серверу интеграции потребуется соединение с сервером OLAP DB2. Рекомендуется всегда заполнять поля сервера OLAP DB2 в окне Регистрация.

Для соединения с сервером OLAP DB2:

- Сервер: Введите имя хоста или IP-адрес сервера OLAP DB2. При работе с Начальным комплектом OLAP DB2 сервер OLAP совпадает с сервером интеграции. Если Сервер интеграции и сервер OLAP установлены на разных хостах, введите имя хоста или IP-адрес, определенные на Сервере интеграции OLAP.
- Имя пользователя и пароль: Сервер интеграции OLAP будет соединяться с сервером OLAP DB2, используя заданные здесь имя пользователя и пароль. Эти имя пользователя и пароль должны быть уже определены на сервере OLAP DB2. Сервер OLAP поддерживает свои имена пользователей и пароли отдельно от операционной системы хоста.

Пример регистрации для Начального комплекта

В следующем примере предполагается, что при установке DB2 UDB 7.1 создан пример OLAP и выбраны *db2admin* в качестве ID пользователя администратора и *password* в качестве пароля администратора.

- Для Сервера интеграции OLAP: Сервер - *localhost*, каталог метаданных OLAP - *TBC_MD*, имя пользователя - *db2admin*, пароль - *password*
- Для сервера OLAP DB2: Сервер - *localhost*, Имя пользователя *db2admin*

Создание и конфигурирование вручную баз данных примера для Сервера интеграции OLAP

Базы данных примера создаются автоматически при установке Сервера интеграции OLAP. Далее описывается, как можно вручную создать базы данных каталога и примера, если это понадобится.

1. В Windows откройте окно Командного центра, выбрав **Пуск → Программы → DB2 for Windows NT → Командное окно**.
2. Создайте базу данных каталога производства:
 - a. Введите команду `db2 create db OLAP_CAT`
 - b. Введите команду `db2 connect to OLAP_CAT`

3. Создайте таблицы в этой базе данных:
 - a. Перейдите в \SQLLIB\IS\ocscript\ocdb2.sql
 - b. Введите команду db2 -tf ocdb2.sql
4. Создайте базу данных источника примера:
 - a. Введите команду db2 connect reset
 - b. Введите команду db2 create db TBC
 - c. Введите команду db2 connect to TBC
5. Создайте таблицы в этой базе данных:
 - a. Перейдите в \SQLLIB\IS\samples\
 - b. Скопируйте tbcdb2.sql в \SQLLIB\samples\db2sampl\tbc
 - c. Скопируйте lddb2.sql в \SQLLIB\samples\db2sampl\tbc
 - d. Перейдите в \SQLLIB\samples\db2sampl\tbc
 - e. Введите команду db2 -tf tbcdb2.sql
 - f. Введите команду db2 -vf lddb2.sql, чтобы загрузить исходные данные примера в эти таблицы.
6. Создайте базу данных каталога примера:
 - a. Введите команду db2 connect reset
 - b. Введите команду db2 create db TBC_MD
 - c. Введите команду db2 connect to TBC_MD
7. Создайте таблицы в этой базе данных:
 - a. Перейдите в \SQLLIB\IS\samples\tbc_md
 - b. Скопируйте ocdb2.sql в \SQLLIB\samples\db2sampl\tbcmd
 - c. Скопируйте lcdb2.sql в \SQLLIB\samples\db2sampl\tbcmd
 - d. Перейдите в \SQLLIB\samples\db2sampl\tbcmd
 - e. Введите команду db2 -tf ocdb2.sql
 - f. Введите команду db2 -vf lcdb2.sql, чтобы загрузить метаданные примера в эти таблицы.
8. Сконфигурируйте ODBC для TBC_MD, TBC, AND OLAP_CAT:
 - a. Откройте Панель управления NT, выбрав **Пуск → Настройка → Панель управления**
 - b. Выберите из списка ODBC (или источники данных ODBC).
 - c. Выберите закладку Системный DSM.
 - d. Нажмите кнопку **Добавить**. Откроется окно **Создать новый источник данных**.
 - e. Выберите из списка IBM DB2 ODBC DRIVER.
 - f. Нажмите кнопку **Завершить**. Откроется окно Драйвер ODBC IBM D2 - Добавить.
 - g. В поле **Имя источника данных** введите имя источника данных (OLAP_CAT).
 - h. В поле **Алиас базы данных** введите алиас или щелкните по стрелке вниз и выберите из списка OLAP_CAT.
 - i. Нажмите кнопку ОК.
 - j. Повторите эти шаги для баз данных TBC_MD и TBC.

Известные проблемы и ограничения

В этом разделе перечислены известные ограничения для Начального набора OLAP DB2, клиента рабочего стола OLAP DB2, клиентов электронных таблиц и Сервера интеграции OLAP DB2

Начальный комплект OLAP DB2

- В настоящее время для Начального комплекта OLAP DB2 поддерживаются следующие платформы: Windows, AIX и операционная среда Sun Solaris.
- Начальный комплект OLAP содержит четыре примера прикладных программ сервера OLAP DB2 с именами Demo, Sampeast, Sample и Samppart. Каждая прикладная программа включает одну или несколько баз данных. В эти базы данных не загружены данные. Чтобы иметь возможность загрузить данные в эти базы данных, необходимо обновить программу до полного сервера OLAP DB2.
- Неправильные окна справки выводятся для:
 - Окна Query Designer **Row** → **Measures** → **Profit**.
 - Окна Cascade Options.
 - Диалогового окна Subset.
 - Клавиша F1 не вызывает правильную информацию справки.
- Некоторые слова или символы не переведены или переведены неправильно для следующих окон:
 - В окне Выходные опции используются английские слова "Default" ("По умолчанию") и "Long Names" ("Длинные имена").
 - В сообщениях в combined.mtx в поля подстановки (%) переменных подставлен английский текст.
 - В Query Designer не выводятся переведенные слова NLV для Ascending (По возрастанию) и Descending (По убыванию), если раскрыть список в поле Сортировка данных в левой панели.
 - В именах объектов баз данных некоторые символы NLV неверны.
 - Файлы журналов содержат некоторые неверные символы NLV.
- Для использования обучающей программы или электронной справки необходимо установить Adobe Acrobat.
- Не поддерживается перенастройка из сервера OLAP DB2 в Начальный комплект OLAP. Начальный комплект OLAP предназначен для пользователей, начинающих использовать OLAP.

Клиент рабочего стола OLAP DB2

- Сервер интеграции OLAP требует, чтобы в модели OLAP была определена ассоциация данных учета с по крайней мере одним измерением. Если используемая модель OLAP создает метамакет, не содержащий ассоциации данных учета с одним или несколькими измерениями, и вы используете закладку Измерения базы данных для создания одного измерения в этом метамакете OLAP, вы сможете сохранить этот метамакет, не получая сообщение об ошибке, но при последующей загрузке члена возникнет ошибка.
- Иногда Сервер интеграции OLAP DB2 выводит номер ошибок Essbase без соответствующих сообщений об ошибках. Обходной путь: Посмотрите файл message.txt, расположенный в каталоге ISHOME/esslib, или файл журнала Сервера интеграции OLAP.
- Рабочий стол Сервера интеграции OLAP DB2 не поддерживает крупные шрифты. Если на компьютере клиента используются разрешение экрана 1024 x 768 или меньше и только крупные шрифты, надписи на кнопках окон Регистрация и Введение рабочего стола Сервера интеграции OLAP будут усечены и ими будет неудобно пользоваться. Обходной путь: На рабочем столе Windows выберите **Пуск** → **Настройка** → **Панель управления** → **Экран** → **Параметры**. Из выпадающего списка Размер шрифта выберите Мелкий шрифт.
- Кнопки обработки на закладках Ассистента моделей OLAP и Ассистента метамакетов OLAP рабочего стола Сервера интеграции OLAP выводятся

неправильно, если не используется цветовая палитра с 65536 цветов. Обходной путь: Чтобы задать подходящую цветовую палитру, на рабочем столе Windows выберите **Пуск → Настройка → Панель управления → Экран → Параметры**. Из выпадающего списка Цветовая палитра выберите High Color (16 разрядов).

- Windows. Если для одного члена используется преобразование даты/времени и фильтр, в Windows NT происходит аварийная остановка сервера DB2. Этой проблемы нет для серверов DB2 в UNIX.
- Рабочий стол Сервера интеграции OLAP поддерживает только имена таблиц длиной от 1 до 30 символов.
- Функция переименования ассоциации ведет себя по-разному в зависимости от того, как она используется. В окне Переименовать для таблицы ассоциации можно использовать имена, содержащие пробелы. В окне Переименовать для свойств ассоциации нельзя использовать имена, содержащие пробелы. Обходной путь: Не используйте пробелы в именах ассоциаций.
- В некоторых средах окно Свойства метамакета не открывается, если метамакет не был перед этим сохранен.
- Если файловая система используемой операционной системы не поддерживает имена длиннее восьми символов, при выполнении функций загрузки и просмотра члена может возникать ошибка номер 2001007.
Чтобы избежать этого, запустите Сервер интеграции OLAP DB2 в каталоге, находящемся в файловой системе, поддерживающей длинные имена.
- После создания метамакета из него нельзя удалять ассоциацию данных учета. Если нужно изменить измерения в ассоциации данных учета, удалите все существующие в метамакете измерения и затем создайте новые измерения.
- Опция Просмотр данных таблицы в стандартном пользовательском интерфейсе модели OLAP, позволяющая просматривать данные из таблицы реляционного источника в левой части главного окна Модель OLAP, может выводить только 1000 строк. В окне выводится 100 строк источника данных, чтобы увидеть следующие 100 строк, нажмите кнопку **Далее**. На экран можно вывести не более 1000 первых строк.

Клиенты электронных таблиц

- Если в имени электронной таблицы Lotus 123 или в самой электронной таблице используются символы NLV, из этой электронной таблицы нельзя будет установить соединение, получить данные или выполнить какие-либо другие функции.
- Быстрый двойной щелчок мыши по клетке встраиваемого модуля для электронных таблиц Lotus 123 может привести к следующему сообщению об ошибке: *Microsoft Visual C++ Runtime error Program c:\lotus\123\123w.exe abnormal program termination*. Обходной путь: Щелкайте мышью не так быстро, чтобы раскрывать клетки по одной.
- Если в Lotus 123 одновременно выбрать две строки и затем выбрать Essbase - Оставить только или Удалить только, будет оставлен или удален только одна строка.
- В Lotus 123 возникает ошибка вычисления Essbase, если выбрано несколько строк.
- AIX. Прикладные программы примера не содержат данных. Можно использовать программы электронных таблиц для получения информации из базы данных, но будут показаны только имена ассоциаций и членов. Для некоторых значений будет указано, что они отсутствуют.
- Если нажать стрелку вниз в окне просмотра связанных объектов, работа электронных таблиц завершается аварийно и выдается сообщение об ошибке "DR Watson". Правильное сообщение об ошибке не выводится.

- Если в Lotus 123 не открыта рабочая тетрадь и вы выбираете пункт меню Сервер OLAP, должно выводиться сообщение об ошибке. Но это сообщение не выводится и вы услышите два предупреждающих звуковых сигнала.
- Если в Excel дважды щелкнуть по формулам в электронной таблице, эти формулы будут удалены из электронной таблице. Задание опции "Сохранять при увеличении" не предотвращает эту ошибку.

Сервер интеграции OLAP DB2

- В настоящее время для функций составления расписания поддерживается только английский язык. В средах с другими языками использование функции **Инструмент** → **Планировщик** в стандартном пользовательском интерфейсе метамакета OLAP на компьютере клиента NT приводит к аварийной остановке Сервера интеграции OLAP. Это происходит потому, что планировщик NT хранит информацию расписания в формате, зависящем от языка, а Сервер интеграции OLAP не может работать с такой информацией.
- Сервер интеграции OLAP DB2 не поддерживает типы данных GRAPHIC, VARGRAPHIC и LONG VARGRAPHIC при операциях просмотра таблицы в стандартном пользовательском интерфейсе модели OLAP. Если в модели OLAP выбрана операция Просмотр таблицы, вместо реляционных таблиц, содержащих тип данных GRAPHIC, VARGRAPHIC или LONG VARGRAPHIC, будут выводиться пробелы. Обходной путь: В файл DB2CLI.INI добавьте следующее: [SAMPLE] PATCH1=65536 < PATCH2=7 < DBALIAS=SAMPLE
- Если Сервер интеграции OLAP обнаруживает пустое значение (NULL) при загрузке данных, он автоматически загружает данные в родительский член этого пустого значения. Однако если это пустое значение поколения 2, Сервер интеграции OLAP не может загрузить данные в родительский член, поскольку этот родительский член - это член уровня ассоциации. В этом случае Сервер интеграции OLAP записывает в файл журнала сообщения об ошибках. Обходной путь: Не включайте пустые значения в поколения 2.
- Сервер интеграции OLAP не поддерживает имена столбцов реляционной СУБД (RDBMS), содержащие пробелы. Если в них есть пробелы, Сервер интеграции OLAP сгенерирует неправильные операторы SQL.
- Сервер интеграции OLAP не воспринимает некоторые символы набора двухбайтных символов (DBCS), такие как символ минуса (-), при получении значений из реляционного источника данных во время операции предварительного просмотра. Если Сервер интеграции OLAP обнаруживает такие символы во время операции предварительного просмотра, выводится сообщение об ошибке "Непредвиденная ошибка в условии".
- Если в конкатенации столбцов в модели OLAP используется двухбайтный символ знака минус (-), при операциях загрузке членов генерируются сообщения о синтаксических ошибках. Обходной путь: При выполнении преобразований для столбцов в модели OLAP не используйте в имени столбца символы минус, дефис или тире (-). Не используйте реляционные таблицы или столбцы реляционных таблиц, имена которых содержат символы минус, дефис или тире (-).
- AIX. При запуске программы примера вы увидите сообщение о том, что сервер не поддерживает преобразование значений валют.
- Windows: При установке создаются две базы данных каталога примера для использования с Сервером интеграции OLAP DB2. Однако при попытке зарегистрироваться в одной из этих баз данных из рабочего стола Сервера интеграции OLAP DB2 выдается сообщение об ошибке CLI "Неправильный атрибут строки соединения". Обходной путь: Измените файл db2cli.ini, расположенный в каталоге sqllib, выполнив следующие действия:
 1. Сделайте резервную копию файла db2cli.ini.

- Из каждого раздела удалите 'DATABASE=OLAPCATP' и 'DATABASE=OLAPCATD'.

```
[OLAPCATP]
DATABASE=OLAPCATP <----- (удалить)
DESCRIPTION=OLAPCATP
DBALIAS=OLAPCATP
```

```
[OLAPCATD]
DATABASE=OLAPCATD <----- (удалить)
DESCRIPTION=OLAPCATD
DBALIAS=OLAPCATD
```

- Сервер интеграции Hyperion не добавляет описание фильтра метамакета в ассистенте метамакетов OLAP.
- Чтобы обойти ошибку составления расписания в английской версии среды AIX, используйте следующий способ.
Для всех новых серверов AIX добавьте <имя_пользователя> в команду /var/adm/cron/cron.allow, чтобы разрешить составление расписания для указанного пользователя. В каталоге var/spool/crontabs создайте пустой файл с именем <имя_пользователя> для этого пользователя и задайте в для него разрешения на доступ, равные 555. Аналогичная настройка требуется для других сред UNIX; подробную информацию смотрите на странице описания crontab.
- Сервер интеграции OLAP не распознает двухбайтные символы пробела. Следующие параметры пространства преобразования не работают для столбцов источника данных, двухбайтные символы пробела:
 - Отбрасывание начальных/конечных пробелов
 - Преобразование пробелов в символы подчеркивания
 - Конкатенация
- Сервер интеграции OLAP не может сохранить описание ассоциации в ассистенте моделей OLAP, если это описание содержит двухбайтные символы.
- Сервер интеграции OLAP не поддерживает сквозные преобразования (зависящие от базы данных) с использованием функций SQL. При задании встроенных функций реляционной СУБД, таких как Substring или Left, Сервер интеграции OLAP генерирует неверные операторы SQL.
- При создании иерархий в ассистенте моделей OLAP Сервер интеграции OLAP создает пустую папку в папке Hyperion\IS\Loadinfo на клиенте. Эта пустая папка содержит пустой файл .txt. Пустые папки и файлы также создаются при вызове просмотра примера из окна Редактирование иерархии или вызове окна предварительного просмотра результатов из окна Редактирование модели OLAP в стандартном пользовательском интерфейсе.
Чтобы избежать накопления пустых папок и файлов, их можно удалить из папки Loadinfo в любое время.
- Сервер интеграции OLAP не выводит сообщение для подтверждения удаления таблицы ассоциации из модели OLAP.
Предлагаемый обходной путь: Если после добавления таблицы ассоциации, которую нужно удалить, модель OLAP не была сохранена, нажмите кнопку **Заккрыть**, чтобы закрыть модель OLAP без сохранения изменений, и затем вернитесь к предыдущей версии этой модели. Все другие изменения, сделанные во время текущего сеанса, будут также утеряны.
- Сервер интеграции OLAP не поддерживает сценарии Essbase ESSCMD. Каталог IS\esscript не был удален из структуры каталогов Сервера интеграции OLAP, создаваемой в процессе установки. Это пустой каталог, он не используется.

Для модуля электронных таблиц Начального комплекта OLAP требуется последнее исправление Windows

Перед установкой сервера OLAP DB2 в Windows NT необходимо применить служебное исправление 5 для MS Windows NT 4.0.

Если возникают ошибки при установке встраиваемого модуля Начального комплекта OLAP для электронных таблиц в Windows 95 или Windows 98, они могут быть вызваны устаревшими уровнями файлов системы Microsoft. Загрузите следующие файлы от Microsoft с помощью служебного пакета Windows 95/98 или распакуйте архив %arborpath%\bin\olapewd.zip и скопируйте эти файлы в системный каталог Windows. Не заменяйте при этом те файлы вашей системы, даты выпуска которых новее дат файлов, предлагаемых для замены. Системные файлы Windows 9x и необходимые их уровни:

- ASYCFILT.DLL 2.20.4118.1
- COMCAT.DLL 4.71.1441.1
- COMPOBJ.DLL 2.10.35.35
- DCOMCNFG.EXE 4.0.1381.4
- DLLHOST.EXE 4.0.1381.4
- IPROP.DLL 4.0.1381.4
- OLE2.DLL 2.10.35.35
- OLEAUT32.DLL 2.20.4118.1
- OLECNV32.DLL 4.0.1381.4
- OLEDLG.DLL 4.0.1381.4
- OLEPRO32.DLL 5.0.4118.1
- OLETHK32.DLL 4.0.1371.1
- RPCLTC1.DLL 4.0.1381.4
- RPCLTCCM.DLL 4.0.1381.4
- RPCLTSCM.DLL 4.0.1381.4
- RPCMQCL.DLL 4.0.1381.4
- RPCMQSVR.DLL 4.0.1381.4
- RPCNS4.DLL 4.0.1371.1
- RPCSS.EXE 4.0.1381.4
- STDOLE2.TLB 2.20.4122.1
- STDOLE32.TLB 2.10.3027.1
- STORAGE.DLL 2.10.35.35

Отсутствие файлов EQD встраиваемого модуля OLAP для электронных таблиц

В Начальном комплекте OLAP дополнительный модуль для электронных таблиц содержит компонент, называемый Query Designer (EQD). Меню электронной справки по EQD содержит кнопку **Учебник**, которая не выводит на экран никакой информации. Материал, который должен выводиться в учебниках по EQD, содержится во второй главе руководств *OLAP Spreadsheet Add-in User's Guide for Excel* и *OLAP Spreadsheet Add-in User's Guide for 1-2-3*. Вся информация учебника по EQD доступна в версиях HTML этих книг в Информационном центре и в версиях PDF.

Поддержка ассоциаций атрибутов

В начальный комплект OLAP DB2 теперь включена поддержка ассоциаций атрибутов. Теперь для анализа данных атрибутов в базе данных OLAP можно создавать ассоциации атрибутов в моделях и метамакетах OLAP. Доступны:

- Содержательные сводки данных, использующие атрибуты посредством создания отчетов перекрестных таблиц. Отчеты перекрестных таблиц позволяют выводить сводки данных на основе нескольких характеристик.
- Доступ к пяти сводным характеристикам всех данных атрибутов: суммам, количеству, а также средним, минимальным и максимальным значениям.
- Четыре типа атрибутов, позволяющие селективно просматривать необходимые сравнения данных: текстовый, численный, логический, и на основании даты.
- Использование численных атрибутов для группировки и сводки данных по диапазонам или значениям.

Ассоциации и члены атрибутов рассчитываются только динамически, что означает, что данные атрибута не сохраняются в базе данных OLAP, что, в свою очередь, приводит к меньшим макетам. Кроме того, во время считывания пользователи могут выбирать, просматривать ли данные атрибутов, что приводит к большей возможности выбора и гибкости в определении вариантов отчетов OLAP на основе потребностей.

Книги, в которые были внесены изменения для нового Начального комплекта OLAP DB2

Содержание следующих книг для Начального комплекта OLAP DB2 было изменено:

- OLAP Integration Server Administration Guide
- OLAP Integration Server Model User's Guide
- OLAP Integration Server Metaoutline User's Guide

Эти книги доступны в Интернете по адресу:

<http://www.ibm.com/software/data/db2/db2olap/library.html>

Глава 23. Менеджер хранилищ данных DB2

"Менеджер хранилищ данных" следует читать как "Менеджер хранилищ данных DB2"

Все словосочетания "Менеджер хранилищ данных" на экране и в документации следует читать как "Менеджер хранилищ данных DB2".

Утилита инициализации менеджера каталогов данных

В утилите инициализации менеджера каталогов данных (ICM) теперь можно добавить оператор SQL к концу оператора CREATE TABLE при помощи следующей команды:

На платформах UNIX:

```
CREATEIC /DBTYPE тип-базы-данных /DGNAME имя-базы-данных /USERID id-пользователя /PASSWORD пароль /KA1 id-пользователя /TABOPT "каталог:\tabopt.file"
```

На платформах Windows:

```
CREATEIC \DBTYPE тип-базы-данных \DGNAME имя-базы-данных \USERID id-пользователя \PASSWORD пароль \KA1 id-пользователя \TABOPT "каталог:\tabopt.file"
```

Ключевое слово TABOPT можно задать в утилите CREATEIC из каталога установки DB2. За ключевым словом TABOPT должно идти полное имя файла tabopt.file. Если имя каталога содержит пробелы, заключите его в кавычки. Содержимое файла tabopt.file должно содержать информацию, добавляемую к оператору CREATE TABLE. Вы можете использовать любой из операторов SQL ниже для записи в этот файл tabopt.file. Утилита ICM прочтет этот файл и добавит его к оператору CREATE TABLE.

Таблица 6. Операторы SQL

IN MYTABLESPACE	Создает таблицу с ее данными в табличном пространстве MYTABLESPACE
DATA CAPTURE CHANGES	Создает таблицу и записывает изменения SQL в расширенном формате
IN ACCOUNTING INDEX IN ACCOUNT_IDX	Создает таблицу с ее данными в табличном пространстве ACCOUNTING, а индекс - в пространстве ACCOUNT_IDX

Максимальный размер содержимого - 1000 однобайтных символов.

Эта новая возможность доступна только в системах Windows и UNIX.

Вопросы лицензирования

Если вы получили следующее сообщение:

```
FLG0083E: У вас нет действительной лицензии для утилиты инициализации Менеджера каталогов данных IBM. Пожалуйста, обратитесь к местному дилеру или торговому представителю IBM.
```

Вы должны приобрести Менеджер хранилищ данных DB2 или сервер OLAP IBM DB2 и установить компонент Менеджер каталогов данных, содержащий утилиту инициализации каталога данных.

Вопросы установки

Если вы установили Менеджер хранилищ данных DB2 или сервера OLAP DB2 IBM, а затем установили на той же рабочей станции другой компонент администратора менеджера каталогов данных (с компакт-диска DB2 Universal Database), утилита инициализации каталога данных может быть перезаписана. В этом случае в каталоге `\sqlib\bin` найдите файлы `createic.bak` и `flgnmwcr.bak` и переименуйте их соответственно в `createic.exe` и `flgnmwcr.exe`.

Если вы устанавливаете дополнительные компоненты Менеджера каталогов данных DB2 Universal Database, эти компоненты должны быть установлены на отдельной рабочей станции (а не на той, где установлен Менеджер хранилищ данных). Дополнительную информацию смотрите в Главе 3, *Installing Information Catalog Manager* (Установка Менеджера каталогов данных) руководства DB2 Warehouse Manager Installation Guide.

Менеджер каталогов данных для Web

При использовании каталога данных, находящегося в системе DB2 UDB for OS/390, поиск без учета регистра недоступен. Это относится и к простому поиску, и к сложному поиску. Электронная справка по простому поиску не поясняет, что все виды поиска в каталоге данных DB2 UDB for OS/390 производятся с учетом регистра. Далее, любой объект категории группировки можно развернуть, даже если у него нет базовых объектов.

Information Catalog Manager Administration Guide

Доступ к каталогам данных DB2 Версии 5 при помощи менеджера каталогов данных DB2 Версии 7.1

Подкомпоненты менеджера каталогов данных DB2 Версии 7.1, сконфигурированные процессом установки DB2 Версии 7.1, поддерживают доступ к каталогам данных, сохраненным в базах данных DB2 Версии 6 и DB2 Версии 7.1. Конфигурацию этих подкомпонентов можно изменить, чтобы обращаться к каталогам данных, сохраненным в базах данных DB2 Версии 5. Подкомпоненты менеджера каталогов данных DB2 Версии 7.1 не поддерживают доступ к данным из DB2 Версии 2 или более ранних версий.

Чтобы настроить администратора каталога данных, пользователя каталога данных и утилиту инициализации каталога данных для доступа к каталогам данных, сохраненным в базах данных DB2 Версии 5:

1. Установите DB2 Connect Enterprise Edition Версии 6 на рабочей станции, отличной от той, где установлен менеджер каталогов данных DB2 Версии 7.1.

DB2 Connect Enterprise Edition входит в DB2 Universal Database Enterprise Edition и DB2 Universal Database Enterprise - Extended Edition. Если установлена Версия 6 одного из этих продуктов, не нужно отдельно устанавливать DB2 Connect.

Ограничение: На одной рабочей станции Windows NT или OS/2 нельзя установить несколько версий DB2. Можно установить DB2 Connect на другой рабочей станции Windows NT или на рабочей станции OS/2 или UNIX.

2. Сконфигурируйте менеджер каталогов данных и DB2 Connect Версии 6 для доступа к данным DB2 Версии 5. Дополнительную информацию смотрите в руководстве *Руководство пользователя DB2 Connect*. Далее представлен обзор требуемых шагов:

- a. В системе DB2 Версии 5 используйте процессор командной строки DB2 для внесения в каталог базы данных Версии 5, к которой будет обращаться менеджер каталогов данных.
- b. В системе DB2 Connect используйте процессор командной строки DB2 для внесения в каталог:
 - Узла TCP/IP для системы DB2 Версии 5
 - Базы данных для системы DB2 Версии 5
 - Записи DCS для системы DB2 Версии 5
- c. На рабочей станции менеджера каталогов данных используйте процессор командной строки DB2 для внесения в каталог:
 - Узла TCP/IP для системы DB2 Connect
 - Базы данных для системы DB2 Connect

Информацию о внесении в каталог баз данных смотрите в руководстве *Дополнение по установке и конфигурированию DB2 Universal Database*.

3. На рабочей станции менеджера каталогов данных свяжите пакет CLI DB2 с каждой базой данных, для обращения к которой будет использоваться DB2 Connect.

Далее показан пример команд DB2 для связывания с базой данных DB2 Версии 5 с именем v5database. Для ввода этих команд используйте процессор командной строки DB2. Файлы db2cli.lst и db2ajgrt находятся в каталоге \sqllib\bnd.

```
db2 connect to v5database user ID_пользователя using пароль
db2 bind db2ajgrt.bnd
db2 bind @db2cli.lst blocking all grant public
```

где *ID_пользователя* - ID пользователя для v5database и *пароль* - пароль для этого ID пользователя.

При связывании db2cli.list с базой данных DB2 Версии 5 возникает ошибка. Эта ошибка вызывается тем, что в этой конфигурации не поддерживаются большие объекты. Эта ошибка не влияет на доступ агента хранилища данных к базе данных DB2 Версии 5.

Для обращения к данным DB2 Версии 5 через DB2 Connect требуется FixPak 14 для DB2 Universal Database Версии 5, доступный с июня 2000 года. Номер APAR для этого FixPak - JR14507.

Конфигурирование каталога данных

Про шаг 2 в первом разделе Главы 1, "Setting up an information catalog" (Конфигурирование каталога данных), говорится:

```
When you install either the DB2 Warehouse Manager
or the DB2 OLAP Server, a default information catalog
is created on DB2 Universal Database for Windows NT.
```

Это утверждение неверно. Вам надо определить новый каталог данных. Дополнительную информацию смотрите в разделе "Creating the Information Catalog" (Создание каталога данных).

Обмен метаданными с другими продуктами

В Главе 6 "Exchanging metadata with other products" (Обмен метаданными с другими продуктами) в разделе "Identifying OLAP objects to publish" (Идентификация объектов OLAP для публикаций) во втором абзаце утверждается:

When you publish DB2 OLAP Integration Server metadata, a linked relationship is created between an information catalog "dimensions within a multi-dimensional database" object type and a table object in the OLAP Integration Server.

Это утверждение следует читать так:

When you publish DB2 OLAP Integration Server metadata, a linked relationship is created between an information catalog "dimensions within a "ассоциация в многомерной базе данных" и таблицей.

Это же утверждение появляется в Приложении С, "Metadata mappings" (Отображение метаданных) в разделе "Metadata mappings between the Information Catalog Manager and OLAP Server" (Отображение метаданных между Менеджером каталогов данных и сервером OLAP).

Обмен метаданными при помощи командами flgnxoln

В Главе 6 "Exchanging metadata" (Обмен метаданными) есть раздел "Identifying OLAP objects to publish" (Идентификация объектов OLAP для публикации). В конце этого раздела есть пример использования команды flgnxoln для публикации метаданных сервера OLAP через каталог данных. В этом примере каталог файлов db2olap.ctl db2olap.ff ошибочно указан как x:\Program Files\sqllib\logging. Правильное имя этого каталога - x:\Program Files\sqllib\exchange, как сказано на странице 87.

Обмен метаданными при помощи команды MDISDGC

Глава 6. "Exchanging metadata with other products" (Обмен метаданными с другими продуктами), раздел "Converting MDIS-conforming metadata into a tag language file" (Конвертирование MDIS-совместимых метаданных в файл языка тегов), страница 97. Команду MDISDGC нельзя ввести в командном окне MS-DOS. Эту команду надо вводить в командном окне DB2. В первом предложении раздела "Converting a tag language file into MDIS-conforming metadata" (Конвертирование MDIS-совместимых метаданных в файл языка тегов) тоже говорится, что команду DGMDISC надо вводить из командной строки MS-DOS. Команду DGMDISC надо вводить в командном окне DB2.

Вызов программ

В некоторых примерах руководства Information Catalog Administration Guide есть команды, содержащие каталог Program Files. Вызывая команду, в имя которой входит Program Files, надо заключить строку вызова в двойные кавычки. Например, в Приложении В, "Predefined Information Catalog Manager object types" (Предопределенные типы объектов Менеджера каталогов данных) в разделе "Initializing your information catalog with the predefined object types" (Инициализация каталога данных с предопределенными типами объектов) приводится пример. Если использовать его из командной строки DOS буквально, вы получите сообщение об ошибке. Правильный текст примера такой:

```
"X:\Program Files\SQLLIB\SAMPLES\SAMPDATA\DGWDEMO"  
/T userid password dname
```

Information Catalog Manager Programming Guide and Reference

Коды причин Менеджера каталогов данных

В Приложении D: Information Catalog Manager reason codes в крайнем правом столбце может не полностью выводиться текст для следующих кодов причин: 31014, 32727, 32728, 32729, 32730, 32735, 32736, 32737, 33000, 37507, 37511 и 39206. Если текст усечен, смотрите HTML-версию этой книги, где он приводится полностью.

Information Catalog Manager User's Guide

В главе 2 есть раздел под названием "Registering a server node and remote information catalog" (Регистрация узла сервера и удаленного каталога данных). В этом разделе перечислены шаги, которые можно выполнить из Центра управления DB2 перед регистрацией удаленного каталога данных с использованием Менеджера каталогов данных. В последнем абзаце этого раздела сказано, что после выполнения последовательности шагов из Центра управления DB2 (добавить систему, добавить экземпляр и добавить базу данных) Центр управления необходимо отключить, и только после этого открывать Менеджер каталогов данных. Это неверно. Отключать Центр управления перед открытием Менеджера каталогов данных не обязательно.

То же исправление касается раздела оперативной справки "Регистрация узла сервера и удаленного каталога данных" и оперативной справки по окну Регистрация узла сервера и каталога данных.

Менеджер каталогов данных: Сообщения

Сообщение FLG0260E

Второе предложение в объяснении этого сообщения должно быть таким:

Эта ошибка вызвана ошибкой отката каталога данных.
Каталог данных не находится в стабильном состоянии, но изменения не были внесены.

Сообщение FLG0051E

Второй абзац в объяснении этого сообщения должен быть таким:

Каталог данных содержит слишком много объектов или типов объектов.

В описании действий администратора должно быть сказано:

Удалите из текущего каталога данных некоторые объекты
или типы объектов с помощью функции импорта.

Сообщение FLG0003E

В объяснении этого сообщения должно быть сказано:

Чтобы каталог данных можно было использовать, он должен быть зарегистрирован.
Возможно, этот каталог данных не был правильно зарегистрирован.

Сообщение FLG0372E

Первое предложение в объяснении этого сообщения должно быть таким:

Проигнорировано значение ATTACHMENT-IND для объекта,
поскольку это объект подключения.

Сообщение FLG0615E

Второе предложение в этом сообщении должно быть таким:

Менеджер каталогов данных обнаружил непредвиденную
ошибку в базе данных или не нашел файл связывания в текущем каталоге или пути.

Менеджер каталогов данных: Электронная справка

Окно каталога данных: В электронной справке по пункту Открыть меню Выбранный ошибочно сказано: "Открывает выбранный объект". Там должно быть сказано: "Открывает окно Определить поиск".

Руководство администратора Query Patroller

Клиент DB2 Query Patroller - это отдельный компонент

Клиент DB2 Query Patroller - отдельный компонент, не входящий в клиент управления DB2. Это значит, что он не устанавливается при установке клиента администратора DB2, как сказано в руководстве Query Patroller Installation Guide. Клиент Query Patroller надо устанавливать отдельно.

Установка Query Patroller вручную в AIX и Solaris

Чтобы установить DB2 Query Patroller при помощи `installp` или `smit`, выполните описанные ниже действия. Подробный синтаксис и информацию о параметрах смотрите в разделе “Команды установки вручную” на стр. 204.

1. Создайте или сконфигурируйте экземпляр DB2 UDB EEE или EE для использования с DB2 Query Patroller.
2. Добавьте в файл `etc/services` запись для использования с сервером DB2 Query Patroller. Например, `dqp1 55000/TCP`.
3. Создайте пользователя с именем `iwm`, если это еще не сделано.
4. Смонтируйте компакт-диск.
5. Перейдите в каталог `/cdrom/db2`.
- 6.

- **Агент**

- a. Если вы устанавливаете агент Query Patroller на AIX, при помощи `smit` установите следующие наборы файлов:

- 1) `db2_07_01.dqr.cln`
- 2) `db2_07_01.mlic`
- 3) `db2_07_01.dqr.agt`

Примечание: Эти наборы надо устанавливать в указанном порядке. Если для установки этих наборов вы не используете `smit`, соблюдайте указанный порядок. Кроме того, если при настройке экземпляра DB2 EE или EEE не были установлены наборы файлов `db2_07_01.cj` и `db2_07_01.jdbc`, нужно установить их перед началом установки агента Query Patroller.

- b. В системе Solaris используйте `pkgadd` для установки следующих пакетов для сервера DB2 Query Patroller:

- 1) `db2qpc71`
- 2) `db2mlic71`
- 3) `db2dqp71`

Примечание: Эти пакеты надо устанавливать в указанном порядке. Кроме того, если при настройке экземпляра DB2 UDB EE или EEE не были установлены наборы файлов `db2cj71` и `db2jdbc71`, нужно установить их перед началом установки агента Query Patroller.

- **Сервер**

- a. Чтобы установить сервер DB2 Query Patroller в AIX, установите набор файлов `db2_07_01.dqr.agt` и необходимые для него наборы файлов, перечисленные выше. Затем установите набор файлов `db2_07_01.dqr.srv`.
- b. Чтобы установить сервер DB2 Query Patroller на Solaris, установите набор файлов `db2dqp71` и необходимые для него наборы, перечисленные выше. Затем установите пакет `db2dqs71`.

При выполнении перенастройки с Версии 6 на Версию 7.1 обратитесь к руководству *DB2 Query Patroller Installation Guide*.

Установив сервер, сконфигурируйте лицензию:

1. Добавьте пользователя `iwm` к первичной группе владельца экземпляра DB2 UDB EE или EEE. Это даст пользователю `iwm` полномочия `SYSADM` для этого экземпляра.
2. Добавьте в файл `.profile` пользователя `iwm` следующие строки. Здесь *INSTHOME* - домашний каталог экземпляра сервера DB2 Query Patroller.
`.INSTHOME/sqllib/db2profile`

Примечание: Если используется оболочка C, добавьте в файл `.login` строку
`source /sqllib/db2cshrc`

3. Зарегистрируйтесь как пользователь `root` и введите команду:
При создании экземпляра сервера Query Patroller должен быть установлен на главном узле DB2 UDB EE или DB2 UDB EEE. Чтобы установить сервер Query Patroller:

- a. Введите следующую команду:

```
dqpcrt -s -p имя_порта имя_экземпляра
```

имя_порта - это имя порта, которое использовалось на шаге 2.

имя_экземпляра - это имя экземпляра DB2 UDB EE или EEE. Подробный синтаксис и информацию о параметрах смотрите в разделе “Команды установки вручную” на стр. 204.

Примечание: Чтобы удалить экземпляр `dqr`, можно ввести команду **dqdrop** *имя_экземпляра*. Эту команду можно запустить только на том узле, где установлен сервер.

- b. Зарегистрируйтесь под именем экземпляра. Выполните команду:

```
dqpsetup -D имя_базы_данных -g имя_группы_узлов -n номер_узла -t  
имя_табличного_пространства -r имя_табличного_пространства_результатов  
-l путь_табличного_пространства
```

Подробный синтаксис и информацию о параметрах смотрите в разделе “Команды установки вручную” на стр. 204.

Чтобы установить агент Query Patroller, введите:

```
dqpcrt -a -p имя_порта имя_экземпляра
```

имя_порта - это имя порта, которое использовалось на шаге 2. *имя_экземпляра* - это имя экземпляра DB2 UDB EE или EEE. Подробный синтаксис и информацию о параметрах смотрите в разделе “Команды установки вручную” на стр. 204.

4. Зарегистрируйте DB2 Query Patroller при помощи команды `db2i` см. Дальнейшую информацию смотрите в справочнике *DB2 Command Reference*.

Создание схемы Query Patroller и связывание файлов связывания программ

Чтобы вручную создать схему DB2 Query Patroller и связать все файлы связывания программ, выполните следующие действия:

1. Создайте табличное пространство DB2, которое будет использоваться для схемы DB2 Query Patroller. Это табличное пространство должно быть создано на одной группе узлов.

- Создайте схему при помощи программы `db2_qr_schema` в каталоге `DB2 bin`. Эта программа использует в качестве входного файл сценария `iwms_schema.sql`. `db2_qr_schema` поддерживает два варианта синтаксиса:


```
db2_qr_schema <имя_входного_файла_схемы> <алиас_базы_данных>
<пользователь> <пароль> <табличное_пространство_querypatroller>
db2_qr_schema <имя_входного_файла_схемы> <алиас_базы_данных>
<табличное_пространство_querypatroller>
```
- Свяжите файлы связывания сервера DB2 Query Patroller при помощи списка файлов связывания `db2qr.lst` в каталоге `DB2 bnd`. Соединившись с базой данных, введите в командной строке команду DB2:


```
db2 bind @db2qr.lst blocking all grant public
```
- Выполните команду:


```
db2 bind iwmsx001.bnd isolation ur blocking all grant public insert buf
datetime iso
```
- Свяжите файлы связывания хранимых процедур DB2 Query Patroller при помощи списка файлов связывания `db2qr_sp.lst` в каталоге `DB2 bnd`. Соединившись с базой данных, введите в командной строке команду DB2:


```
db2 bind @db2qr_sp.lst blocking all
```
- Создайте табличное пространство для таблиц результатов DB2 Query Patroller.

Команды установки вручную dpcrt

Следующая команда используется для назначения узла системы DB2 UDB EE или DB2 UDB EEE в качестве сервера DB2 Query Patroller. Имя порта для использования экземпляром DB2 Query Patroller и имя экземпляра DB2 UDB EE или EEE для использования в качестве сервера DB2 Query Patroller указывать обязательно. Синтаксис:

```
►► dpcrt -s -p имя_порта имя_экземпляра
        -a -p имя_порта имя_экземпляра
        -h
```

Таблица 7. Параметры команды `dpcrt`

Параметр	Описание
-s	Используется для создания сервера DB2 Query Patroller server на указанном экземпляре DB2 UDB EE или EEE.
-a	Используется для создания агента DB2 Query Patroller agent на указанном экземпляре DB2 UDB EE или EEE.
<i>имя_порта</i>	Задаёт имя порта для использования с сервером или агентом DB2 Query Patroller.
<i>имя_экземпляра</i>	Задаёт имя экземпляра DB2 UDB EE или EEE, который будет использоваться как экземпляр сервера DB2 Query Patroller.
-h	Выводит информацию об использовании команды.

dqpsetup

Эта команда используется для задания параметров сервера DB2 Query Patroller. Параметр *размер_DMS* и флаг *-o* задавать не обязательно. Флаг *-o* можно использовать для удаления объектов схемы из ранее установленной версии продукта. Синтаксис:

► dqpsetup [параметры установки] h

параметры установки:

► -d *имя_базы_данных* -g *имя_группы_узлов* -n *номер_узла*

► -t *имя_табличного_пространства* -r *имя_табличного_пространства_результатов*

► -l *путь_табличного_пространства* -s *размер_DMS* -o

► *имя_экземпляра*

Таблица 8. Параметры команды *dqpsetup*

Параметр	Описание
-d <i>имя_базы_данных</i>	Имя базы данных, которая будет использоваться с сервером DB2 Query Patroller.
-g <i>имя_группы_узлов</i>	Имя группы узлов, где находится табличное пространство для сервера DB2 Query Patroller.
-n <i>номер_узла</i>	Номер отдельного узла, где определена группа узлов.
-t <i>имя_табличного_пространства</i>	Имя табличного пространства DB2 Query Patroller. Его тип по умолчанию - табличное пространство SMS.
-r <i>имя_табличного_пространства_результатов</i>	Имя используемого табличного пространства результатов.
-l <i>путь_табличного_пространства</i>	Полное имя пути для табличного пространства.
-s <i>размер_DMS</i>	Размер табличного пространства DMS. Флаг -s используется для задания размера табличного пространства DMS. Этот параметр необязателен; он задается, только если используется табличное пространство DMS. По умолчанию используется табличное пространство SMS.
-o	Переопределяет все прежние объекты схем DB2 Query Patroller. Это необязательный параметр.
<i>имя_экземпляра</i>	Имя экземпляра DB2 UDB EE или EEE, который будет использоваться как экземпляр сервера DB2 Query Patroller.

Таблица 8. Параметры команды *dqpssetup* (продолжение)

Параметр	Описание
-h	Выводит информацию об использовании команды.

dqplist

Эта команда используется для определения имени экземпляра DB2 UDB EE или DB2 UDB EEE для использования в качестве сервера DB2 Query Patroller. Эту команду можно вводить только на узле, на котором создан сервер DB2 Query Patroller. Синтаксис:

```
►► dqplist -h
```

Флаг **-h** выводит информацию об использовании команды.

dqpdrop

Эта команда используется для отбрасывания существующего экземпляра сервера DB2 Query Patroller. Ее можно вводить только на узле, на котором создан сервер DB2 Query Patroller. Синтаксис:

```
►► dqpdrop имя_экземпляра -h
```

Флаг **-h** выводит информацию об использовании. Параметр *имя_экземпляра* задает имя экземпляра DB2 Query Patroller, который вы хотите отбросить.

Перенастройка из Версии 6 DB2 Query Patroller с использованием dqpmigrate

Команду *dqpmigrate* необходимо использовать, если Версия 7 сервера Query Patroller установлена поверх Версии 6. Для FixPak 2 или более позднего нет необходимости запускать *dqpmigrate* вручную, поскольку FixPak запускает эту команду автоматически. Если не использовать эту команду, у существующих пользователей, определенных в Версии 6, не будет привилегий EXECUTE для нескольких новых хранимых процедур, добавленных в Версии 7.

Примечание: *dqpmigrate.bnd* находится в каталоге *sqllib/bnd*, а *dqpmigrate.exe* - в каталоге *sqllib/bin*.

Чтобы использовать **dqpmigrate** вручную для предоставления привилегий EXECUTE, после установки FixPak:

1. Свяжите файл пакета */sqllib/bnd/dqpmigrate.bnd* с базой данных, на которой был установлен сервер Query Patroller, введя команду:
`db2 bind dqpmigrate.bnd`
2. Выполните **dqpmigrate**, введя команду:
`dqpmigrate dbalias userid passwd`

Включение управления запросами

В главе "Getting Started" в разделе "Enabling Query Management" текст должен выглядеть так:

Для задания параметров конфигурации базы данных необходимо быть владельцем этой базы данных или обладать полномочиями SYSADM, SYSCTRL или SYSMAINT.

Запуск QueryAdministrator

В главе "Using QueryAdministrator to Administer DB2 Query Patroller" содержатся инструкции по запуску QueryAdministrator из меню Пуск Windows. В первом шаге содержится следующий текст:

If you are using Windows, you can select **DB2 Query Patroller** -> **QueryAdministrator** from the **IBM DB2** program group.

Этот текст должен выглядеть так:

DB2 Query Patroller -> **QueryAdmin**.

Управление пользователями

В разделе "User Administration" главы "Using QueryAdministrator to Administer DB2 Query Patroller" в определении параметра максимального затраченного времени сказано, что если он имеет значение 0 или -1, запрос всегда будет выполняться до завершения. Этому параметру нельзя задать отрицательное значение. В тексте должно быть сказано, что если для него задано значение 0, запрос всегда будет выполняться до завершения.

Параметр максимального числа запросов задает максимальное число заданий, одновременно выполняемых DB2 Query Patroller. Его значение должно быть целым числом от 0 до 32767.

Создание очереди заданий

В разделе "Job Queue Administration" главы "Using QueryAdministrator to Administer DB2 Query Patroller" вид экрана в шагах для "Создания очереди заданий" должен идти после второго шага. Окно Информация о новой очереди заданий открывается после нажатия на кнопку **Новое** на странице Управление очередью заданий инструмента QueryAdministrator.

Ссылки на страницу Очереди заданий и закладку Очереди заданий должны указывать на страницу Управление очередью заданий и закладку Управление очередью заданий соответственно.

Использование интерфейса командной строки

Чтобы передать запрос и чтобы была создана таблица результатов, пользователю с полномочиями User в системе DB2 Query Patroller могут быть нужны полномочия CREATETAB для базы данных. Такому пользователю не требуются полномочия CREATETAB для базы данных, если переменная профиля DQP_RES_TBLSPC не задана или если в качестве ее значения задано имя табличного пространства по умолчанию. В этих случаях таблица результатов будет успешно создана, поскольку пользователи имеют полномочия на создание таблиц в табличном пространстве по умолчанию.

Замечания по Query Enabler

- При использовании инструментов запросов других производителей, использующих указатель с набором ключей, запросы не будут перехватываться. Чтобы Query Enabler перехватывал такие запросы, нужно изменить файл `db2cli.ini`, задав в нем:

```
[common]
DisableKeySetCursor=1
```

- Для клиентов AIX убедитесь, что не задана переменная среды `LIBPATH`. Поставляемая с JDK библиотека `libXext.a` не совместима с библиотекой в каталоге `/usr/lib/X11`. Это вызывает проблемы с графическим интерфейсом Query Enabler.

DB2 Query Patroller Tracker может вернуть пустую страницу столбцов

FixPak 3 содержит исправление к DB2 Query Patroller Tracker. Теперь он корректно возвращает результаты запросов, для которых не найдено столбцов. Пример такого запроса - "SELECT COUNT(*) FROM ...". Поскольку такому запросу не соответствует ни один столбец таблицы, Tracker возвращает в качестве страницы столбцов пустую страницу. Это не является дефектом.

Приложение В. Troubleshooting DB2 Query Patroller Clients (Устранение неисправностей клиентов DB2 Query Patroller)

В Приложении В, Troubleshooting DB2 Query Patroller Clients (Устранение неисправностей клиентов DB2 Query Patroller) в разделе Common Query Enabler Problems (Общие ошибки Query Enabler) в описании второй ошибки текст первого пункта надо заменить на:

Убедитесь, что значение `path` содержит `jre`.

Часть 7. Разработка прикладных программ

Глава 24. Administrative API Reference	213	VisualAge C++ Версии 4.0 в OS/2 и Windows	242
db2ArchiveLog (новый API)	213	Глава 26. Application Development Guide	243
db2ArchiveLog	214	Написание хранимых процедур автоматизации OLE	243
db2ConvMonStream.	216	Глава 6. Common DB2 Application Techniques	243
db2DatabasePing (новый API)	216	Генерирование последовательных значений	243
db2DatabasePing - проверка связи с базой данных	217	Управление поведением последовательности	244
db2HistData	218	Улучшение производительности с помощью	
db2HistoryOpenScan	219	объектов последовательности	245
db2XaGetInfo (новый API)	220	Сравнение объектов последовательности и	
db2XaGetInfo - Получить информацию для		столбцов идентификации.	246
менеджера ресурсов	221	Глава 7. Stored Procedures	246
db2XaListIndTrans (новый API, заменяющий sqlxphqr)	221	Ошибка типа DECIMAL в Java-процедурах Linux	246
db2XaListIndTrans - Получить список		Использование указателей в рекурсивных хранимых	
неоднозначных транзакций	222	процедурах	246
db2GetSnapshot – получить снимок	225	Глава 12. Working with Complex Objects: User-Defined	
sqlaintrp – Получить сообщение об ошибке	225	Structured Types	247
sqlbctcq – Закрыть запрос к контейнеру табличного		Вставка в столбцы атрибутов	
пространства	225	структурированного типа	247
sqlubkr – Резервное копирование базы данных	225	Глава 20. Programming in C and C++	247
sqlureot – Реорганизовать таблицу.	226	Типы C/C++ для хранимых процедур,	
sqlurestore – Восстановить базу данных	226	пользовательских функций и методов	247
Ошибка документации о поддержке расширенной		Приложение В. Примеры программ	249
совместно используемой памяти AIX.	226	Активация встраиваемых модулей проектов и	
SQLFUPD	226	инструментов IBM DB2 Universal Database для	
locklist.	226	Microsoft Visual C++	249
SQLEDBDESC	227	Сигнатура метода Java в процедурах и функциях с	
Ошибка документации SQLFUPD	227	PARAMETER STYLE JAVA	249
Глава 25. Application Building Guide.	229	Провайдер OLE DB IBM DB2	249
Глава 1. Introduction	229	Особенности языка/Программирование на	
Поддерживаемое программное обеспечение	229	Java/Создание прикладных программ и апплетов	
Примеры программ	230	Java/Поддержка апплетов в Java	249
Глава 3. General Information for Building DB2		Глава 27. CLI Guide and Reference	251
Applications	231	Функции Unicode CLI и SQL_C_WCHAR	
Файлы построения, make-файлы и утилиты		поддерживаются только в AIX	251
проверки ошибок	231	Связывание утилит базы данных при использовании	
Глава 4. Building Java Applets and Applications	231	клиента времени выполнения	252
Задание среды	231	Дополнение к разделу "Using Compound SQL"	252
Уровень JDK в OS/2	233	Глава 3. Using Advanced Features	253
Глава 5. Building SQL Procedures	233	Поддержка указателей с возможностью	
Настройка среды процедур SQL	233	прокрутки на стороне сервера для OS/390	253
Задание переменных среды компилятора	234	Атрибуты указателя	253
Настройка команды компиляции	235	Поддерживаемые направления выборки	253
Удаление промежуточных файлов.	238	Изменение управляемого указателя	253
Резервное копирование и восстановление	239	Устранение неисправностей в программах,	
Создание процедур SQL	239	созданных до появления поддержки	
Вызов хранимых процедур	239	указателей с возможностью прокрутки	254
Распространение скомпилированных процедур		Глава 4. Configuring CLI/ODBC and Running Sample	
SQL	240	Applications	255
Глава 7. Building HP-UX Applications.	240	Ключевые слова конфигурации	255
HP-UX C	240	Написание хранимой процедуры в CLI	255
HP-UX C++	241	Хранимые процедуры и автоматическое связывание	
Глава 10. Building PTX Applications	241	CLI.	255
ptx/C++	241		
Глава 12. Building Solaris Applications	241		
SPARCCompiler C++.	241		

Дополнение к приложению D "Extended Scalar Functions": функции DAYOFWEEK_ISO() и WEEK_ISO().	255	SQL0228N (новое сообщение)	268
Приложение K. Using the DB2 CLI/ODBC/JDBC Trace Facility.	256	SQL0231W (новое сообщение)	268
Использование статического SQL в прикладных программах CLI	256	SQL0243N (новое сообщение)	269
Ограничения статического профилирования JDBC/ODBC/CLI	256	SQL0244N (новое сообщение)	269
Исправление описания параметра функции CLI SQLBindFileToParam().	257	SQL0270N (новый код причины 40)	269
SQLNextResult - Связать следующий набор результатов с другим хэндлом оператора	257	SQL0301N (новый текст объяснения)	269
Назначение	257	SQL0303N (новый текст)	270
Синтаксис	257	SQL0336N (новое сообщение)	270
Аргументы функции	257	SQL0348N (новое сообщение)	270
Использование	257	SQL0349N (новое сообщение)	271
Коды возврата	258	SQL0357N (новый код причины 3)	271
Диагностика	258	SQL0358N (новое действие пользователя 26)	271
Ограничения	259	SQL0368N (новое сообщение)	271
Ссылки	259	SQL0408N (новый текст)	272
Преобразования ADT	259	SQL0423N (исправленный текст)	272
Глава 28. Справочник по сообщениям	261	SQL0590N (новое сообщение)	272
CLI0645E (новое сообщение)	261	SQL0670N (исправленный текст)	273
CLI0646E (новое сообщение)	261	SQL0845N (новое сообщение)	274
CLI0647E (новое сообщение)	261	SQL0846N (новое сообщение)	274
DB21086I (новое сообщение)	261	SQL1179W (исправленный текст)	274
DB210060E (новое сообщение)	261	SQL1186N (новое сообщение)	274
DB210061E (новое сообщение)	262	SQL1550N (новый SQLCODE)	275
DB210062E (новое сообщение)	262	SQL1551N (новый SQLCODE)	275
DB210113E (новое сообщение)	262	SQL1552N (новый SQLCODE)	275
DB210114E (новое сообщение)	263	SQL1553N (новый SQLCODE)	276
DB210115E (новое сообщение)	263	SQL1704N (новые коды причины)	276
DB210116E (новое сообщение)	263	SQL2077W (новое сообщение)	276
DB210117E (новое сообщение)	263	SQL2078N (новое сообщение)	276
DB210118E (новое сообщение)	263	SQL2417N (новое сообщение)	277
DB210120E (новое сообщение)	263	SQL2426N (новое сообщение)	277
DB210200I (новое сообщение)	264	SQL2571N (новое сообщение)	277
DB210201I (новое сообщение)	264	SQL2572N (новое сообщение)	277
DBI1793W (новое сообщение)	264	SQL2573N (новое сообщение)	278
DBI1794E (новое сообщение)	264	SQL2574N (новое сообщение)	278
DBI1795E (новое сообщение)	264	SQL2575N (новое сообщение)	278
DBI1796W (новое сообщение)	264	SQL2576N (новое сообщение)	279
DBI1797I (новое сообщение)	265	SQL4942N (новый текст)	279
DWC13603E (новое сообщение)	265	SQL5012N (новое сообщение)	279
DWC13700E (новое сообщение)	265	SQL6583N (новое сообщение)	279
DWC13701E (новое сообщение)	265	SQL20005N (новое сообщение)	280
DWC13702E (новое сообщение)	265	SQL20117N (изменен код причины 1)	280
DWC13703E (новое сообщение)	266	SQL20121N (новое сообщение)	280
DWC13705E (новое сообщение)	266	SQL20133N (новое сообщение)	280
DWC13706E (новое сообщение)	266	SQL20134N (новое сообщение)	280
DWC13707E (новое сообщение)	266	SQL20135N (новое сообщение)	280
SQL0017N (новое сообщение)	267	SQL20143N (новое сообщение)	281
SQL0056N (новое сообщение)	267	SQL20144N (новое сообщение)	281
SQL0057N (новое сообщение)	267	SQL20145N (новое сообщение)	281
SQL0058N (новое сообщение)	267	SQL20146N (новое сообщение)	281
SQL0097N (новое сообщение)	267	SQL20147N (новое сообщение)	281
SQL0224N (новое сообщение)	267	SQL20148N (новое сообщение)	282
SQL0225N (новое сообщение)	268	SQL20153N (новое сообщение)	282
SQL0227N (новое сообщение)	268	Новые значения SQLSTATE	282
		Глава 29. SQL Reference	285
		Функции OLAP	285
		Процедуры и составные операторы SQL	287
		LCASE и UCASE (Unicode)	289
		WEEK_ISO	289
		Правила именования и неявные спецификаторы имен объектов	289

Queries (Запросы)	290	Синтаксическая диаграмма	329
оператор select/условие fetch-first	290	Справочник по последовательностям	330
оператор select/синтаксическая диаграмма	290	Глава 4. Functions (Функции)	333
Библиотеки, используемые оператором CREATE WRAPPER в Linux	291	Процедуры	333
Теперь поддерживается возможность изменения ключа разделения	291	GET_ROUTINE_SAR	334
Оператор: ALTER TABLE	291	PUT_ROUTINE_SAR	334
Оператор: CREATE TABLE	291	Скалярные функции	336
Оператор: DECLARE GLOBAL TEMPORARY TABLE PARTITIONING KEY (имя-столбца,...)	291	IDENTITY_VAL_LOCAL	336
Оператор: SET переходная-переменная	291	Глава 6. SQL Statements (Операторы SQL)	340
Оператор: UPDATE	291	Составной оператор (динамический)	340
Длинные ключи индекса для баз данных Unicode	291	Составной оператор (динамический)	341
CREATE TABLE	291	Глава 7 "SQL Procedures" теперь называется "Управляющие операторы SQL"	343
CREATE INDEX	292	Оператор процедуры SQL	343
ALTER TABLE	292	Оператор процедуры SQL	344
Включение новых встроенных скалярных функций SQL	292	Заголовок раздела Compound Statement изменен на Составной оператор процедуры	345
ABS или ABSVAL	292	Оператор FOR	345
MULTIPLY_ALT	293	Оператор FOR	346
ROUND	294	Оператор RETURN	347
Примеры:	295	оператор RETURN	348
Language Elements/DATALINK Assignments (Элементы языка/Присваивания DATALINK)	295	Оператор SIGNAL	349
Функция ENCRYPT (новая скалярная функция)	296	Оператор SIGNAL	350
ENCRYPT	296	Приложение D: Catalog Views (Производные таблицы каталога)	352
Функция DECRYPT_BIN и DECRYPT_CHAR (новая скалярная функция)	298	SYSCAT.SEQUENCES	352
DECRYPT_BIN и DECRYPT_CHAR	298	Глава 30. Построитель хранимых процедур DB2	355
Функция GETHINT (новая скалярная функция)	299	Поддержка Java 1.2 для построителя хранимых процедур DB2	355
GETHINT	299	Удаленная отладка хранимых процедур DB2	356
CREATE SEQUENCE (новый оператор SQL)	300	Построение процедур SQL на платформах Windows, OS/2 или UNIX	356
CREATE SEQUENCE	301	Использование построителя хранимых процедур DB2 на платформе Solaris	356
ALTER SEQUENCE (новый оператор SQL)	305	Известные проблемы и ограничения	356
ALTER SEQUENCE	306	Использование построителя хранимых процедур DB2 с традиционным китайским языком	357
GRANT (привилегии для последовательности) (новый оператор SQL)	309	Установки в UNIX (AIX, Sun Solaris, Linux) и построитель хранимых процедур	357
GRANT (привилегии для последовательности)	310	Построение хранимых процедур SQL в OS/390	357
SET ENCRYPTION PASSWORD (новый оператор SQL)	310	Отладка хранимых процедур SQL	357
SET ENCRYPTION PASSWORD	311	Экспорт хранимых процедур Java	357
Изменения для оператора ALTER TABLE	311	Вставка хранимых процедур в OS/390	358
Изменения раздела Compound SQL (Embedded) (Составные операторы SQL (встроенные))	315	Задание опций построения для хранимых процедур SQL на сервере рабочих станций	358
Изменения описания оператора CREATE FUNCTION (с источником или шаблоном)	315	Автоматическое обновление адресного пространства WLM для построения хранимых процедур в OS/390	359
Изменения описания оператора CREATE FUNCTION (скалярная SQL, табличная или строковая)	316	Глава 31. Обновления для Unicode	361
Изменения оператора CREATE METHOD	318	Введение	361
Изменения оператора CREATE TRIGGER	320	Базы данных Unicode и прикладные программы DB2	361
Изменения для оператора DROP	322	Обновление документации	363
Изменения для оператора DELETE	323	SQL Reference	364
Изменения для оператора INSERT	324	Глава 3. Language Elements	364
Изменения для оператора SELECT INTO	325	Расширение типов данных	364
Изменения для оператора SET переходная-переменная	325	Преобразование типов данных	364
SET переменная	326	Назначения и сравнения	365
Изменения для оператора UPDATE	329	Правила для типов данных результатов	366
Глава 3. Language Elements (Элементы языка)	329		
Expressions (Выражения)	329		

Rules for String Conversions	367
Expressions	367
Predicates	367
Глава 4. Functions	368
Scalar Functions	368
CLI Guide and Reference	368
Глава 3. Using Advanced Features	368
Написание прикладной программы Unicode	
CLI DB2	368
Приложение C. DB2 CLI and ODBC	373
Прикладные программы Unicode ODBC	373
Data Movement Utilities Guide and Reference	374
Приложение C. Export/Import/Load Utility File	
Formats	374

Глава 24. Administrative API Reference

| **db2ArchiveLog (новый API)**

|

db2ArchiveLog - архивировать активный журнал

db2ArchiveLog

Закрывает и усекает активный файл журнала восстановимой базы данных. Если включен обработчик пользователя, шлет требование архивирования.

Авторизация

Одни из следующих:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

Необходимое соединение

Этот API автоматически устанавливает соединение с указанной базой данных. Если соединение с указанной базой данных уже существует, API возвращает ошибку.

Файлы включения API

db2ApiDf.h

Синтаксис API C

```
/* Файл: db2ApiDf.h */
/* API: архивировать активный журнал */
SQL_API_RC SQL_API_FN
db2ArchiveLog (
    db2UInt32 version,
    void *pDB2ArchiveLogStruct,
    struct sqlca * pSqlca);

typedef struct
{
    char          *piDatabaseAlias;
    char          *piUserName;
    char          *piPassword;
    db2UInt16     iAllNodeFlag;
    db2UInt16     iNumNodes;
    SQL_PDB_NODE_TYPE *piNodeList;
    db2UInt32     iOptions;
}
```

Общий синтаксис API

```

/* Файл: db2ApiDf.h */
/* API: архивировать активный журнал */
SQL_API_RC SQL_API_FN
db2gArchiveLog (
    db2UInt32 version,
    void *pDB2ArchiveLogStruct,
    struct sqlca * pSqlca);

typedef struct
{
    db2UInt32          iAliasLen;
    db2UInt32          iUserNameLen;
    db2UInt32          iPasswordLen;
    char              *piDatabaseAlias;
    char              *piUserName;
    char              *piPassword;
    db2UInt16          iAllNodeFlag;
    db2UInt16          iNumNodes;
    SQL_PDB_NODE_TYPE *piNodeList;
    db2UInt32          iOptions;
}

```

Параметры API**version**

Входной. Задаёт уровень версии и выпуска для переменной, передаваемой во втором параметре *-iOptions*.

pDB2ArchiveLogStruct

Входной. Указатель на структуру *db2ArchiveLogStruct*.

pSqlca Выходной. Указатель на структуру *sqlca*.

iAliasLen

Входной. 4-байтное целое без знака, задающее длину в байтах алиаса базы данных.

iUserNameLen

4-байтное целое без знака, задающее длину в байтах имени пользователя. Если имя пользователя не используется, имеет значение 0.

iPasswordLen

Входной. 4-байтное целое без знака, задающее длину в байтах пароля. Если имя пользователя не используется, имеет значение 0.

piDatabaseAlias

Входной. Строка, содержащая алиас (как он внесен в системный каталог баз данных) базы данных, для которой надо архивировать активный журнал.

piUserName

Входной. Строка, содержащая имя пользователя для попытки соединения.

piPassword

Входной. Строка, содержащая пароль имени пользователя для попытки соединения.

iAllNodeFlag

Только для MPP. Входной. Флаг, указывающий, на всех ли узлах *db2nodes.cfg* должна выполняться команда архивирования журнала. Допустимые значения:

DB2ARCHIVELOG_NODE_LIST

Применяется на узлах из переданного в параметре *piNodeList* списка.

db2ArchiveLog - архивировать активный журнал

DB2ARCHIVELOG_ALL_NODES

Применяется на всех узлах. piNodeList должен иметь значение NULL.
Это значение по умолчанию.

DB2ARCHIVELOG_ALL_EXCEPT

Применяется на всех узлах, кроме узлов из переданного в параметре piNodeList списка.

iNumNodes

Только для MPP. Входной. Задаёт число узлов в списке piNodeList.

piNodeList

Только для MPP. Входной. Указатель на массив номеров узлов, на которых выполняется команда архивирования журнала.

iOptions

Входной. Резервирован для будущего использования.

db2ConvMonStream

В разделе Usage Notes структура для типа потока данных переменной снимка SQLM_ELM_SUBSECTION должна быть sqlm_subsection.

db2DatabasePing (новый API)

db2DatabasePing - проверка связи с базой данных

Проверяет время ответа сети для соединения между клиентом и сервером баз данных. Этот API может использоваться любой программой, когда обращение к серверу баз данных хоста осуществляется через DB2 Connect, как напрямую, так и через шлюз.

Авторизация

Нет

Необходимое соединение

База данных

Файлы включения API

db2ApiDf.h

Синтаксис API C

```
/* Файл: db2ApiDf.h */
/* API: Проверка соединения с базой данных */
/* ... */
```

```
SQL_API_RC SQL_API_FN
db2DatabasePing (
    db2UInt32 versionNumber,
    void      *pParmStruct,
    struct sqlca *pSqlca);
/* ... */
```

```
typedef SQL_STRUCTURE db2DatabasePingStruct
{
    char          iDbAlias[SQL_ALIAS_SZ + 1];
    db2UInt16     iNumIterations;
    db2UInt32     *poElapsedTime;
}
```

Общий синтаксис API

```
/* Файл: db2ApiDf.h */
/* API: Проверка соединения с базой данных */
/* ... */
```

```
SQL_API_RC SQL_API_FN
db2gDatabasePing (
    db2UInt32 versionNumber,
    void      *pParmStruct,
    struct sqlca *pSqlca);
/* ... */
```

```
typedef SQL_STRUCTURE db2gDatabasePingStruct
{
    db2UInt16     iDbAliasLength;
    char          iDbAlias[SQL_ALIAS_SZ];
    db2UInt16     iNumIterations;
    db2UInt32     *poElapsedTime;
}
```

Параметры API

versionNumber

Входной. Версия и выпуск продукта DB2 Universal Database или DB2 Connect, которые использует программа.

Примечание: Для DB2 Версии 7.1 или новее должна использоваться константа db2Version710.

db2ArchiveLog - архивировать активный журнал

iDbAliasLength

Входной. Длина алиаса базы данных.

Примечание: Этот параметр сейчас не используется. Он зарезервирован для будущего использования.

iDbAlias

Входной. Алиас базы данных.

Примечание: Этот параметр сейчас не используется. Он зарезервирован для будущего использования.

iNumIterations

Входной. Число повторений требований тестирования. Значение должно находиться в диапазоне от 1 до 32767 включительно.

poElapsedTime

Выходной. Указатель на массив 32-битных целых, число элементов равно значению iNumIterations. Каждый элемент в этом массиве будет содержать время в микросекундах, затраченное на одно выполнение требования тестирования.

Примечание: За выделение памяти для этого массива перед вызовом данного API отвечает прикладная программа.

pSqlca Выходной. Указатель на структуру *sqlca*. Дополнительную информацию об этой структуре смотрите в руководстве *Administrative API Reference*.

Замечания по использованию

Перед вызовом данного API соединение с базой данных должно существовать, иначе произойдет ошибка.

Эту функцию можно также вызвать при помощи команды PING. Описание этой команды смотрите в руководстве *Command Reference*.

db2HistData

К Таблице 11 надо добавить следующие строки. Поля в структуре db2HistData:

Имя поля	Тип данных	Описание
oOperation	char	Смотрите таблицу 12.
oOptype	char	Смотрите таблицу 13.

После таблицы 11 надо добавить следующую таблицу.

Таблица 12. Допустимые значения событий для oOperation в структуре db2HistData

Значение	Описание	Определение на C	Определение на COBOL/FORTRAN
A	добавить табличное пространство	DB2HISTORY_OP_ADD_TABLESPACE	DB2HIST_OP_ADD_TABLESPACE
B	резервное копирование	DB2HISTORY_OP_BACKUP	DB2HIST_OP_BACKUP
C	загрузка и копирование	DB2HISTORY_OP_LOAD_COPY	DB2HIST_OP_LOAD_COPY
D	отброшенная таблица	DB2HISTORY_OP_DROPPED_TABLE	DB2HIST_OP_DROPPED_TABLE
F	повтор транзакций	DB2HISTORY_OP_ROLLFWD	DB2HIST_OP_ROLLFWD
G	перестроить таблицу	DB2HISTORY_OP_REORG	DB2HIST_OP_REORG
L	загрузка	DB2HISTORY_OP_LOAD	DB2HIST_OP_LOAD

db2ArchiveLog - архивировать активный журнал

Значение	Описание	Определение на C	Определение на COBOL/FORTRAN
N	переименовать табличное пространство	DB2HISTORY_OP_REN_TABLESPACE	DB2HIST_OP_REN_TABLESPACE
O	отбросить табличное пространство	DB2HISTORY_OP_DROP_TABLESPACE	DB2HIST_OP_DROP_TABLESPACE
Q	стабилизация	DB2HISTORY_OP_QUIESCE	DB2HIST_OP_QUIESCE
R	восстановление	DB2HISTORY_OP_RESTORE	DB2HIST_OP_RESTORE
S	запустить статистику	DB2HISTORY_OP_RUNSTATS	DB2HIST_OP_RUNSTATS
T	изменить табличное пространство	DB2HISTORY_OP_ALT_TABLESPACE	DB2HIST_OP_ALT_TBS
U	выгрузить	DB2HISTORY_OP_UNLOAD	DB2HIST_OP_UNLOAD

Надо также добавить следующую таблицу.

Таблица 13. Допустимые значения oOptype в структуре db2HistData

oOperation	oOptype	Описание	Описание C/COBOL/FORTRAN
B	F	автономный	DB2HISTORY_OPTYPE_OFFLINE
	N	оперативный	DB2HISTORY_OPTYPE_ONLINE
	I	инкрементный автономный	DB2HISTORY_OPTYPE_INCR_OFFLINE
	O	инкрементный оперативный	DB2HISTORY_OPTYPE_INCR_ONLINE
	D	разностный автономный	DB2HISTORY_OPTYPE_DELTA_OFFLINE
	E	разностный оперативный	DB2HISTORY_OPTYPE_DELTA_ONLIN
F	E	до конца журнала	DB2HISTORY_OPTYPE_EOL
	P	до момента времени	DB2HISTORY_OPTYPE_PIT
L	I	вставка	DB2HISTORY_OPTYPE_INSERT
	R	замена	DB2HISTORY_OPTYPE_REPLACE
Q	S	стабилизация в совместном режиме	DB2HISTORY_OPTYPE_SHARE
	U	изменение стабилизации	DB2HISTORY_OPTYPE_UPDATE
	X	стабилизация в монопольном режиме	DB2HISTORY_OPTYPE_EXCL
	Z	сброс стабилизации	DB2HISTORY_OPTYPE_RESET
R	F	автономный	DB2HISTORY_OPTYPE_OFFLINE
	N	оперативный	DB2HISTORY_OPTYPE_ONLINE
	I	инкрементный автономный	DB2HISTORY_OPTYPE_INCR_OFFLINE
	O	инкрементный оперативный	DB2HISTORY_OPTYPE_INCR_ONLINE
T	C	добавить контейнеры	DB2HISTORY_OPTYPE_ADD_CONT
	R	перебалансировка	DB2HISTORY_OPTYPE_REB

db2HistoryOpenScan

К параметру **iCallerAction** надо добавить следующее значение.

DB2HISTORY_LIST_CRT_TABLESPACE

Выбирает только записи CREATE TABLESPACE и DROP TABLESPACE, прошедшие прочие фильтры.

db2ArchiveLog - архивировать активный журнал

db2XaGetInfo (новый API)

db2XaGetInfo - Получить информацию для менеджера ресурсов

Получает информацию для конкретного менеджера ресурсов после того, как были сделаны вызовы `xa_open`.

Авторизация

Нет

Необходимое соединение

База данных

Файлы включения API

`sqlxa.h`

Синтаксис API C

```

/* Файл: sqlxa.h */
/* Функция API: Получить информацию для менеджера ресурсов */
/* ... */
SQL_API_RC SQL_API_FN
db2XaGetInfo (
    db2UInt32 versionNumber,
    void * pParmStruct,
    struct sqlca * pSqlca);

typedef SQL_STRUCTURE db2XaGetInfoStruct
{
    db2int32 iRmid;
    struct sqlca oLastSqlca;
} db2XaGetInfoStruct;

```

Параметры API

versionNumber

Входной. Задаёт уровень версии и выпуска для структуры, передаваемой во втором параметре `pParmStruct`.

pParmStruct

Входной. Указатель на структуру `db2XaGetInfoStruct`.

pSqlca Выходной. Указатель на структуру `sqlca`. Дополнительную информацию об этой структуре смотрите в руководстве *Administrative API Reference*.

iRmid Входной. Задаёт менеджер ресурсов, для которого требуется информация.

oLastSqlca

Выходной. Содержит `sqlca` для последнего вызова API XA.

Примечание: Можно получить только `sqlca` для последнего неудачного вызова API XA.

db2XaListIndTrans (новый API, заменяющий sqlxphqr)

db2ArchiveLog - архивировать активный журнал

db2XaListIndTrans - Получить список неоднозначных транзакций

Выдает список всех неоднозначных транзакций для базы данных, с которой в настоящее время установлено соединение.

Область действия

Эта функция API влияет только на узел, на котором она вызвана.

Авторизация

Одни из следующих:

- *sysadm*
- *dbadm*

Необходимое соединение

База данных

Файлы включения API

db2ApiDf.h

Синтаксис API C

```
/* Файл: db2ApiDf.h */
/* Функция API: Получить список неоднозначных транзакций */
/* ... */
SQL_API_RC SQL_API_FN
db2XaListIndTrans (
    db2UInt32 versionNumber,
    void * pParmStruct,
    struct sqlca * pSqlca);

typedef SQL_STRUCTURE db2XaListIndTransStruct
{
    db2XaRecoverStruct * piIndoubtData;
    db2UInt32            iIndoubtDataLen;
    db2UInt32            oNumIndoubtsReturned;
    db2UInt32            oNumIndoubtsTotal;
    db2UInt32            oReqBufferLen;
} db2XaListIndTransStruct;

typedef SQL_STRUCTURE db2XaRecoverStruct
{
    sqluint32            timestamp;
    SQLXA_XID            xid;
    char                  dbalias[SQLXA_DBNAME_SZ];
    char                  applid[SQLXA_APPLID_SZ];
    char                  sequence_no[SQLXA_SEQ_SZ];
    char                  auth_id[SQL_USERID_SZ];
    char                  log_full;
    char                  connected;
    char                  indoubt_status;
    char                  originator;
    char                  reserved[8];
} db2XaRecoverStruct;
```

Параметры API

versionNumber

Входной. Задаёт уровень версии и выпуска для структуры, передаваемой во втором параметре *pParmStruct*.

pParmStruct

Входной. Указатель на структуру *db2XaListIndTransStruct*.

db2ArchiveLog - архивировать активный журнал

pSqlca Выходной. Указатель на структуру *sqlca*. Дополнительную информацию об этой структуре смотрите в руководстве *Administrative API Reference*.

piIndoubtData

Входной. Указатель на поставляемый прикладной программой буфер, в который будут возвращена информация о неоднозначных транзакциях. Эта возвращаемая информация имеет формат *db2XaRecoverStruct*. Прикладная программа может просмотреть список неоднозначных транзакций, начиная с адреса, указанного в этом параметре, и используя размер структуры *db2XaRecoverStruct*.

Если этот параметр имеет пустое значение, DB2 вычислит необходимый размер буфера и вернет его значение в *oReqBufferLen*. *oNumIndoubtsTotal* будет содержать общее число неоднозначных транзакций. Прикладная программа может выделить буфер нужного размера и вновь вызвать эту функцию API.

oNumIndoubtsReturned

Выходной. Число записей о неоднозначных транзакциях, возвращенных в буфере, указанном в *pIndoubtData*.

oNumIndoubtsTotal

Выходной. Общее число записей о неоднозначных транзакциях, доступных в момент вызова этой функции API. Если буфер *piIndoubtData* недостаточно большой, чтобы вместить все эти записи, значение *oNumIndoubtsTotal* будет больше, чем *oNumIndoubtsReturned*. Прикладная программа может повторно вызвать эту функцию API, чтобы получить все записи.

Примечание: Между вызовами API это число может измениться - в результате автоматической или эвристической ресинхронизации неоднозначных транзакций или из-за того, что другие транзакции окажутся в неоднозначном состоянии.

oReqBufferLen

Выходной. Длина буфера, необходимая для вмещения всех записей о неоднозначных транзакциях, доступных в момент вызова этой функции API. Прикладная программа может использовать это значение, чтобы определить требуемый размер буфера, вызвав эту функцию API с пустым значением параметра *pIndoubtData*. Затем она может использовать это значение для выделения требуемого буфера и вновь вызвать эту функцию API, задав в *pIndoubtData* адрес выделенного буфера.

Примечание: Требуемый размер буфера может измениться между вызовами API - в результате автоматической или эвристической ресинхронизации неоднозначных транзакций или из-за того, что другие транзакции окажутся в неоднозначном состоянии. Учитывая это, прикладная программа может выделять буфер большего размера.

отметка времени (timestamp)

Выходной. Указывает время, когда транзакция перешла в неоднозначное состояние.

xid Выходной. Указывает идентификатор XA, присвоенный менеджером транзакций для однозначной идентификации глобальной транзакции.

dbalias

Выходной. Указывает алиас базы данных, в которой обнаружена эта неоднозначная транзакция.

db2ArchiveLog - архивировать активный журнал

applid Выходной. Указывает идентификатор прикладной программы, присвоенный менеджером баз данных этой транзакции.

sequence_no Выходной. Указывает порядковый номер, присвоенный менеджером баз данных в качестве расширения для *applid*.

auth_id Выходной. Указывает ID авторизации пользователя, выполнявшего эту транзакцию.

log_full Выходной. Указывает, вызвала ли эта транзакция условие переполнения журнала. Допустимые значения:

SQLXA_TRUE
Эта неоднозначная транзакция вызвала условие переполнения журнала.

SQLXA_FALSE
Эта неоднозначная транзакция не вызвала условия переполнения журнала.

connected Выходной. Указывает, соединена ли эта прикладная программа. Допустимые значения:

SQLXA_TRUE
Для этой транзакции выполняется нормальная обработка точек синхронизации и она ожидает второй фазы двухфазного принятия.

SQLXA_FALSE
Эта транзакция осталась неоднозначной из-за более ранней ошибки и теперь она ожидает ресинхронизации менеджером транзакций.

indoubt_status Выходной. Указывает состояние этой неоднозначной транзакции. Допустимые значения:

SQLXA_TS_PREP
Транзакция подготовлена. Можно использовать параметр *connected*, чтобы определить, ожидает транзакция второй фазы нормальной обработки принятия или возникла ошибка и требуется ресинхронизация менеджером транзакций.

SQLXA_TS_HCOM
Для транзакции выполнено эвристическое принятие.

SQLXA_TS_HROL
Для транзакции выполнен эвристический откат.

SQLXA_TS_MACK
Транзакция не получила подтверждение принятия от узла в многораздельной базе данных.

SQLXA_TS_END
Транзакция закончена для этой базы данных. Эта транзакция впоследствии может быть вновь активирована, или для нее могут быть выполнены принятие или откат. Возможно также, что менеджер транзакций обнаружит ошибку и транзакция не будет выполнена. В таком случае для этой транзакции требуются эвристические действия, поскольку она может удерживать блокировки и мешать другим прикладным программам обращаться к данным.

Замечания по использованию

Типичная прикладная программа выполняет следующие шаги после установления соединения с базой данных или с узлом координатора многораздельной базы данных:

1. Вызывает **db2XaListIndTrans** с пустым (NULL) значением параметра *piIndoubtData*. Будут возвращены значения *oReqBufferLen* и *oNumIndoubtsTotal*.
2. Использует возвращенное значение *oReqBufferLen*, чтобы выделить буфер. Этот буфер может оказаться недостаточно большим, если после вызова этой функции API для получения значения *oReqBufferLen* появились дополнительные неоднозначные транзакции. Прикладная программа может выделить буфер больше, чем задано значением *oReqBufferLen*.
3. Определяет, все ли записи о неоднозначных транзакциях были получены. Для этого она сравнивает значения *oNumIndoubtsReturned* и *oNumIndoubtTotal*. Если *oNumIndoubtsTotal* больше, чем *oNumIndoubtsReturned*, прикладная программа может повторить описанные выше шаги.

Смотрите также

"sqlxhfrg - Забыть состояние транзакции", "sqlxphcm - Выполнить принятие неоднозначной транзакции" и "sqlxphrl - Выполнить откат неоднозначной транзакции" в руководстве *Administrative API Reference*.

db2GetSnapshot – получить снимок

Синтаксис API db2GetSnapshot должен выглядеть так:

```
int db2GetSnapshot( unsigned char version;  
db2GetSnapshotData *data,  
struct sqlca *sqlca);
```

Параметры, описанные в данных:

```
typedef struct db2GetSnapshotData{  
    sqlma *piSqlmaData;  
    sqlm_collected *poCollectedData  
    void *poBuffer;  
    db2uint32 iVersion;  
    db2int32 iBufferSize;  
    db2uint8 iStoreResult;  
    db2uint16 iNodeNumber;  
    db2uint32 *poOutputFormat;  
  
}db2GetSnapshotData;
```

sqlaintp – Получить сообщение об ошибке

В описание этого API надо добавить следующее замечание по использованию:

В многопоточных прикладных программах функция sqlaintp должна быть подключена к правильному контексту. В противном случае не может быть получен текст сообщения для SQLCODE -1445.

sqlbctscq – Закрывает запрос к контейнеру табличного пространства

У загрузки нет достаточного уровня полномочий для данного API.

sqlubkr – Резервное копирование базы данных

Для параметра **BackupType** значение SQLUB_FULL изменено на SQLUB_DB. При этом снимается резервная копия всех табличных пространств в базе данных.

db2ArchiveLog - архивировать активный журнал

Для поддержки новой возможности инкрементного копирования добавлены также параметры SQLUB_INCREMENTAL и SQLUB_DELTA. Инкрементная резервная копия - это копия всех данных базы данных, измененных с момента последнего успешного снятия полной копии. Разностная резервная копия - это копия всех данных базы данных, измененных с момента последнего успешного снятия копии любого типа.

sqlureot – Реорганизовать таблицу

В Замечания по использованию надо добавить следующее предложение:

REORGANIZE TABLE не может использовать индекс на основе расширения индекса.

sqlurestore – Восстановить базу данных

Для параметра **RestoreType** значение SQLUD_FULL изменено на SQLUD_DB. При этом восстанавливаются все табличные пространства в базе данных. Восстановление производится в автономном режиме.

Для поддержки новой возможности инкрементного восстановления добавлен также параметр SQLUD_INCREMENTAL.

Инкрементная резервная копия - это копия всех данных базы данных, измененных с момента последнего успешного снятия полной копии.

Ошибка документации о поддержке расширенной совместно используемой памяти AIX

В Приложении E "Threaded Applications with Concurrent Access" примечание 2 следует читать так:

2. По умолчанию AIX не разрешает 32-битным программам присоединять более 11 сегментов совместно используемой памяти на один процесс, максимум 10 из которых можно использовать для соединений DB2.

Хотя расширенную совместно используемую память можно использовать для увеличения максимального числа сегментов совместно используемой памяти для процесса и для программ клиентов, DB2 не поддерживает расширенную совместно используемую память, поэтому она не может превысить максимальное число - 10 соединений DB2 на процесс.

SQLFUPD

locklist

Имя маркера изменено с SQLF_DBTN_LOCKLIST на SQLF_DBTN_LOCK_LIST. Тип параметра locklist изменен с SMALLINT на 64-битное INTEGER без знака. В таблицу Изменяемые параметры конфигурации базы данных надо внести следующее дополнение.

Имя параметра	Маркер	Значение маркера	Тип данных
locklist	SQLF_DBTN_LOCK_LIST	704	Uint64

Новое максимальное значение для этого параметра - 524 288.

SQLLEDBDESC

В список допустимых значений для SQLDBCSS (определенный в sqlenv) добавлены два новых значения:

SQL_CS_SYSTEM-NLSCHAR

Последовательность слияния для системы использует национальную версию подпрограмм сравнения для символьных типов.

SQL_CS_USER-NLSCHAR

Последовательность слияния для пользователя использует национальную версию подпрограмм сравнения для символьных типов.

Ошибка документации SQLFUPD

В Главе 3 “Data Structures” в Таблице 53 Updatable Database Configuration для параметра dbhear указано неверное значение маркера - 701. Верное значение - 58.

db2ArchiveLog - архивировать активный журнал

Глава 25. Application Building Guide

Глава 1. Introduction

Поддерживаемое программное обеспечение

AIX

Список версий компилятора C и C++ должен выглядеть так:

IBM C and C++ Compilers for AIX Версия 3.6.6 (Версия 3.6.6.3 для 64-битных систем)
IBM C for AIX 4.4
IBM VisualAge C++ Версия 4.0

Примечание: Загрузите для этих версий компиляторов последние пакеты исправлений, доступные по адресу

<http://www.ibm.com/software/ad/vacpp/service/csd.html>

Список версий компилятора Micro Focus COBOL должны выглядеть так:

AIX 4.2.1

Micro Focus COBOL Версия 4.0.20 (PRN 12.03 или новее)
Micro Focus COBOL Версии 4.1.10 (PRN 13.04 или новее)

AIX 4.3

Micro Focus COBOL Server Express Версия 1.0

Примечание: Информацию о поддержке DB2 хранимых процедур Micro Focus COBOL и пользовательских функций в AIX 4.3 смотрите на Web-странице DB2 Application Development:

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

При построении 64-битных программ с помощью компилятора IBM XL Fortran for AIX Версии 5.1.0 на стадиях компилирования и компоновки используйте опцию "-q64". Имейте в виду, что 64-битные программы не поддерживаются прежними версиями этого компилятора.

HP-UX

Список версий компилятора C++ должен выглядеть так:

HP aC++, Версия A.03.25

Примечание: HP не поддерживает двоичной совместимости между объектами, скомпилированными старыми и новыми компиляторами, поэтому это вызовет перекомпиляцию всех прикладных программ C++, используемых для обращения к DB2 в HP-UX. Обработка исключительных ситуаций в прикладных программах C++ также должна быть скомпилирована новым компилятором.

Руководство по переходу к aCC смотрите по адресу:

<http://www.hp.com/esy/lang/cpp/tguide>. Информацию о несовместимости C++ можно найти по адресу:

<http://www.hp.com/esy/lang/cpp/tguide/transcontent.html#RN.CVT.1.2>

<http://www.hp.com/esy/lang/cpp/tguide/transcontent.html#RN.CVT.3.3>

Сравнение C и C++ можно найти по адресу:

<http://www.hp.com/esy/lang/cpp/tguide/transcontent.html#RN.CVT.3.3.1>

Хотя C и aCC совместимы, если используются два различных типа объектов, для компиляции объекта, содержащего "main", и для компоновки конечного выполняемого файла должен использоваться aCC.

Linux

DB2 for Linux поддерживает следующую версию REXX:

Object REXX Interpreter for Linux Версия 2.1

Linux/390

DB2 for Linux/390 поддерживает только языки Java, C и C++.

OS/2

Список версий компилятора C/C++ должен выглядеть так:

IBM VisualAge C++ for OS/2 Версия 3.6.5 и Версия 4.0

Примечание: Загрузите для этих версий компиляторов последние пакеты исправлений, доступные по адресу

<http://www.ibm.com/software/ad/vacpp/service/csd.html>

Solaris

Список версий компилятора Micro Focus COBOL должен выглядеть так:

Micro Focus COBOL Server Express Версия 1.0

32-битные операционные системы Windows

Список версий компилятора IBM VisualAge C++ должен выглядеть так:

IBM VisualAge C++ для Windows Версий 3.6.5 и 4.0

Примечание: Загрузите для этих версий компиляторов последние пакеты исправлений, доступные по адресу

<http://www.ibm.com/software/ad/vacpp/service/csd.html>

Список версий компилятора Micro Focus COBOL должны выглядеть так:

Micro Focus COBOL Версия 4.0.20
Micro Focus COBOL Net Express Версия 3.0

Примеры программ

В раздел "Object Linking and Embedding Samples" надо добавить:

salarycltc

Написанный на Visual C++ пример CLI DB2, вызывающий хранимую процедуру salarysrv, написанную на Visual Basic.

SALSVADO

Пример хранимой процедуры автоматизации OLE (SALSVADO) и клиента

SALCLADO (SALCLADO), реализованного на 32-битном Visual Basic и ADO, который вычисляет средний оклад в таблице staff2.

В раздел "Log Management User Exit Samples" (Примеры обработчиков пользователей для управления журналами) должно быть добавлено следующее:

В AIX для построения прикладных программ, использующих клиент API ADSM уровня 3.1.6 и выше, должны использоваться вызовы компилятора xlc_r или xlc_r, а не xlc или xlc, даже если эти прикладные программы однопоточные. Это гарантирует использование библиотек, обеспечивающих правильную работу потоков. Это относится к примеру обработчика пользователя для управления журналами db2uext2.cadsm.

Если прикладная программа скомпилирована с библиотекой, не обеспечивающей работу потоков, можно применить исправление IC21925E или обратиться к поставщику этой прикладной программы. Это исправление можно получить с анонимного ftp-сервера index.storsys.ibm.com. Оно понижает уровень API ADSM до 3.1.3.

Глава 3. General Information for Building DB2 Applications

Файлы построения, make-файлы и утилиты проверки ошибок

Запись для blddevm в таблице 16 должна выглядеть так:

blddevm

Пример программы монитора событий, evm (доступен только в операционных системах AIX, OS/2 и 32-битных операционных системах Windows).

Таблица 17 должна содержать следующие записи:

bldmevm

Пример программы монитора событий, evm, с компилятором Microsoft Visual C++.

bldvevm

Пример программы монитора событий, evm, с компилятором VisualAge C++.

Глава 4. Building Java Applets and Applications

Задание среды

Если для построения программ SQLJ на поддерживаемых платформах используется IBM JDK 1.1.8, он должен иметь дату построения 24 ноября 1999 (или более позднюю). В противном случае во время компиляции могут возникать ошибки JNI rapic.

Если для построения программ SQLJ на поддерживаемых платформах используется IBM JDK 1.2.2, он должен иметь дату построения 17 апреля 2000 (или более позднюю). В противном случае во время компиляции могут возникать ошибки Invalid Java type.

В подразделах **AIX**, **HP-UX**, **Linux** и **Solaris** замените информацию о JDBC 2.0 на следующий текст:

Использование драйвера JDBC 2.0 с прикладными программами Java

Драйвер JDBC 1.22 остается драйвером по умолчанию для всех операционных систем. Чтобы использовать преимущества новых возможностей JDBC 2.0, необходимо установить поддержку JDK 1.2. Перед выполнением прикладной программы, использующей преимущества новых возможностей JDBC 2.0, нужно настроить среду, выполнив команду `usejdbc2` из каталога `sql11ib/java12`. Если нужно, чтобы прикладные программы всегда использовали драйвер JDBC 2.0, можно добавить следующую строку в профиль регистрации (например, `.profile`) или в сценарий инициализации оболочки (например, `.bashrc`, `.cshrc` или `.kshrc`):

```
. sql11ib/java12/usejdbc2
```

Эта команда должна стоять после команды запуска `db2profile`, так как `usejdbc2` должна выполняться после `db2profile`.

Чтобы переключиться назад на драйвер JDBC 1.22, выполните следующую команду из каталога `sql11ib/java12`:

```
. usejdbc1
```

Использование драйвера JDBC 2.0 с хранимыми процедурами и пользовательскими функциями Java

Чтобы использовать драйвер JDBC 2.0 с хранимыми процедурами и пользовательскими функциями Java, необходимо настроить среду для ID пользователя, под которым будут выполняться изолированные пользовательские функции и хранимые процедуры для этого экземпляра. По умолчанию это ID пользователя `db2fenc1`. Чтобы настроить среду для этого ID пользователя, выполните следующие действия:

1. Добавьте следующую строку в профиль (например, `.profile`) или в сценарий инициализации оболочки (например, `.bashrc`, `.cshrc` или `.kshrc`) ID пользователя, под которым будут выполняться изолированные пользовательские функции и хранимые процедуры:

```
. sql11ib/java12/usejdbc2
```

2. Введите следующую команду в командной строке:

```
db2set DB2_USE_JDK12=1
```

Чтобы переключиться назад на поддержку драйвера JDBC 1.2 для пользовательских функций и хранимых процедур Java, выполните следующие действия:

1. Удалите следующую строку из профиля (например, `.profile`) или из сценария инициализации оболочки (например, `.bashrc`, `.cshrc` или `.kshrc`) ID пользователя, под которым будут выполняться изолированные пользовательские функции и хранимые процедуры:

```
. sql11ib/java12/usejdbc2
```

2. Введите следующую команду в командной строке:

```
db2set DB2_USE_JDK12=
```

Если нужно, чтобы прикладные программы всегда использовали драйвер JDBC 2.0, можно добавить следующую строку в профиль регистрации (например, `.profile`) или в сценарий инициализации оболочки (например, `.bashrc`, `.cshrc` или `.kshrc`):

```
. sql11ib/java12/usejdbc2
```

Эта команда должна стоять после команды запуска `db2profile`, так как `usejdbc2` должна выполняться после `db2profile`.

HP-UX

В DB2 for HP-UX с JDK 1.1 хранимые процедуры и пользовательские функции Java не поддерживаются.

Silicon Graphics IRIX

Если при построении прикладных программ SQLJ с типом объектов -o32 с помощью компилятора Java JIT с JDK 1.2.2 транслятор SQLJ сообщает об ошибке сегментации, попробуйте отключить компилятор JIT командой:

```
export JAVA_COMPILER=NONE
```

На Silicon Graphics IRIX для построения программы Java SQLJ необходим JDK 1.2.2.

32-битные операционные системы Windows

Использование драйвера JDBC 2.0 с хранимыми процедурами и пользовательскими функциями Java

Чтобы использовать драйвер JDBC 2.0 с хранимыми процедурами и пользовательскими функциями Java, необходимо настроить среду, выполнив следующие действия:

1. Выполните следующую команду из каталога sqllib\java12:

```
usejdbc2
```
2. Введите следующую команду в командной строке:

```
db2set DB2_USE_JDK12=1
```

Чтобы переключиться назад на поддержку драйвера JDBC 1.2 для пользовательских функций и хранимых процедур Java, выполните следующие действия:

1. Выполните следующую команду из каталога sqllib\java12:

```
usejdbc2
```
2. Введите следующую команду в командной строке:

```
db2set DB2_USE_JDK12=
```

Уровень JDK в OS/2

В OS/2 с версиями JDK 1.1.8 выпуска ранее 09/99 не будут выводиться некоторые сообщения. Убедитесь, что используется самый свежий выпуск JDK Версии 1.1.8.

Глава 5. Building SQL Procedures

Настройка среды процедур SQL

Ниже приводятся дополнительные инструкции по настройке среды DB2 при установке.

Для поддержки процедур SQL необходимо установить клиент разработки программ на сервере. Информацию об установке клиента разработки программ смотрите в книге *Быстрый старт* для вашей платформы. Информацию о компиляторах C и C++, поддерживаемых DB2 на вашей платформе, смотрите в разделе "Supported Software by Platform".

Примечание: В файловой системе OS/2 FAT имя схемы для процедур SQL ограничено 8 символами. Для имен схем длиннее 8 символов необходимо использовать файловую систему HPFS.

Конфигурирование компилятора состоит из двух частей: настройка переменных среды для компилятора и определение команды компиляции. Эти переменные среды задают пути к двоичным файлам, библиотекам и включаемым файлам компилятора. Команда компиляции - это полнофункциональная команда, которую DB2 будет использовать для компиляции файлов на языке C, сгенерированных для процедур SQL.

Задание переменных среды компилятора

Правила конфигурирования среды в OS/2, Windows и в системах на базе UNIX отличаются, как описано ниже. В некоторых случаях никакого конфигурирования не требуется; в других случаях в переменной реестра DB2 DB2_SQLROUTINE_COMPILER_PATH нужно указать путь к исполняемому сценарию, который сконфигурирует данную среду должным образом.

Примечание: Для задания значения этой переменной реестра DB2 можно использовать либо команду **db2set**, либо диалоговое окно Опции построения хранимых процедур SQL. Использование окна Опции построения хранимых процедур позволяет обойтись без физического обращения к серверу баз данных для ввода команды, а также без перезапуска сервера для вступления изменений в силу.

В OS/2:

для IBM VisualAge C++ for OS/2 Версии 3.6:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\ibmcxxo\bin\setenv.cmd"
```

для IBM VisualAge C++ for OS/2 Версии 4:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\ibmcpp40\bin\setenv.cmd"
```

Примечание: В этих командах предполагается, что компилятор C++ установлен на диске c:. Если нужно, измените диск или путь, чтобы отразить положение компилятора C++ в вашей системе.

В 32-битных операционных системах Windows, если переменные среды компилятора заданы как системные переменные, никакого конфигурирования не требуется. Иначе задайте переменную реестра DB2 DB2_SQLROUTINE_COMPILER_PATH DB2:

для Microsoft Visual C++ Версии 5.0:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\devstudio\vc\bin\vcvars32.bat"
```

для Microsoft Visual C++ Версии 6.0:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\Micros~1\vc98\bin\vcvars32.bat"
```

для IBM VisualAge C++ for Windows Версии 3.6:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\ibmcxxw\bin\setenv.bat"
```

для IBM VisualAge C++ for Windows Версии 4:

```
db2set DB2_SQLROUTINE_COMPILER_PATH="c:\ibmcppw40\bin\setenv.bat"
```

Примечание: В этих командах предполагается, что компилятор C++ установлен на диске c:. Если нужно, измените диск или путь, чтобы отразить положение компилятора C++ в вашей системе.

В системах на базе UNIX DB2 сгенерирует файл исполняемого сценария \$HOME/sqlib/function/routine/sr_cpath (который содержит для переменных среды компилятора значения по умолчанию) при первой компиляции хранимой процедуры. Если значения по умолчанию не подходят для вашего компилятора, этот файл можно отредактировать. Другой вариант - указать в переменной реестра DB2 DB2_SQLROUTINE_COMPILER_PATH полное имя другого исполняемого сценария, где задаются желаемые параметры (смотрите вышеприведенные примеры).

Настройка команды компиляции

При установке клиента разработки программ по умолчанию предоставляется команда компиляции, которая работает как минимум для одного из компиляторов, поддерживаемых на каждой платформе:

AIX: IBM C Set++ for AIX, Версии 3.6.6
Solaris: SPARCompiler C++ Версия 4.2 и 5.0
HP-UX: HP-UX C++ Версия A.12.00
Linux: GNU/Linux g++ Версия egcs-2.90.27 980315 (выпуск egcs-1.0.2)
PTX: ptx/C++ Версия 5.2
OS/2: IBM VisualAge C++ for OS/2 Версия 3
Windows NT и Windows 2000: Microsoft Visual C++ Версия 5.0 и 6.0

Чтобы использовать другие компиляторы или чтобы настроить данную команду по умолчанию, нужно указать в переменной реестра DB2

DB2_SQLROUTINE_COMPILE_COMMAND следующую команду:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=команда_компиляции
```

где *команда_компиляции* - команда компиляции C или C++ с опциями и параметрами, требуемыми для создания хранимых процедур.

В команде компиляции при помощи ключевого слова SQLROUTINE_FILENAME замените имена файлов для сгенерированных SQC, C, PDB, DEF, EXP, журнала сообщений и файлов совместно используемых библиотек. Только в AIX: при помощи ключевого слова SQLROUTINE_ENTRY измените имя точки входа.

Примечание: Для задания значения этой переменной реестра DB2 можно использовать либо команду **db2set**, либо диалоговое окно Опции построения хранимых процедур SQL. Использование окна Опции построения хранимых процедур позволяет обойтись без физического обращения к серверу баз данных для ввода команды, а также без перезапуска сервера для вступления изменений в силу.

Ниже показаны значения по умолчанию для DB2_SQLROUTINE_COMPILE_COMMAND для компиляторов C или C++ на поддерживаемых платформах серверов.

AIX

Чтобы использовать IBM C for AIX Версии 3.6.6:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=xlc -H512 -T512 \  
-I$HOME/sqlib/include SQLROUTINE_FILENAME.c -bE:SQLROUTINE_FILENAME.exp \  
-e SQLROUTINE_ENTRY -o SQLROUTINE_FILENAME -L$HOME/sqlib/lib -lc -ldb2
```

Чтобы использовать IBM C Set++ for AIX Версии 3.6.6:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=xlc -H512 -T512 \  
-I$HOME/sqlib/include SQLROUTINE_FILENAME.c -bE:SQLROUTINE_FILENAME.exp \  
-e SQLROUTINE_ENTRY -o SQLROUTINE_FILENAME -L$HOME/sqlib/lib -lc -ldb2
```

Это команда компилятора по умолчанию, используемая, если не задана переменная реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND.

Примечание: Для компиляции в AIX 64-битных процедур SQL добавьте в показанные выше команды опцию -q64.

Чтобы использовать IBM VisualAge C++ for AIX Версии 4:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld"
```

Если после команды vacbld не задан файл конфигурации, DB2 создаст следующий файл конфигурации по умолчанию при первой попытке создания какой-либо процедуры SQL:

```
$HOME/sql1lib/function/routine/sqlproc.icc
```

Если нужно использовать другой файл конфигурации, можно задать его при задании переменной реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld  
%DB2PATH%/function/sqlproc.icc"
```

HP-UX

Чтобы использовать компилятор HP C Версии A.11.00.03:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=cc +DAportable +ul -Aa +z \  
-I$HOME/sql1lib/include -c SQLROUTINE_FILENAME.c; \  
ld -b -o SQLROUTINE_FILENAME SQLROUTINE_FILENAME.o \  
-L$HOME/sql1lib/lib -ldb2
```

Чтобы использовать HP-UX C++ Версии A.12.00:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=CC +DAportable +a1 +z -ext \  
-I$HOME/sql1lib/include -c SQLROUTINE_FILENAME.c; \  
ld -b -o SQLROUTINE_FILENAME SQLROUTINE_FILENAME.o \  
-L$HOME/sql1lib/lib -ldb2
```

Это команда компилятора по умолчанию, используемая, если не задана переменная реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND.

Linux

Чтобы использовать GNU/Linux gcc Версии 2.7.2.3:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=cc \  
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \  
-shared -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib -ldb2
```

Чтобы использовать GNU/Linux g++ Версии egcs-2.90.27 980315 (выпуск egcs-1.0.2):

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=g++ \  
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \  
-shared -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib -ldb2
```

Это команда компилятора по умолчанию, используемая, если не задана переменная реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND.

PTX

Чтобы использовать ptx/C Версии 4.5:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=cc -KPIC \  
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \  
-G -o SQLROUTINE_FILENAME.so -L$HOME/sql1lib/lib -ldb2 ; \  
cp SQLROUTINE_FILENAME.so SQLROUTINE_FILENAME
```

Чтобы использовать ptx/C++ Версии 5.2:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=c++ -KPIC \  
-D RWSTD_COMPILE_INSTANTIATE=0 \  
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \  
-G -o SQLROUTINE_FILENAME.so -L$HOME/sql1lib/lib -ldb2 ; \  
cp SQLROUTINE_FILENAME.so SQLROUTINE_FILENAME
```

Это команда компилятора по умолчанию, используемая, если не задана переменная реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND.

OS/2

Чтобы использовать IBM VisualAge C++ for OS/2 Версии 3:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="icc -Ge- -Gm+ -w2 \  
-I%DB2PATH%\include SQLROUTINE_FILENAME.c \  
/B\ /NOFREE /NOI /ST:64000" SQLROUTINE_FILENAME.def \  
%DB2PATH%\lib\db2api.lib"
```

Это команда компилятора по умолчанию, используемая, если не задана переменная реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND.

Чтобы использовать IBM VisualAge C++ for OS/2 Версии 4:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld"
```

Если после команды vacbld не задан файл конфигурации, DB2 создаст следующий файл конфигурации по умолчанию при первой попытке создания какой-либо процедуры SQL:

```
%DB2PATH%\function\routine\sqlproc.icc
```

Если нужно использовать другой файл конфигурации, можно задать его при задании переменной реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld \  
%DB2PATH%\function\sqlproc.icc"
```

Solaris

Чтобы использовать SPARCompiler C Версий 4.2 и 5.0:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=cc -xarch=v8plusa -Kpic \  
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \  
-G -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib \  
-R$HOME/sql1lib/lib -ldb2
```

Чтобы использовать SPARCompiler C++ Версий 4.2 и 5.0:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=CC -xarch=v8plusa -Kpic \  
-I$HOME/sql1lib/include SQLROUTINE_FILENAME.c \  
-G -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib \  
-R$HOME/sql1lib/lib -ldb2
```

Это команда компилятора по умолчанию, используемая, если не задана переменная реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND.

Примечания:

1. В команду компилятора по умолчанию добавлена опция компилятора `-xarch=v8plusa`. Дополнительную информацию о том, почему была добавлена эта опция, смотрите в разделе “Глава 12. Building Solaris Applications” на стр. 241.
2. Для компиляции в Solaris 64-битных процедур SQL удалите опцию `-xarch=v8plusa` и добавьте в показанные выше команды опцию `-xarch=v9`.

Windows NT и Windows 2000

Примечание: В Windows 98 и Windows 95 процедуры SQL не поддерживаются.

Чтобы использовать Microsoft Visual C++ Версий 5.0 и 6.0:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=c1 -Od -W2 /TC -D_X86_=1
-I%DB2PATH%\include SQLROUTINE_FILENAME.c /link -dll
-def:SQLROUTINE_FILENAME.def /out:SQLROUTINE_FILENAME.dll
%DB2PATH%\lib\db2api.lib
```

Это команда компилятора по умолчанию, используемая, если не задана переменная реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND.

Чтобы использовать IBM VisualAge C++ for Windows Версии 3.6:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="ilib /GI
SQLROUTINE_FILENAME.def &icc -Ti -Ge- -Gm+ -W2
-I%DB2PATH%\include SQLROUTINE_FILENAME.c
/B\"/ST:64000 /PM:VIO /DLL\" SQLROUTINE_FILENAME.exp
%DB2PATH%\lib\db2api.lib"
```

Чтобы использовать IBM VisualAge C++ for Windows Версии 4:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld"
```

Если после команды vacbld не задан файл конфигурации, DB2 создаст следующий файл конфигурации по умолчанию при первой попытке создания какой-либо процедуры SQL:

```
%DB2PATH%\function\routine\sqlproc.icc
```

Если нужно использовать другой файл конфигурации, можно задать его при задании переменной реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="vacbld
%DB2PATH%\function\sqlproc.icc"
```

Чтобы вернуться к опциям компилятора по умолчанию, с помощью следующей команды задайте пустое значение для переменной реестра DB2 DB2_SQLROUTINE_COMPILE_COMMAND:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND=
```

Удаление промежуточных файлов

Необходимо вручную удалить промежуточные файлы, которые могут остаться при неудачном завершении процедуры SQL. Эти файлы находятся в следующих каталогах:

UNIX \$DB2PATH/function/routine/sqlproc/\$DATABASE/\$SCHEMA/tmp

где \$DB2PATH - каталог, в котором был создан этот экземпляр, \$DATABASE представляет имя базы данных, а \$SCHEMA - имя схемы, с которыми были созданы процедуры SQL.

OS/2 и Windows

```
%DB2PATH%\function\routine\sqlproc%\DATABASE%\%SCHEMA%\tmp
```

где %DATAPATH% - каталог, в котором был создан этот экземпляр, %DATABASE% - имя базы данных, а %SCHEMA% - имя схемы, с которыми были созданы процедуры SQL.

Резервное копирование и восстановление

При создании процедуры SQL сгенерированная совместно используемая библиотека DLL сохраняется в таблице каталога, если она меньше 2 Мбайт. При резервном копировании и восстановлении базы данных любая процедура SQL со сгенерированной библиотекой DLL меньше 2 Мбайт будет скопирована и восстановлена вместе с версией, хранимой в этой таблице каталога. Если имеются процедуры SQL со сгенерированной библиотекой DLL больше 2 Мбайт, убедитесь, что вместе с резервным копированием и восстановлением базы данных происходит также копирование и восстановление файловой системы. Если это не так, надо воссоздать совместно используемую библиотеку DLL процедуры SQL вручную, воспользовавшись исходной библиотекой в таблице каталога `syscat.procedures`.

Примечание: Во время восстановления базы данных все выполняемые процедуры SQL в файловой системе, принадлежащей восстанавливаемой базе данных, будут удалены. Если для параметра конфигурации создания индекса (`indexrec`) задано значение `RESTART`, все выполняемые процедуры SQL будут извлечены из таблицы каталога и помещены назад в файловую систему при следующем соединении. Иначе выполняемые программы SQL будут извлечены при первом выполнении процедур SQL.

Выполняемые программы помещаются назад в следующий каталог:

```
UNIX $DB2PATH/function/routine/sqlproc/$DATABASE
```

где `$DB2PATH` - каталог, в котором был создан этот экземпляр, а `$DATABASE` - имя базы данных, с которым были созданы процедуры SQL.

OS/2 и Windows

```
%DB2PATH%\function\routine\sqlproc\%DATABASE%
```

где `%DB2PATH%` - каталог, в котором создавался экземпляр, а `%DATABASE%` - имя базы данных, с которым создавались процедуры SQL.

Создание процедур SQL

При разработке процедур SQL задайте для параметра конфигурации менеджера баз данных `KEEPDARI` значение `'NO'`. Если процедура SQL остается загруженной после выполнения, могут возникнуть проблемы при отбрасывании и создании новой хранимой процедуры с тем же именем, так как нельзя будет обновить библиотеку и удалить выполняемые файлы из файловой системы. Могут также возникнуть проблемы при попытке выполнить откат изменений или отбросить базу данных, так как нельзя будет удалить выполняемые файлы.

Дополнительную информацию о задании параметра `KEEPDARI` смотрите в разделе `'Updating the Database Manager Configuration File'` Главы 2 "Setup" руководства `'Application Building Guide'`.

Примечание: Процедуры SQL не поддерживают для параметров следующие типы данных:

- `LONG VARGRAPHIC`
- Двоичный большой объект (`BLOB`)
- Символьный большой объект (`CLOB`)
- Двухбайтный символьный большой объект (`DBCLOB`)

Вызов хранимых процедур

Первый абзац в разделе `'Using the CALL Command'` должен выглядеть так:

Для использования команды CALL нужно ввести имя хранимой процедуры, все параметры IN или INOUT, а также '?' вместо каждого параметра OUT. Подробное описание синтаксиса команды CALL смотрите в разделе "CALL" на стр. 110.

Распространение скомпилированных процедур SQL

Примечание: Чтобы распространить скомпилированные процедуры SQL по серверам DB2 надо выполнить описанные ниже шаги на каждом сервере DB2, который служит как источник или назначение скомпилированной процедуры SQL:

Шаг 1. Установите FixPak 3

Шаг 2. Введите команду **db2updv7**, чтобы разрешить DB2 извлекать и устанавливать скомпилированные процедуры SQL:

```
db2updv7 -d имя_базы_данных
```

Когда вы определяете процедуру SQL, она преобразуется в программу C, прекомпилируется, связывается с базой данных назначения, компилируется и компоуется для создания совместно используемой библиотеки. Для шагов компиляции и компоновки нужно, чтобы на компьютере сервера баз данных был доступен компилятор C или C++. Однако определив процедуру SQL, вы можете распространить ее в скомпилированной форме на базы данных DB2 той же платформы, не обязательно имеющие доступ к компилятору C или C++. DB2 позволяет пользователю извлекать процедуры SQL в скомпилированной форме из одной базы данных и устанавливать их в другую базу данных.

Для извлечения и установки в DB2 есть и интерфейс командной строки, и программный интерфейс. Интерфейс командной строки состоит из двух команд: GET ROUTINE и PUT ROUTINE. Программный интерфейс состоит из двух встроенных хранимых процедур: GET_ROUTINE_SAR и PUT_ROUTINE_SAR. Дополнительную информацию об интерфейсе командной строки смотрите в справочнике Command Reference. Дополнительную информацию о программном интерфейсе смотрите в справочнике SQL Reference.

Чтобы распространить скомпилированную процедуру SQL с одного сервера баз данных на другой, выполните следующие действия:

Шаг 1. Разработайте программу и определите, как ее часть, процедуры SQL.

Шаг 2. Проверив эти процедуры, извлеките скомпилированную версию каждой процедуры в отдельный файл. Дополнительную информацию смотрите в описании команды GET ROUTINE в справочнике Command Reference или хранимой процедуры GET_ROUTINE_SAR в справочнике SQL Reference.

Шаг 3. Вставьте скомпилированную версию каждой процедуры на каждый сервер либо при помощи команды PUT ROUTINE, либо вызвав хранимую процедуру PUT_ROUTINE_SAR; используйте файлы, созданные на шаге 2. На каждом сервере баз данных должна быть та же операционная система и тот же уровень DB2.

Глава 7. Building HP-UX Applications.

HP-UX C

В разделе "Multi-threaded Applications" в файле сценария bldmt были изменены опции компиляции. Новая версия находится в каталоге sqllib/samples/c.

HP-UX C++

Для компилятора HP aC++ в сценариях построения переменная компилятора C++ CC была заменена на aCC. Исправленные сценарии построения находятся в каталоге `sqllib/samples/cpp`.

Для построения хранимых процедур и пользовательских функций при помощи компилятора aCC надо использовать опцию компиляции `"+u1"`. Эта опция разрешает обращаться к невыровненным данным. В примеры сценариев построения, поставляемых с DB2 for HP-UX (`bldsrv` и `bldudf`), а также в `make`-файл примера эта опция не внесена. Перед использованием этих примеров это необходимо сделать. Новый шаг компиляции для сценариев `bldsrv` и `bldudf` должен выглядеть так:

```
aCC +DAportable +u1 -Aa +z -ext -I$DB2PATH/include -c $1.C
```

В разделе "Multi-threaded Applications" в файле сценария `bldmt` были изменены опции компиляции. Новая версия находится в каталоге `sqllib/samples/cpp`.

Глава 10. Building PTX Applications

ptx/C++

Для построения хранимых процедур и пользовательских функций при компоновке библиотек надо использовать опцию `-shared`. Необходимо изменить файл `makefile`, сценарии построения `bldsrv` и файл `bldudf` в каталоге `sqllib/samples`, добавив в них эту опцию, как показано ниже для шага компоновки из `bldsrv`:

```
++ -shared -G -o $1 $1.o -L$DB2PATH/lib -ldb2
```

Глава 12. Building Solaris Applications

SPARCompiler C++

Ошибки выполнения прикладных программ C/C++ и процедур SQL в Solaris

Если при использовании компилятора Sun WorkShop Compiler C/C++ вы столкнулись с проблемами с выполняемыми файлами и получаете сообщения об ошибках такого типа:

1. `syntax error at line 1: '(' unexpected`
2. `ksh: <имя программы>: cannot execute (где имя программы - имя скомпилированного выполняемого файла)`

возможно, эта ошибка вызвана тем, что компилятор создает неправильные выполняемые файлы при компоновке с `libdb2.so`. Чтобы исправить эту ошибку, можно добавить следующую опцию компилятора в команды компиляции и компоновки:

```
-xarch=v8plusa
```

Например, при компиляции примера прикладной программы `dynamic.sqc`:

```
embprep dynamic sample
embprep utilemb sample
cc -c utilemb.c -xarch=v8plusa -I/export/home/db2inst1/sqllib/include
cc -o dynamic dynamic.c utilemb.o -xarch=v8plusa -I/export/home/db2inst1/sqllib/include \
-L/export/home/db2inst1/sqllib/lib -R/export/home/db2inst1/sqllib/lib -l db2
```

Примечания:

1. Если в системе Solaris используются процедуры SQL и строка компиляции задается в переменной профиля DB2_SQLROUTINE_COMPILE_COMMAND, в ней должна быть задана указанная выше опция компилятора. Команда компиляции по умолчанию содержит эту опцию:

```
db2set DB2_SQLROUTINE_COMPILE_COMMAND="cc -# -Kpic -xarch=v8plusa -I$HOME/sql1lib/include \  
SQLROUTINE_FILENAME.c -G -o SQLROUTINE_FILENAME -L$HOME/sql1lib/lib -R$HOME/sql1lib/lib -ldb2
```

2. Для компиляции в Solaris 64-битных процедур SQL удалите опцию `-xarch=v8plusa` и добавьте в показанные выше команды опцию `-xarch=v9`.

VisualAge C++ Версии 4.0 в OS/2 и Windows

Примечание: Это исправление для раздела "VisualAge C++ Version 4.0" Главы 6 "Building AIX Applications". Этот раздел содержит информацию, общую для систем AIX, OS/2 и 32-битных операционных систем Windows.

В OS/2 и Windows вместо команды `export` используйте команду `set`. Например, `set CLI=tbinfo`.

В разделе "DB2 CLI Applications", подраздел "Building and Running Embedded SQL Applications" для OS/2 и Windows вместо файла `cli.icc` надо использовать файл `cliapi.icc`, так как компоновка библиотеки `db2api.lib` для прикладных программ со встроенным SQL должна выполняться `cliapi.icc`.

Глава 26. Application Development Guide

Написание хранимых процедур автоматизации OLE

В следующем абзаце (второй абзац раздела "Writing OLE automation Stored Procedures") пропущено последнее предложение:

Написав код объекта автоматизации OLE, необходимо зарегистрировать методы этого объекта в качестве хранимых процедур при помощи оператора CREATE PROCEDURE. Чтобы зарегистрировать хранимую процедуру автоматизации OLE, используйте оператор CREATE PROCEDURE с условием LANGUAGE OLE. Внешнее имя состоит из ID программы OLE, идентифицирующего этот объект автоматизации OLE, и имени метода, разделенных символом ! (восклицательный знак). Объект автоматизации OLE должен быть реализован как внутривещественный сервер (.DLL).

Глава 6. Common DB2 Application Techniques

Генерирование последовательных значений

Генерирование последовательных значений - обычная проблема при разработке прикладных программ для баз данных. Наилучшее решение этой проблемы - использование объектов последовательности и выражений последовательности на языке SQL. Каждый *объект последовательности* - это объект базы данных с уникальным именем, к которому можно обращаться только с помощью выражений последовательности. Существует два *выражения последовательности*: выражение PREVVAL и выражение NEXTVAL. Выражение PREVVAL возвращает последнее из сгенерированных для предыдущего оператора значений заданной последовательности. Выражение NEXTVAL увеличивает значение объекта последовательности и возвращает его новое значение.

Чтобы создать объект последовательности, выполните оператор CREATE SEQUENCE. Например, чтобы создать объект последовательности под названием id_values с атрибутами по умолчанию, используйте следующий оператор:

```
CREATE SEQUENCE id_values
```

Чтобы вывести текущее значение этого объекта последовательности, выполните оператор VALUES с выражением PREVVAL:

```
VALUES PREVVAL FOR id_values
```

```
1
-----
1
```

1 record(s) selected.

Текущее значение объекта последовательности можно получать несколько раз, причем возвращаемое значение не изменится, пока не будет выполнено выражение NEXTVAL. В следующем примере выражение PREVVAL возвращает значение 1 до тех пор, пока выражение NEXTVAL не увеличит значение объекта последовательности:

```
VALUES PREVVAL FOR id_values
```

```
1
-----
1
```

```

1 record(s) selected.
VALUES PREVVAL FOR id_values
1
-----
1

```

```

1 record(s) selected.
VALUES NEXTVAL FOR id_values
1
-----
2

```

```

1 record(s) selected.
VALUES PREVVAL FOR id_values
1
-----
2

```

```

1 record(s) selected.

```

Чтобы изменить значение столбца на следующее значение объекта последовательности, включите выражение NEXTVAL в оператор UPDATE, как показано ниже:

```

UPDATE staff
  SET id = NEXTVAL FOR id_values
  WHERE id = 350

```

Чтобы вставить новую строку в таблицу, используя следующее значение объекта последовательности, включите выражение NEXTVAL в оператор INSERT, как показано ниже:

```

INSERT INTO staff (id, name, dept, job)
  VALUES (NEXTVAL FOR id_values, 'Kandil', 51, 'Mgr')

```

Дополнительную информацию о выражениях PREVVAL и NEXTVAL смотрите в справочнике *SQL Reference*.

Управление поведением последовательности

Поведение объектов последовательности можно настроить в соответствии с потребностями вашей прикладной программы. Атрибуты объекта последовательности изменяют операторы CREATE SEQUENCE (создание нового объекта последовательности) и ALTER SEQUENCE (изменение существующего объекта последовательности). Ниже приводятся некоторые из доступных вам атрибутов объекта последовательности:

Тип данных

Условие AS оператора CREATE SEQUENCE задает числовой тип данных объекта последовательности. Тип данных, как указано в приложении “SQL Limits” справочника *SQL Reference*, определяет возможные минимальное и максимальное значения объекта последовательности. Тип данных объекта последовательности нельзя изменить; вместо этого надо отбросить объект последовательности при помощи оператора DROP SEQUENCE и выполнить оператор CREATE SEQUENCE с новым типом данных.

Начальное значение

Условие START WITH оператора CREATE SEQUENCE задает начальное

значение объекта последовательности. Условие RESTART WITH оператора ALTER SEQUENCE возвращает значение объекта последовательности к заданному значению.

Минимальное значение

Условие MINVALUE задает минимальное значение объекта последовательности.

Максимальное значение

Условие MAXVALUE задает максимальное значение объекта последовательности.

Значение приращения

Условие INCREMENT BY задает значение, добавляемое к объекту последовательности каждым выражением NEXTVAL. Чтобы значение объекта последовательности уменьшалось, задайте в условии отрицательную величину.

Защивание последовательности

Условие CYCLE заставляет значение объекта последовательности после достижения минимума или максимума вернуться при очередном выражении NEXTVAL к исходному значению.

Например, чтобы создать объект последовательности под названием id_values, с начальным значением 0, максимальным значением 1000, увеличением значения на 2 с каждым выражением NEXTVAL и возвращением к начальному значению по достижении максимального, используйте оператор:

```
CREATE SEQUENCE id_values
  START WITH 0
  INCREMENT BY 2
  MAXVALUE 1000
  CYCLE
```

Дополнительную информацию об операторах CREATE SEQUENCE и ALTER SEQUENCE смотрите в справочнике *SQL Reference*.

Улучшение производительности с помощью объектов последовательности

Как и для столбцов идентификации, использование объектов последовательности для генерирования значений в целом улучшает производительность ваших прикладных программ в сравнении с другими подходами. Вместо объектов последовательности можно было бы создать таблицу с одним столбцом, где хранится текущее значение, и увеличивать это значение при помощи триггера или под управлением прикладной программы. В распределенной среде, где программы одновременно обращаются к одностолбцовой таблице, блокировка, необходимая для принудительного последовательного доступа к этой таблице, может серьезно повлиять на производительности.

Объекты последовательности позволяют избежать проблем с блокировкой, связанных с использованием одностолбцовой таблицы; можно кэшировать значения в памяти, что улучшает время ответа DB2. Чтобы максимизировать производительность прикладных программ, использующих объекты последовательности, убедитесь, что ваш объект последовательности кэширует достаточное количество значений. Условие CACHE операторов CREATE SEQUENCE и ALTER SEQUENCE задает максимальное количество значений последовательности, которые DB2 генерирует и хранит в памяти.

Если ваш объект последовательности должен генерировать упорядоченные значения без разрывов между ними в случае системной ошибки или деактивации базы данных,

задайте в операторе CREATE SEQUENCE условия ORDER и NO CACHE. Условие NO CACHE гарантирует отсутствие разрывов между сгенерированными значениями за счет некоторого снижения производительности ваших программ, так как при этом объект последовательности записывает в журнал базы данных информацию о каждом новом сгенерированном значении.

Сравнение объектов последовательности и столбцов идентификации

Хотя у объектов последовательностей и столбцов идентификации при работе с прикладными программами DB2 имеют сходное назначение, между ними есть ряд существенных различий:

- Столбец идентификации автоматически генерирует значения для столбца только одной таблицы. Объект последовательности генерирует последовательные значения, которые можно использовать в любом операторе SQL.
- Столбец идентификации генерирует заведомо уникальные значения. Включение условия CYCLE в операторы CREATE SEQUENCE или ALTER SEQUENCE разрешает объекту последовательности генерировать повторяющиеся значения.

Глава 7. Stored Procedures

Ошибка типа DECIMAL в Java-процедурах Linux

Эта ошибка происходит из-за того, что комплект разработчика IBM для Java не создает связей для своих библиотек в каталоге /usr/lib. Модель защиты для процедур DB2 не разрешает им доступ к библиотекам за пределами стандартных библиотек системы. Чтобы включить в Linux поддержку DECIMAL в Java-процедурах, выполните следующие действия:

1. Создайте символические связи из библиотек комплекта разработчика IBM для Java в /usr/lib/, введя следующую команду с полномочиями root:

Для комплекта разработчика IBM для Java 1.1.8:

```
ln -sf /usr/jdk118/lib/linux/native_threads/* /usr/lib/
```

Для комплекта разработчика IBM для Java 1.3:

```
ln -sf /opt/IBMJava2-13/jre/bin/*.so /usr/lib/
```

2. Введите команду **ldconfig**, чтобы исправить список библиотек уровня системы.

Использование указателей в рекурсивных хранимых процедурах

Во избежание ошибок при использовании процедур SQL или хранимых процедур, написанных на встроенном SQL перед использованием рекурсивного оператора CALL закройте все открытые указатели.

Например, предположим, что в хранимой процедуре MYPROC содержится следующий фрагмент программного кода:

```
OPEN c1;  
CALL MYPROC();  
CLOSE c1;
```

При вызове MYPROC DB2 возвратит ошибку, поскольку указатель c1 остается открытым, когда MYPROC выполняет рекурсивный оператор CALL. Конкретная ошибка, возвращаемая DB2, зависит от действия, выполняемого MYPROC над указателем.

Для успешного вызова MYPROC перепишите ее так, чтобы она закрывала все открытые указатели до вложенного оператора CALL, как показано в следующем примере:

```
OPEN c1;
CLOSE c1;
CALL MYPROC();
```

Во избежание ошибки закрывайте все открытые указатели перед использованием встроенного оператора CALL.

Глава 12. Working with Complex Objects: User-Defined Structured Types

Вставка в столбцы атрибутов структурированного типа

Для встроенных статических операторов SQL применяется следующее правило: Чтобы вставить атрибут пользовательского структурированного типа в столбец того же типа данных, что и этот атрибут, заключите в круглые скобки переменную хоста, представляющую экземпляр этого типа, и после закрывающей скобки добавьте две точки и имя атрибута. Например, рассмотрим такую ситуацию:

- PERSON_T - структурированный тип, содержащий атрибут NAME типа VARCHAR(30).
- T1 - таблица, содержащая столбец C1 типа VARCHAR(30).
- personhv - переменная хоста, объявленная для типа PERSON_T в языке программирования.

Правильный синтаксис для вставки атрибута NAME в столбец C1:

```
EXEC SQL INSERT INTO T1 (C1) VALUES (:personhv)..NAME
```

Глава 20. Programming in C and C++

Информация в следующей таблице дополняет информацию, включенную в главу 7, "Stored Procedures", в главу 15, "Writing User-Defined Functions and Methods" и в главу 20, "Programming in C and C++". В этой таблице перечислены поддерживаемые отображения между типами данных SQL и типами данных C для хранимых процедур, пользовательских функций и методов.

Типы C/C++ для хранимых процедур, пользовательских функций и методов

Таблица 9. Типы данных SQL, соответствующие объявлениям C/C++

Тип столбца SQL	Тип данных C/C++	Описание типа столбца SQL
SMALLINT (500 или 501)	sqlint16	16-битное целое со знаком
INTEGER (496 или 497)	sqlint32	32-битное целое со знаком
BIGINT (492 или 493)	sqlint64	64-битное целое со знаком
REAL (480 или 481)	float	Число с плавающей точкой, одинарной точности
DOUBLE (480 или 481)	double	Число с плавающей точкой, двойной точности
DECIMAL(p, s) (484 или 485)	Не поддерживается.	Чтобы передать значение типа DECIMAL, задайте для этого параметра тип данных, в который может быть преобразован тип DECIMAL (например, CHAR или DOUBLE), и явно преобразуйте аргумент в этот тип данных.
CHAR(n) (452 или 453)	char[n+1] где n достаточно велико, чтобы вместить данные 1<=n<=254	Символьная строка фиксированной длины с пустым символом-ограничителем

Таблица 9. Типы данных SQL, соответствующие объявлениям C/C++ (продолжение)

Тип столбца SQL	Тип данных C/C++	Описание типа столбца SQL
CHAR(<i>n</i>) FOR BIT DATA (452 или 453)	char[<i>n</i> +1] где <i>n</i> достаточно велико, чтобы вместить данные 1<= <i>n</i> <=254	Символьная строка фиксированной длины
VARCHAR(<i>n</i>) (448 или 449) (460 или 461)	char[<i>n</i> +1] где <i>n</i> достаточно велико, чтобы вместить данные 1<= <i>n</i> <=32 672	Строка переменной длины с пустым символом-ограничителем
VARCHAR(<i>n</i>) FOR BIT DATA (448 или 449)	struct { sqluint16 length; char[<i>n</i>] }; 1<= <i>n</i> <=32 672	Символьная строка переменной длины без пустого символа-ограничителя
LONG VARCHAR (456 или 457)	struct { sqluint16 length; char[<i>n</i>] }; 32 673<= <i>n</i> <=32 700	Символьная строка переменной длины без пустого символа-ограничителя
CLOB(<i>n</i>) (408 или 409)	struct { sqluint32 length; char data[<i>n</i>]; }; 1<= <i>n</i> <=2 147 483 647	Символьная строка переменной длины без пустого символа-ограничителя с 4-байтным индикатором длины строки
BLOB(<i>n</i>) (404 или 405)	struct { sqluint32 length; char data[<i>n</i>]; }; 1<= <i>n</i> <=2 147 483 647	Двоичная строка переменной длины без пустого символа-ограничителя с 4-байтным индикатором длины строки
DATE (384 или 385)	char[11]	Символьный формат с пустым символом-ограничителем
TIME (388 или 389)	char[9]	Символьный формат с пустым символом-ограничителем
TIMESTAMP (392 или 393)	char[27]	Символьный формат с пустым символом-ограничителем
Примечание: Следующие типы данных доступны только в среде DBCS или EUC при прекомпиляции с опцией WCHARTYPE NOCONVERT.		
GRAPHIC(<i>n</i>) (468 или 469)	sqldbchar[<i>n</i> +1] где <i>n</i> достаточно велико, чтобы вместить данные 1<= <i>n</i> <=127	Строка двухбайтных символов фиксированной длины с пустым символом-ограничителем
VARGRAPHIC(<i>n</i>) (400 или 401)	sqldbchar[<i>n</i> +1] где <i>n</i> достаточно велико, чтобы вместить данные 1<= <i>n</i> <=16 336	Строка двухбайтных символов переменной длины без пустого символа-ограничителя
LONG VARGRAPHIC (472 или 473)	struct { sqluint16 length; sqldbchar[<i>n</i>] }; 16 337<= <i>n</i> <=16 350	Строка двухбайтных символов переменной длины без пустого символа-ограничителя
DBCLOB(<i>n</i>) (412 или 413)	struct { sqluint32 length; sqldbchar data[<i>n</i>]; }; 1<= <i>n</i> <=1 073 741 823	Символьная строка переменной длины без пустого символа-ограничителя с 4-байтным индикатором длины строки

Приложение В. Примеры программ

В раздел "Object Linking and Embedding Samples" надо добавить:

salarycltvc Написанный на Visual C++ пример CLI DB2, вызывающий хранимую процедуру salarysrv, написанную на Visual Basic.

SALSVADO Пример хранимой процедуры автоматизации OLE (SALSVADO) SALCLADO и клиента (SALCLADO), реализованного на 32-битном Visual Basic и ADO, который вычисляет средний оклад в таблице staff2.

Активация встраиваемых модулей проектов и инструментов IBM DB2 Universal Database для Microsoft Visual C++

Перед выполнением команды **db2vccmd** (шаг 1) убедитесь, что Visual C++ был по крайней мере один раз запущен и остановлен с вашим текущим ID регистрации. При первом запуске Visual C++ для вашего ID пользователя создается профиль; именно он обновляется командой **db2vccmd**. Если попытаться выполнить команду **db2vccmd**, не запустив перед этим один раз Visual C++, можно получить сообщение об ошибке такого вида:

```
"Registering DB2 Project add-in ...Failed! (rc = 2)"
```

Сигнатура метода Java в процедурах и функциях с PARAMETER STYLE JAVA

Если после имени метода Java в условии EXTERNAL NAME оператора CREATE PROCEDURE или CREATE FUNCTION задана сигнатура метода Java, она должна соответствовать отображению типов Java по умолчанию для сигнатуры, заданной после имени процедуры или функции. Например, отображению Java по умолчанию для типа SQL INTEGER - "int", а не "java.lang.Integer".

Провайдер OLE DB IBM DB2

Установка в DB2 IBM Версии 7.1 FixPak 1 или более нового устраняет причину следующей ошибки DB2:

Тестирование соединения закончилось неудачно из-за ошибки при инициализации провайдера. Провайдер OLE DB IBM в настоящее время недоступен. Дополнительную информацию смотрите в файле readme.

Особенности языка/Программирование на Java/Создание прикладных программ и апплетов Java/Поддержка апплетов в Java

Важно, чтобы уровень FixPak файла db2java.zip, используемого апплетом Java, совпадал с уровнем сервера апплетов JDBC. При нормальных условиях db2java.zip загружается с того сервера Web, на котором запущен сервер апплетов JDBC, как показано на рисунке 22 в этой книге. Это гарантирует совпадение уровней. Если же в вашей конфигурации есть апплет Java, загружающий db2java.zip из другого положения, может произойти несовпадение. До FixPak 2 это может вызывать непредвиденные ошибки. После применения FixPak 2 совпадение уровней FixPak для этих двух файлов принудительно проверяется во время соединения. При обнаружении несовпадения соединение не устанавливается, и клиент получает одну из следующих исключительных ситуаций:

- Если уровень db2java.zip - FixPak 2 или более новый:

```
COM.ibm.db2.jdbc.DB2Exception: [IBM] [Драйвер JDBC]
CLI0621E Неподдерживаемая конфигурация сервера JDBC.
```

- Если уровень db2java.zip раньше, чем FixPak 2:
COM.ibm.db2.jdbc.DB2Exception: [IBM] [Драйвер JDBC]
CLI0601E Неправильный хэндл оператора или оператор закрыт.
SQLSTATE=S1000

При несовпадении сервер апплетов JDBC записывает в файл jdbcerr.log одно из следующих сообщений:

- Если уровень сервера апплетов JDBC - FixPak 2 или более новый:
jdbcFSQLConnect: JDBC Версии сервера апплетов и клиента (db2java.zip) не совпадают. Невозможно
- Если уровень сервера апплетов JDBC раньше, чем FixPak 2:
jdbcServiceConnection(): Получено неверное требование., einfo= 0

Глава 27. CLI Guide and Reference

Функции Unicode CLI и SQL_C_WCHAR поддерживаются только в AIX

Функции Unicode CLI принимают в своих аргументах указатели на символьные строки или на SQLPOINTER. Строки аргументов имеют формат UCS2. Эти функции реализованы как функции с суффиксом W.

В функциях Unicode, возвращающих или получающих строки, в аргументах длины передается число символов. Для функций, возвращающих информацию о длине для данных сервера, размер и точность вывода на экран описываются числом символов. Если длина может описываться как длину строки, так и длину нестроковых данных, она описывается в числе байтов. Прототипы функций для функций Unicode можно найти в файле `sqlcli.h`.

Список функций Unicode:

```
SQLColAttributeW  
SQLColAttributesW  
SQLColumnPrivilegesW  
SQLColumnsW  
SQLConnectW  
SQLDataSourcesW  
SQLDescribeColW  
SQLDriverConnectW  
SQLBrowseConnectW  
SQLErrorW  
SQLExecDirectW  
SQLForeignKeysW  
SQLGetCursorNameW  
SQLGetInfoW  
SQLNativeSqlW  
SQLPrepareW  
SQLPrimaryKeysW  
SQLProcedureColumnsW  
SQLProceduresW  
SQLSetCursorNameW  
SQLSpecialColumnsW  
SQLStatisticsW  
SQLTablePrivilegesW  
SQLTablesW  
SQLGetDiagFieldW  
SQLGetDiagRecW  
SQLSetConnectAttrW  
SQLSetStmtAttrW  
SQLGetDescFieldW  
SQLSetDescFieldW
```

Прикладную программу можно написать так, чтобы ее можно было скомпилировать как прикладную программу Unicode или как прикладную программу ANSI. Прикладная программа компилируется как прикладная программа Unicode, если включено объявление (`define`) `UNICODE`. В этом случае символьные типы данных могут быть объявлены как `SQL_C_TCHAR`. В файле `sqlcli.h` есть специальная макрокоманда. Эта макрокоманда вставляет `SQL_C_WCHAR`, если прикладная программа компилируется как программа Unicode, или `SQL_C_CHAR`, если программа компилируется как программа ANSI. Прототипы функций для функций Unicode можно найти в файле `sqlcli.h`. Если прикладная программа компилируется

с включенным объявлением UNICODE, для вызовов функций без суффикса W будут подставляться вызовы функций с суффиксом W.

Вызовы функций Unicode и ANSI нельзя использовать одновременно.

Все типы данных SQL, которые могут быть преобразованы в SQL_C_CHAR, могут также быть преобразованы в SQL_C_WCHAR; верно и обратное.

Применяются следующие ограничения:

- С точки зрения ODBC, CLI не является драйвером UNICODE. Однако поскольку функция SQLConnectW не экспортируется, она отображается на SQLConnectWInt в файле sqlcli1.h.
- В настоящее время функции Unicode и SQL_C_WCHAR поддерживаются только в AIX. Чтобы использовать функции Unicode CLI и SQL_C_WCHAR в прикладных программах для AIX, используйте файл sqlcli1.h и компилируйте программу с объявлением UNICODE. Для прикладных программ Windows NT, которым требуются функции Unicode и SQL_C_WCHAR, используйте драйвер ODBC 3.5. Менеджер драйверов ODBC 3.5 будет считать драйвер CLI DB2 UDB драйвером ANSI. Менеджер драйверов ODBC 3.5 будет преобразовывать вызовы функций Unicode (с суффиксом W) в вызовы функций ANSI и передавать их драйверу CLI. Менеджер драйверов ODBC 3.5 будет также отображать SQL_C_WCHAR в SQL_C_CHAR.
- В настоящее время поддержка SQL_C_WCHAR обеспечивается преобразованием данных из UCS2 в кодовую страницу прикладной программы и наоборот.
- Отсутствует поддержка SQL_WCHAR, SQL_WVARCHAR или SQL_WLONGVARCHAR.
- WCHARTYPE NOCONVERT не поддерживается для функций Unicode или для SQL_C_CHAR.

Связывание утилит базы данных при использовании клиента времени выполнения

Клиент времени выполнения не может использоваться для связывания утилит базы данных (import, export, georg, процессор командной строки) или файлов связывания CLI DB2 с базами данных, чтобы они могли использоваться с этими базами данных. Вместо него нужно использовать клиент администратора DB2 или клиент разработки прикладных программ DB2.

Прежде чем утилиты базы данных и файлы связывания CLI DB2 можно будет использовать с базой данных, они должны быть связаны с этой базой данных. Если в сетевой среде используется несколько клиентов, работающих в разных операционных системах или имеющих разные уровни версий или служб DB2, необходимо связать утилиты один раз для каждой комбинации операционной системы и версии DB2.

Дополнение к разделу "Using Compound SQL"

В этой книге пропущен следующий комментарий:

Все операторы SQL, которые могут быть подготовлены динамически и не являются запросами, могут выполняться.
Примечание: Внутри составного оператора Atomic Compound SQL не разрешены также операторы SQL savepoint, release savepoint и rollback to savepoint.
И наоборот, оператор Atomic Compound SQL нельзя использовать в точке сохранения.

Глава 3. Using Advanced Features

В раздел "Scrollable Cursors" (Указатели с возможностью прокрутки) надо добавить следующую информацию:

Поддержка указателей с возможностью прокрутки на стороне сервера для OS/390

Клиент UDB для платформ Unix, Windows и OS/2 при применении к базе данных OS/390 Версии 7 поддерживает указатели с возможностью изменения и прокрутки на стороне сервера. Для обращения к указателю OS/390 с возможностью прокрутки в трехуровневой среде на клиенте и на шлюзе должны работать с DB2 UDB Версии 7.1, FixPak 3 или новее.

Существует два программных интерфейса программ, обеспечивающих доступ к указателям с возможностью прокрутки: ODBC и JDBC. Интерфейс JDBC позволяет обращаться только к статическим указателям с прокруткой; интерфейс ODBC позволяет обращаться как к статическим, так и к управляемым клавиатурой указателям с прокруткой на стороне сервера.

Атрибуты указателя

В следующей таблице приведены значения по умолчанию атрибутов указателей OS/390 Версии 7 в ODBC.

Таблица 10. Атрибуты по умолчанию для указателей OS/390 в ODBC

Тип указателя	Чувствительность указателя	Возможность изменения указателя	Одновременность указателя	Возможность прокрутки указателя
только-вперед ^а	не задается	без возможности изменения	одновременность только для чтения	без возможности прокрутки
статический	нечувствительный	без возможности изменения	одновременность только для чтения	с возможностью прокрутки
управляемый	чувствительный	с возможностью изменения	одновременность значений	с возможностью прокрутки

^а Только-вперед - это поведение по умолчанию указателя с прокруткой без условия FOR UPDATE. Если для указателя только-вперед задать FOR UPDATE, будет создан указатель с возможностью изменения, одновременностью блокировок, без возможности прокрутки.

Поддерживаемые направления выборки

Все направления выборки ODBC поддерживаются с помощью интерфейсов SQLFetchScroll или SQLExtendedFetch.

Изменение управляемого указателя

Управляемый указатель - это указатель с возможностью изменения. Драйвер CLI добавляет условие FOR UPDATE к запросу, кроме случаев, когда запрос введен как запрос SELECT ... FOR READ ONLY или условие FOR UPDATE уже есть. Реализованный в DB2 для OS/390 управляемый указатель - это указатель одновременности значений. Указатель одновременности значений выражается в оптимистичной блокировке, то есть блокировка не производится, пока не будет предпринята попытка произвести изменение или удаление. При попытке произвести изменение или удаление сервер базы данных сравнивает предыдущие значения, извлеченные программой, с текущими значениями в основной таблице. Если эти значения совпадают - изменение или удаление успешно выполняется. Если значения не совпадают - операция

завершается неудачно. Если происходит ошибка, программа должна опять запросить значения и, если нужно, повторить операцию изменения или удаления.

Программа может изменять управляемый указатель двумя способами:

- Выполнить оператор UPDATE WHERE CURRENT OF "<имя указателя>" или DELETE WHERE CURRENT OF "<имя указателя>", используя SQLPrepare() с SQLExecute() или SQLExecDirect().
- Использовать SQLSetPos() или SQLBulkOperations(), чтобы изменить, удалить или добавить строку набора результатов.

Примечание: Строки, добавленные в набор результатов с помощью SQLSetPos() или SQLBulkOperations(), вставляются в таблицу на сервере, но не добавляются в набор результатов сервера. Поэтому эти строки не могут изменяться и нечувствительны к изменениям, произведенным другими транзакциями. Однако вставленные строки будут частью набора результатов, так как они кэшируются на клиенте. Любые триггеры, применяемые к вставленным строкам, для программы будут выглядеть так, как будто они не применялись. Чтобы вставляемые строки были изменяемыми, чувствительными, и был виден результат соответствующих триггеров, программа должна повторить запрос, чтобы повторно сгенерировать набор результатов.

Устранение неисправностей в программах, созданных до появления поддержки указателей с возможностью прокрутки

Так как поддержка указателей с возможностью прокрутки появилась недавно, у некоторых прикладных программ ODBC, работавших с предыдущими версиями UDB для OS/390 или UDB для Unix, Windows, и OS/2 может измениться поведение или производительность. Это происходит из-за того, что до появления поддержки указателей с прокруткой, программы, требовавшие такой указатель, получали указатель только-вперед. Чтобы восстановить поведение программы до появления поддержки указателей с прокруткой, задайте следующие ключевые слова конфигурации в файле db2cli.ini:

Таблица 11. Значения ключевых слов конфигурации, восстанавливающие поведение программ до появления поддержки указателей с возможностью прокрутки

Задание ключевого слова конфигурации	Описание
PATCH2=6	Возвращает сообщение, что указатели с возможностью прокрутки (как управляемые, так и статические) не поддерживаются. CLI автоматически заменяет требование указателя с прокруткой на требование указателя только-вперед.
DisableKeysetCursor=1	Запрещает управляемые указатели с прокруткой с обеих сторон - сервера и клиента. Можно использовать, чтобы драйвер CLI предоставлял программе статический указатель, когда затребован управляемый указатель.
UseServerKeysetCursor=0	Запрещает управляемые указатели на стороне сервера для программ, использующих библиотеку управляемых указателей клиентской стороны, чтобы симулировать управляемый указатель. Используйте эту возможность, только если возникли проблемы с управляемым указателем на стороне сервера, так как указатель на стороне клиента приводит к значительным издержкам и обычно имеет худшую производительность, чем указатель на стороне сервера.

Глава 4. Configuring CLI/ODBC and Running Sample Applications

Ключевые слова конфигурации

Пропустите последний параграф в описании ключевого слова CURRENTFUNCTIONPATH. Правильная информация:

Это ключевое слово применяется при разрешении неспецифицированных обращений к функциям и хранимым процедурам, для которых могло быть задано имя схемы, отличное от схемы текущего пользователя. Порядок имен схем задает порядок, в котором будут разрешаться имена функций и процедур. Подробности о разрешении функций и процедур смотрите в справочнике SQL Reference.

Написание хранимой процедуры в CLI

Недокументированное ограничение на хранимые процедуры CLI:

Выполняя вызовы нескольких хранимых процедур CLI, прикладная программа должна закрыть открытые указатели, полученные от одной хранимой процедуры, перед вызовом другой хранимой процедуры. Это значит, что первый набор открытых указателей должен быть закрыт прежде, чем следующая хранимая процедура попытается открыть указатель.

Хранимые процедуры и автоматическое связывание CLI

Драйвер CLI/ODBC в норме будет автоматически связывать пакеты CLI при первом выполнении прикладной программой CLI/ODBC SQL для базы данных при условии, что у пользователя есть надлежащие привилегия или авторизация. Автоматическое связывание пакетов CLI не может быть выполнено из хранимой процедуры и поэтому не произойдет, если самым первым действием прикладной программы будет вызов хранимой процедуры CLI. Перед запуском прикладной программы CLI, вызывающей хранимую процедуру CLI для новой базы данных DB2, необходимо один раз связать пакеты CLI при помощи следующей команды:

UNIX

```
db2 bind <BNDPATH>/@db2cli.lst blocking all
```

Windows и OS/2

```
db2bind "%DB2PATH%\bnd\@db2cli.lst" blocking
```

Во избежание автоматического связывания во время выполнения рекомендуется всегда связывать эти пакеты во время создания базы данных. Автоматическое связывание может закончиться неудачно, если у пользователя нет привилегии или если в это же самое время пытается автоматически связаться другая программа.

Дополнение к приложению D "Extended Scalar Functions": функции DAYOFWEEK_ISO() и WEEK_ISO()

Следующие функции пропущены в разделе Date and Time Functions приложения D "Extended Scalar Functions":

DAYOFWEEK_ISO(*date_exp*)

Возвращает день недели в *date_exp* как целое значение в диапазоне от 1 до 7, где 1 соответствует понедельнику. Обратите внимание на разницу между этой функцией и функцией DAYOFWEEK(), где 1 соответствует воскресенью.

WEEK_ISO(*date_exp*)

Возвращает неделю года в *date_exp* как целое значение в диапазоне от 1 до

53. Неделя 1 определена как первая неделя года, содержащая четверг. Таким образом, Week1 эквивалентно неделе, содержащей 4 января, поскольку первым днем недели считается понедельник.

Обратите внимание на отличие WEEK_ISO() от текущего определения WEEK(), которая возвращает значения до 54. Для функции WEEK() первая неделя - неделя содержащая первую субботу. Это эквивалентно неделе, содержащей 1 января, даже если эта неделя состоит всего из одного дня.

DAYOFWEEK_ISO() и WEEK_ISO() автоматически доступны в любой базе данных, созданной в версии 7.1. Если база данных создана до версии 7.1, эти функции могут быть недоступны. Чтобы сделать функции DAYOFWEEK_ISO() и WEEK_ISO() доступными в такой базе данных, используйте системную команду **db2updb**. Дополнительную информацию об **db2updb** смотрите в разделе "Command Reference" данных Замечаний по выпуску.

Приложение K. Using the DB2 CLI/ODBC/JDBC Trace Facility

Некоторые разделы этого приложения были обновлены. Самую свежую информацию об этом средстве трассировки смотрите в главе "Traces" руководства *Troubleshooting Guide*.

Использование статического SQL в прикладных программах CLI

Дополнительную информацию об использовании статического SQL в прикладных программах CLI смотрите на странице Web по адресу:
<http://www.ibm.com/software/data/db2/udb/staticcli/>

Ограничения статического профилирования JDBC/ODBC/CLI

Статическое профилирование JDBC/ODBC/CLI в настоящее время предназначено для простых прикладных программ. Оно не предназначено для сложных прикладных программ, состоящих из многих функциональных компонентов и реализующих сложные логические алгоритмы.

Чтобы оператор SQL был захвачен в сеанс профилирования, он должен быть успешно выполнен. Динамические операторы, для которых в сеансе поиска совпадающих операторов не будут найдены совпадающие операторы, будут продолжать выполняться как динамические вызовы JDBC/ODBC/CLI.

Чтобы оператор SQL был приемлемым кандидатом для совпадения операторов, он должен посимвольно совпадать с захваченным и связанным оператором. Пробелы существенны: например, "COL = 1" не считается совпадающим с "COL=1". Использование маркеров параметров вместо литералов увеличивает вероятность обнаружения совпадения.

При выполнении прикладной программы с предварительно связанными статическими операторами SQL динамические регистры, управляющие поведением динамических операторов, не будут влиять на выполнение операторов, преобразованных в статический вид.

Если прикладная программа выдает операторы DDL для объектов, используемых в последующих операторах DML, вы обнаружите все эти операторы в файле захвата. Средство связывания статического профилирования JDBC/ODBC/CLI попытается выполнить для них связывание. Эта попытка связывания будет успешной при работе с СУБД, поддерживающими опцию связывания VALIDATE(RUN), и не будет

успешной с СУБД, не поддерживающими эту опцию. В этом случае для прикладной программы не следует использовать статическое профилирование.

Администратор базы данных может отредактировать файл захвата, чтобы добавить, изменить или удалить операторы SQL в соответствии с требованиями конкретной прикладной программы.

Исправление описания параметра функции CLI SQLBindFileToParam()

Последний параметр функции CLI SQLBindFileToParam() - IndicatorValue - описан как "output (deferred)". На самом деле он "input (deferred)".

SQLNextResult - Связать следующий набор результатов с другим хэндлом оператора

К Главе 5 "DB2 CLI Functions" надо добавить следующий текст:

Назначение

Спецификация: DB2 CLI 7.x

Синтаксис

```
SQLRETURN SQLNextResult (SQLHSTMT StatementHandle1  
SQLHSTMT StatementHandle2);
```

Аргументы функции

Таблица 12. Аргументы SQLNextResult

Тип данных	Аргумент	Использование	Описание
SQLHSTMT	<i>StatementHandle</i>	входной	Хэндл оператора.
SQLHSTMT	<i>StatementHandle</i>	входной	Хэндл оператора.

Использование

Хранимая процедура возвращает несколько наборов результатов, оставляя после выхода один или несколько указателей открытыми. Доступ к первому набору результатов всегда осуществляется путем использования хэндла оператора, вызвавшего хранимую процедуру. Если возвращается несколько наборов результатов, для описания и считывания набора результатов можно использовать либо SQLMoreResults(), либо SQLNextResult().

SQLMoreResults() используется для закрытия указателя для первого набора результатов и позволяет обработку следующего набора результатов, а SQLNextResult() перемещает следующий набор результатов в *StatementHandle2* без закрытия указателя при *StatementHandle1*. Обе функции возвращают SQL_NO_DATA_FOUND, если нет считываемых наборов результатов.

Использование SQLNextResult() позволяет обрабатывать наборы результатов в любом порядке после того, как они были перенесены на другие хэндлы операторов. Смешанные вызовы SQLMoreResults() и SQLNextResult() разрешены до тех пор, пока не исчерпаются указатели (открытые наборы результатов) на *StatementHandle1*.

Когда `SQLNextResult()` возвращает `SQL_SUCCESS`, следующий набор результатов больше не связан с `StatementHandle1`. Вместо этого следующий набор результатов связывается с `StatementHandle2`, как если бы `SQLExecDirect()` только что успешно обработала запрос к `StatementHandle2`. Поэтому указатель может быть описан с использованием `SQLNumResultSets()`, `SQLDescribeCol()` или `SQLColAttribute()`.

После того, как была вызвана `SQLNextResult()`, набор результатов, теперь связанный с `StatementHandle2`, удаляется из цепочки оставшихся результатов и не может быть снова использован ни в `SQLNextResult()`, ни в `SQLMoreResults()`. Это означает, что для 'n' наборов результатов `SQLNextResult()` может быть вызвана успешно максимум 'n-1' раз.

Если `SQLFreeStmt()` вызывается с опцией `SQL_CLOSE` или `SQLFreeHandle()` вызывается с установленным для `HandleType` `SQL_HANDLE_STMT`, все находящиеся в состоянии ожидания на этом хэндле наборы результатов отбрасываются.

`SQLNextResult()` возвращает `SQL_ERROR`, если у `StatementHandle2` есть открытый указатель или если `StatementHandle1` и `StatementHandle2` не находятся на одном и том же соединении. Если возвращаются ошибки или предупреждения, `SQLERROR()` должна всегда вызываться с `StatementHandle1`.

Примечание: `SQLMoreResults()` работает также с параметризованным запросом с массивом значений входных параметров, заданным при помощи `SQLParamOptions()` и `SQLBindParameter()`. Однако `SQLNextResult()` не поддерживает такой способ.

Коды возврата

- `SQL_SUCCESS`
- `SQL_SUCCESS_WITH_INFO`
- `SQL_STILL_EXECUTING`
- `SQL_ERROR`
- `SQL_INVALID_HANDLE`
- `SQL_NO_DATA_FOUND`

Диагностика

Таблица 13. `SQLSTATE` для `SQLNextResult`

<code>SQLSTATE</code>	Описание	Объяснение
40003 08S01	Ошибка связи.	Ошибка связи между прикладной программой и источником данных до выполнения функции.
58004	Непредвиденная системная ошибка.	Неисправимая системная ошибка.
HY001	Ошибка выделения памяти.	DB2 CLI не в состоянии выделить память, необходимую для выполнения или завершения функции.
HY010	Ошибочная последовательность вызова функций.	Функция была вызвана во время операции обработки данных (<code>SQLParamData()</code> , <code>SQLPutData()</code>). У <code>StatementHandle2</code> есть открытый связанный с ним указатель. Функция была вызвана из операции SQL, ограниченной <code>BEGIN COMPOUND</code> и <code>END COMPOUND</code> .
HY013	Непредвиденная ошибка при работе с памятью.	CLI DB2 не сумел получить доступ к памяти, необходимой для выполнения или завершения функции.

Таблица 13. SQLSTATE для SQLNextResult (продолжение)

SQLSTATE	Описание	Объяснение
HYT00	Истек срок ожидания.	Истек срок ожидания возврата набора результатов источником данных. Истечение времени поддерживается только в системах без многозадачности, таких как Windows 3.1 и Macintosh System 7. Срок ожидания задается при помощи атрибута SQL_ATTR_QUERY_TIMEOUT для SQLSetConnectAttr().

Ограничения

Для параметризованных запросов можно использовать только SQLMoreResults().

Ссылки

- "SQLMoreResults - Determine If There Are More Result Sets" на странице 535
- "Returning Result Sets from Stored Procedures" на странице 120

Преобразования ADT

Следующая информация замещает существующую информацию в этой книге.

- Существует новый тип дескриптора (smallint) SQL_DESC_USER_DEFINED_TYPE_CODE со значениями:
 - SQL_TYPE_BASE 0 (это не USER_DEFINED_TYPE)
 - SQL_TYPE_DISTINCT 1
 - SQL_TYPE_STRUCTURED 2
 Это значение можно запросить с помощью SQLColAttribute или SQLGetDescField (только для IRD).

Чтобы получить действительные имена типов, добавлены атрибуты:

 - SQL_DESC_REFERENCE_TYPE
 - SQL_DESC_STRUCTURED_TYPE
 - SQL_DESC_USER_TYPE
 Указанные выше значения можно запросить с помощью SQLColAttribute или SQLGetDescField (только для IRD).
- Добавьте тип SQL_DESC_BASE_TYPE, если он требуется в прикладной программе. Например, прикладная программа может не распознавать структурированный тип и использоваться только для его выборки или вставки, оставляя работу с подробностями их структуры другим программным модулям.
- Добавьте новый атрибут соединения с именем SQL_ATTR_TRANSFORM_GROUP, чтобы позволить прикладной программе задавать группу преобразования (вместо того, чтобы использовать оператор SQL "SET CURRENT DEFAULT TRANSFORM GROUP").
- Добавьте новый атрибут оператора/соединения с именем SQL_ATTR_RETURN_USER_DEFINED_TYPES, который можно задать или запросить с помощью SQLSetConnectAttr; если задан этот атрибут, CLI возвращает в качестве допустимого типа SQL значение SQL_DESC_USER_DEFINED_TYPE_CODE. Этот атрибут требуется перед использованием любых преобразований.
 - По умолчанию этот атрибут выключен и в качестве типа SQL возвращается информация о базовом типе.
 - Когда он включен, в качестве SQL_TYPE возвращается SQL_DESC_USER_DEFINED_TYPE_CODE. Прикладная программа будет проверять SQL_DESC_USER_DEFINED_TYPE_CODE и затем получать соответствующее имя типа. Это будет возможно для SQLColAttribute, SQLDescribeCol и SQLGetDescField.

- Функция `SQLBindParameter` не вызывает ошибки при связывании `SQL_C_DEFAULT`, поскольку нет кода, позволяющего `SQLBindParameter` задавать тип `SQL_USER_DEFINED_TYPE`. Будут использоваться стандартные типы языка C по умолчанию в соответствии с типом SQL, переданным на сервер. Например:

```
sqlrc = SQLBindParameter (hstmt, 2, SQL_PARAM_INPUT, SQL_C_CHAR, SQL_VARCHAR, 30,  
                          0, &c2, 30, NULL);
```

Глава 28. Справочник по сообщениям

CLI0645E (новое сообщение)

CLI0645E Ошибка регистрации драйвера “<имя-класса>”. Сообщение: “<сообщение>”. SQLSTATE: “<sqlstate>”. SQLCODE: “<sqlcode>”.

Объяснение: Менеджер драйверов не может зарегистрировать этот драйвер JDBC DB2.

Действия пользователя: Исправьте ошибку, указанную возвращенным сообщением, SQLSTATE и SQLCODE, и запустите программу снова.

CLI0646E (новое сообщение)

CLI0646E Не удалось найти библиотеку “<имя-библиотеки>”.

Объяснение: “<имя-библиотеки>” не найдено в пути библиотек. Путь библиотек определяется в зависимости от операционной системы следующими переменными среды:

AIX (Java 1.1)
LD_LIBRARY_PATH

AIX (Java 1.2 или новее)
LIBPATH

HP-UX SHLIB_PATH

Linux LD_LIBRARY_PATH

OS/2 LIBPATH

Silicon Graphics IRIX
LD_LIBRARY_PATH

Операционная среда Solaris
LD_LIBRARY_PATH

32-битные операционные системы Windows
PATH

Действия пользователя: Проверьте правильность конфигурации среды прикладной программы для использования DB2. На платформах UNIX убедитесь, что сценарий sqllib/db2profile запущен для задания ваших переменных среды.

CLI0647E (новое сообщение)

CLI0647E Ошибка при размещении хэндла среды DB2. Код возврата = “<код-возврата>”.

Объяснение: Не удастся установить среду DB2 CLI.

Действия пользователя: Проверьте правильность конфигурации среды прикладной программы для

использования DB2. На платформах UNIX убедитесь, что сценарий sqllib/db2profile запущен для задания ваших переменных среды. Интерпретацию кодов возврата смотрите в справочном руководстве *CLI Guide and Reference* в разделе SQLAllocHandle().

DB21086I (новое сообщение)

DB21086I Операция инкрементного восстановления этого образа резервной копии завершена успешно, но для завершения операции в целом надо восстановить дополнительные образы резервных копий.

Объяснение: Чтобы завершить операцию инкрементного восстановления, надо восстановить все

образы инкрементных резервных копий в цепочке восстановления. Текущая операция завершилась успешно, но для завершения операции в целом надо восстановить дополнительные образы резервных копий.

Действия пользователя: Восстановите следующий образ резервной копии.

DB210060E (новое сообщение)

DB210060E Ошибка при попытке запустить службу “<имя-службы>” (причина = “<код-причины>”).

Объяснение: Не удалось запустить службу. Коды причин:

- 1 Служба не существует.
- 3 Служба уже запущена.
- 5 У текущего пользователя нет полномочий запускать или останавливать эту службу.
- 6 Ошибка при запуске службы.

Действия пользователя:

- 1 Проверьте, существует ли служба. Если эта

служба не существует, возможно, она была удалена вручную, или же установка DB2 Query Patroller не завершена успешно. В любом случае DB2 Query Patroller надо переустановить.

- 3 Служба уже запущена. Чтобы перезапустить ее, выполните команду DQPSTOP и повторите команду DQPSTART.

- 5 Задайте правильную информацию регистрации для запуска службы и повторите команду.

- 6 Повторите команду. Если ошибка повторится, обратитесь к администратору системы или посмотрите дополнительную диагностическую информацию в файлах 'syserr.log' и 'db2diag.log'.

DB210061E (новое сообщение)

DB210061E Ошибка при попытке остановить службу “<имя-службы>” (причина = “<код-причины>”).

Объяснение:

- 1 Служба не существует.
- 3 Служба не была запущена или же уже установлена.
- 5 У текущего пользователя нет полномочий запускать или останавливать эту службу.
- 6 Ошибка при остановке службы.

Действия пользователя:

- 1 Проверьте, существует ли служба. Если эта

служба не существует, возможно, она была удалена вручную, или же установка DB2 Query Patroller не завершена успешно. В любом случае DB2 Query Patroller надо переустановить.

- 3 Служба уже остановлена. Чтобы перезапустить ее, выполните команду DQPSTART.

- 5 Задайте правильную информацию регистрации для запуска службы и повторите команду.

- 6 Повторите команду. Если ошибка повторится, обратитесь к администратору системы или посмотрите дополнительную диагностическую информацию в файлах 'syserr.log' и 'db2diag.log'.

DB210062E (новое сообщение)

DB210062E Системная ошибка общих программ инициализации (SQLCODE = “<sqlcode>”).

Объяснение: Не удалось инициализировать среду. SQLCODE = “<sqlcode>”.

Действия пользователя: Проверьте SQLCODE и повторите команду. Если ошибка повторится, обратитесь к администратору системы или посмотрите дополнительную диагностическую информацию в файлах 'syserr.log' и 'db2diag.log'.

DB210113E (новое сообщение)

DB210113E Ошибка обработки запуска или остановки узла. Код причины “<код-причины>”.

Объяснение: DB2 Query Patroller не удалось запустить на заданном узле. Коды причин:

- 1 Узел сервера уже есть в файле dqpnodes.cfg. Допустим только один узел сервера.
- 2 Узла сервера нет в файле dqpnodes.cfg.
- 3 В файле dqpnodes.cfg есть запись с тем же номером узла.
- 4 Номера узла нет в файле dqpnodes.cfg.

- 5 В файле dqpnodes.cfg есть запись с тем же именем хоста.

- 6 Недопустимый тип узла.

- 7 Не удалось изменить запись узла сервера для агента. Узел сервера должен уже быть в файле dqpnodes.cfg.

- 8 Не удалось удалить запись узла сервера. Узел сервера должен уже быть в файле dqpnodes.cfg.

- 9 В файле dqpnodes.cfg обнаружено несколько записей узла сервера. Допустим только один узел сервера.

Действия пользователя:

- 1 Повторите команду без попытки запуска второго узла сервера.
- 2 Добавьте узел сервера в файл dqpnodes.cfg.
- 3 Проверьте номер узла и повторите команду с другим номером узла.
- 4 Проверьте номер узла и повторите команду с существующим номером узла.
- 5 Проверьте имя хоста и повторите команду с другим именем хоста.
- 6 Повторите данную команду, используя тип узла "server" или "agent".

- 7 Повторите команду без попытки замены узла сервера для агента.
- 8 Повторите команду без попытки отбрасывания узла сервера.
- 9 Определите, какая из записей правильна, и удалите остальные записи узла сервера из файла dqpnodes.cfg.

Примечание: Можно перезапустить узел сервера с другим именем хоста, но нельзя отбросить или заменить единственную запись узла сервера.

DB210114E (новое сообщение)

DB210114E Имя компьютера, заданное переменной реестра DB2 DQP_SERVER, не соответствует имени узла сервера в файле dqpnodes.cfg.

Объяснение: Переменная реестра DB2 DQP_SERVER имеет значение[хост]:[порт], где [хост] - IP-адрес

компьютера, где установлен сервер DB2 Query Patroller. Этот IP-адрес не соответствует IP-адресу узла сервера в файле dqpnodes.cfg.

Действия пользователя: Обеспечьте соответствие IP-адреса записи узла сервера в файле dqpnodes.cfg IP-адресу в переменной реестра DB2 DQP_SERVER.

DB210115E (новое сообщение)

DB210115E Текущий пользователь не имеет полномочий для выполнения команды "<имя-команды>".

Объяснение: У текущего пользователя нет необходимых полномочий выполнять эту команду. Если это команда

DQPSTART, DQPSTOP или IWM_TRACKER, их может выполнять только пользователь "iwm".

Действия пользователя: Если это команда DQPSTART, DQPSTOP или IWM_TRACKER, зарегистрируйтесь как пользователь "iwm" и повторите команду.

DB210116E (новое сообщение)

DB210116E DB2 Query Patroller уже запущен на этом узле.

Объяснение: DB2 Query Patroller уже работал в момент выполнения команды DQPSTART.

Действия пользователя: Чтобы перезапустить DB2 Query Patroller, выполните команду DQPSTOP, а затем команду DQPSTART.

DB210117E (новое сообщение)

DB210117E DB2 Query Patroller уже остановлен на этом узле.

Объяснение: Сервер DB2 Query Patroller не работал в момент выполнения команды DQPSTOP.

DB210118E (новое сообщение)

DB210118E Файл "<имя-файла>" нельзя удалить (причина = "<код-причины>").

Объяснение: Ошибка при попытке удалить файл "<имя-файла>".

Действия пользователя: Убедитесь в том, что имя файла "<имя-файла>" задано правильно, что файл существует и к нему разрешен доступ.

DB210120E (новое сообщение)

DB210120E Не удалось соединиться с базой данных “<имя-базы-данных>” с ID пользователя “<ID-пользователя>” из-за устранимой ошибки с SQLCODE “<sqlcode>”. Попытка соединения будет повторена через “<интервал-повторения>” минут.

Объяснение: Сервер DB2 Query Patroller не смог соединиться с базой данных из-за устранимой ошибки с SQLCODE “<sqlcode>”.

Действия пользователя: Проверьте SQLCODE и устраните причину ошибки. Сервер DB2 Query Patroller попытается соединиться с этой базой данных снова через “<интервал-повторения>” минут.

DB210200I (новое сообщение)

DB210200I До того, как сделанные изменения вступят в силу, все программы должны быть отсоединены от этой базы данных.

Объяснение: Команда ADD или DROP DATALINKS MANAGER выполнена успешно. Однако изменения не вступают в силу, пока все программы не отсоединятся от

этой базы данных. После отсоединения всех программ от базы данных при первом соединении сделанные изменения вступают в силу.

Действия пользователя: Убедитесь, что все программы отсоединены от базы данных, и выполните оператор CONNECT.

DB210201I (новое сообщение)

DB210201I Надо запустить утилиту согласования для таблиц со ссылками на файлы менеджера связей данных DB2. Менеджер связей данных DB2 не выполняет операций прекращения связей для этих файлов.

Объяснение: Команда DROP DATALINKS MANAGER выполнена успешно. Перед отбрасыванием менеджера связей данных DB2 надо убедиться, что в таблицах не

осталось ссылок на файлы менеджера связей данных DB2. Если такие ссылки остались, их надо удалить из таблиц базы данных при помощи утилиты согласования. Обратите внимание на то, что сами файлы останутся в связанном состоянии в файловой системе.

Дополнительные подробности смотрите в замечаниях по использованию команды DROP DATALINKS MANAGER в книге *Command Reference*.

DBI1793W (новое сообщение)

DBI1793W Неизвестный ID пользователя “<ID-пользователя>”.

Объяснение: Заданный пользователь неизвестен или не найден в текущей системе.

Действия пользователя: Проверьте, есть ли этот пользователь в данной системе, и повторите попытку.

DBI1794E (новое сообщение)

DBI1794E Попытка поместить в одну группу слишком много значков.

Объяснение: Была сделана попытка поместить в требуемую папку больше значков, чем позволяет менеджер рабочего стола.

Действия пользователя: Удалите ненужные значки с вашего рабочего стола и повторите команду.

DBI1795E (новое сообщение)

DBI1795E Недостаточно свободной памяти для создания всех требуемых значков.

Объяснение: Утилита создания значков израсходовала при создании значков всю доступную память.

Действия пользователя: Закройте ненужные программы и повторите попытку.

DBI1796W (новое сообщение)

DBI1796W Попытка создания значков в каталоге без необходимых разрешений.

Объяснение: У этого ID пользователя нет разрешения создавать значки для требуемого пользователя.

Действия пользователя: Зарегистрируйтесь как пользователь с полномочиями root или с необходимыми разрешениями для создания файлов в каталоге пользователя и повторите команду.

DBI1797I (новое сообщение)

DBI1797I Значки созданы успешно.

DWC13603E (новое сообщение)

DWC13603E Утилита экспорта не смогла открыть файл журнала.

Объяснение: Центр хранилищ данных всегда пытается создать файл журнала для захвата всех подробностей процесса экспорта. Эта ошибка указывает на то, что Центр хранилищ данных не имеет доступа к файлу журнала или не может открыть его. Если Центр хранилищ данных не может создать файл журнала, процесс экспорта не может продолжаться. Некоторые типичные причины этого сообщения:

- Задано неправильное имя файла.
- Задан неправильный путь.
- У вас нет полномочий для записи в каталог журнала.

Действия пользователя: Убедитесь, что у вас есть полномочия записи в указанный каталог журнала и в системе достаточно памяти и места. Если ошибку не удастся исправить, обратитесь в службу программной поддержки IBM.

DWC13700E (новое сообщение)

DWC13700E Не найден объект Центра хранилищ данных “<тип-объекта1>” с именем “<имя-объекта1>”, необходимый для импорта объекта Центра хранилищ данных “<тип-объекта2>” с именем “<object2-name>”.

Объяснение: Это внутренняя ошибка происходит, когда утилита импорта не находит объект, который должен

существовать в Центре хранилищ данных. Если требуемый объект не был создан во время импорта, утилита импорта не может продолжать работу.

Действия пользователя: Проверьте, не поврежден ли файл XML, который вы импортируете. Для этого регенерируйте его из исходного источника и запустите утилиту импорта снова. Если это сообщение повторится, обратитесь в службу программной поддержки IBM или к поставщику этого файла.

DWC13701E (новое сообщение)

DWC13701E Невозможно импортировать объект Центра хранилищ данных “<тип-объекта1>” с именем “<имя-объекта1>”, поскольку не найден общий объект метамодели хранилища типа “<тип-объекта2>”.

Объяснение: Для создания объекта этого типа Центр хранилищ данных должен найти необходимый общий объект метамодели хранилища. Без такого общего объекта объект Центра хранилищ данных не будет

допустимым. Если вы получили это сообщение, вероятно, в импортируемом файле XML отсутствует необходимый общий объект метамодели хранилища.

Действия пользователя: Проверьте, не поврежден ли файл XML, который вы импортируете. Для этого регенерируйте его из исходного источника и запустите утилиту импорта снова. Если это сообщение повторится, обратитесь в службу программной поддержки IBM или к поставщику этого файла.

DWC13702E (новое сообщение)

DWC13702E Первичный ключ уже существует и изменять его нельзя. Процесс импорта не может быть продолжен.

Объяснение: У вашей управляющей базы данных хранилища есть первичный ключ, а данные, которые вы пытаетесь импортировать, содержат другой первичный ключ для той же таблицы. Чтобы процесс импорта завершился успешно, первичный ключ должен быть один, или же два первичных ключа должны совпадать.

Нельзя иметь два разных первичных ключа.

Действия пользователя: Чтобы устранить несоответствие первичных ключей, выполните одно из следующих действий:

- Измените первичный ключ управляющей базы данных хранилища, чтобы привести его в соответствии с первичным ключом данных, которые вы хотите импортировать.

- Удалите первичный ключ из данных, которые вы хотите импортировать, и используйте первичный ключ из управляющей базы данных хранилища.
- Измените первичный ключ данных, которые вы хотите импортировать, чтобы привести его в соответствии с первичным ключом управляющей базы данных хранилища.

DWC13703E (новое сообщение)

DWC13703E Внешний ключ уже существует и изменять его нельзя. Процесс импорта не может быть продолжен.

Объяснение: У вашей управляющей базы данных хранилища есть внешний ключ, а данные, которые вы пытаетесь импортировать, содержат другой внешний ключ для той же таблицы. Чтобы процесс импорта завершился успешно, внешний ключ должен быть один, или же два внешних ключа должны совпадать. Нельзя иметь два разных внешних ключа.

Действия пользователя: Чтобы устранить несоответствие внешних ключей, выполните одно из следующих действий:

- Измените внешний ключ управляющей базы данных хранилища, чтобы привести его в соответствии со внешним ключом данных, которые вы хотите импортировать.
- Удалите внешний ключ из данных, которые вы хотите импортировать, и используйте внешний ключ из управляющей базы данных хранилища.
- Измените внешний ключ данных, которые вы хотите импортировать, чтобы привести его в соответствии со внешним ключом управляющей базы данных хранилища.

DWC13705E (новое сообщение)

DWC13705E Утилита импорта не смогла создать временный файл XML в каталоге EXCHANGE. Исключительная ситуация = “<код-исключительной-ситуации>”.

Объяснение: Центр хранилищ данных должен иметь возможность создать копию файла XML в том же каталоге, что и файл CWM.DTD. Это сообщение говорит о том, что Хранилище данных не может создать этот

файл XML. Если Центр хранилищ данных не может создать этот файл, процесс импорта не может продолжаться.

Действия пользователя: Убедитесь, что у вас есть полномочия записи в указанный каталог EXCHANGE и в системе достаточно памяти и места. Если ошибка повторится, запишите код исключительной ситуации, указанный в сообщении, и обратитесь в службу программной поддержки IBM.

DWC13706E (новое сообщение)

DWC13706E Нельзя загрузить файл XML “<файл-XML>”. Исключительная ситуация = “<код-исключительной-ситуации>”

Объяснение: Это внутренняя ошибка происходит, когда Центр хранилищ данных не может прочесть файл XML в процессе импорта. Обычные причины этого - файл либо поврежден, либо не содержит данных XML. Если Центр

хранилищ данных не может прочесть файл XML, процесс импорта не может продолжаться.

Действия пользователя: Проверьте, не поврежден ли файл XML, который вы импортируете. Для этого регенерируйте его из исходного источника и запустите утилиту импорта снова. Если это сообщение повторится, обратитесь в службу программной поддержки IBM или к поставщику этого файла.

DWC13707E (новое сообщение)

DWC13707E Утилита импорта не смогла открыть файл журнала.

Объяснение: Центр хранилищ данных всегда пытается создать файл журнала для захвата всех подробностей процесса импорта. Эта ошибка указывает на то, что Центр хранилищ данных не имеет доступа к файлу журнала или не может открыть его. Если Центр хранилищ данных не может создать файл журнала, процесс импорта не может продолжаться.

Действия пользователя: Некоторые типичные причины этого сообщения:

- Задано неправильное имя файла.
- Задан неправильный путь.
- У вас нет полномочий для записи в каталог журнала.

Установите причину ошибки и внесите необходимые изменения или же обратитесь в службу программной поддержки IBM.

SQL0017N (новое сообщение)

SQL0017N В методе или функции SQL должен быть задан и выполнен оператор RETURN.

Объяснение: Метод или функция SQL либо не содержит оператора RETURN, либо не завершается выполнением оператора RETURN.

Действия пользователя: Обеспечьте, чтобы функция или метод выполняли оператор RETURN.

sqlcode: -17

sqlstate: 42632

SQL0056N (новое сообщение)

SQL0056N Объявление переменной SQLSTATE или SQLCODE находится во вложенном составном операторе.

Объяснение: Объявление переменной SQLSTATE или SQLCODE находится во вложенном составном операторе, а не во внешнем составном операторе подпрограммы SQL.

Оператор невозможно обработать.

Действия пользователя: Объявляйте переменные SQLSTATE и SQLCODE только во внешнем составном операторе подпрограммы SQL.

sqlcode: -56

sqlstate: 42630

SQL0057N (новое сообщение)

SQL0057N Оператор RETURN в методе или функции SQL должен содержать возвращаемое значение.

Объяснение: В методе или функции SQL задан оператор RETURN без возвращаемого значения.

Оператор невозможно обработать.

Действия пользователя: Задайте значение в операторе RETURN.

sqlcode: -57

sqlstate: 42631

SQL0058N (новое сообщение)

SQL0058N Тип данных значения оператора RETURN в процедуре SQL должен быть INTEGER.

Объяснение: В процедуре SQL задан оператор RETURN со значением или выражением, тип которого - не INTEGER.

Оператор невозможно обработать.

Действия пользователя: Задайте в операторе RETURN значение типа INTEGER.

sqlcode: -58

sqlstate: 428F2

SQL0097N (новое сообщение)

SQL0097N Переменные и параметры типов данных LONG VARCHAR и LONG VARGRAPHIC в подпрограммах SQL не допускаются.

Объяснение: Подпрограммы SQL (процедуры, функции и методы) не поддерживают переменные и параметры типов данных LONG VARCHAR и LONG VARGRAPHIC.

Действия пользователя: Не используйте переменные и параметры типов данных LONG VARCHAR и LONG VARGRAPHIC в подпрограммах SQL. Вместо LONG VARCHAR используйте VARCHAR с явно заданной длиной. Вместо LONG VARGRAPHIC используйте VARGRAPHIC с явно заданной длиной.

sqlcode: -97

sqlstate: 42601

SQL0224N (новое сообщение)

SQL0224N Таблица результатов не согласуется с базовой таблицей при использовании указателя “<имя-указателя>”.

Объяснение: Была предпринята попытка выполнения позиционированного оператора UPDATE или DELETE с

использованием указателя “<имя-указателя>”, определенного как SENSITIVE STATIC, на строке, где значения столбца в таблице результатов не совпадают с текущими значениями в строке базовой таблицы. Строки не совпадают потому, что строка базовой таблицы была изменена в период между выборкой в таблицу

результатов и обработкой позиционированного оператора UPDATE или DELETE.

Оператор невозможно обработать. Позиция указателя не меняется.

Действия пользователя: Измените уровни изоляции, чтобы строка базовой таблицы не менялась в ходе

операции указателя или измените программу, применив FETCH INSENSITIVE, и повторите обработку позиционированного оператора UPDATE или DELETE.

sqlcode: -224

sqlstate: 24512

SQL0225N (новое сообщение)

SQL0225N Оператор FETCH для указателя “<имя-указателя>” недопустим, поскольку этот указатель определен без условия SCROLL.

Объяснение: Оператор FETCH для указателя без прокрутки “<имя-указателя>” был задан с одним из следующих ключевых слов, предназначенных для указателя с прокруткой: PRIOR, FIRST, LAST, BEFORE, AFTER, CURRENT, ABSOLUTE или RELATIVE. Для указателей без прокрутки можно задавать только NEXT. Никакие данные не считываются.

Оператор невозможно обработать. Позиция указателя не меняется.

Действия пользователя: Измените оператор FETCH, удалив текущее ключевое слово ориентации выборки, например PRIOR или FIRST, и заменив его словом NEXT. Другой вариант - изменить определение указателя, сделав его указателем с прокруткой.

sqlcode: -225

sqlstate: 42872

SQL0227N (новое сообщение)

SQL0227N FETCH NEXT, PRIOR, CURRENT или RELATIVE недопустимы, поскольку позиция указателя “<имя-указателя>” неизвестна (“<sqlcode>”, “<sqlstate>”).

Объяснение: Позиция указателя “<имя-указателя>” неизвестна. Предыдущий оператор выборки нескольких строк FETCH для указателя “<имя-указателя>” привел к ошибке (SQLCODE “<sqlcode>”, SQLSTATE “<sqlstate>”) в ходе обработки нескольких полученных строк. Одну или несколько запрашиваемых строк не удается вернуть в программу после ошибки, позиция указателя осталась неизвестной.

Если бы в предыдущем операторе выборки нескольких строк FETCH была предусмотрена структура индикатора, был бы возвращен положительный

SQLCODE и все полученные строки можно было вернуть в программе.

Оператор невозможно обработать. Позиция указателя не меняется.

Действия пользователя: Закройте и вновь откройте указатель, чтобы сбросить позицию. Для указателей с прокруткой можно изменить оператор FETCH, задав одну из других ориентаций выборки, например, FIRST, LAST, BEFORE, AFTER или ABSOLUTE, чтобы установить допустимую позицию указателя и считать строку данных.

sqlcode: -227

sqlstate: 24513

SQL0228N (новое сообщение)

SQL0228N Условие FOR UPDATE задано для указателя только для чтения “<имя-указателя>”.

Объяснение: Указатель “<имя-указателя>” определен как INSENSITIVE SCROLL, но соответствующий оператор SELECT содержит условие FOR UPDATE.

Оператор невозможно обработать.

Действия пользователя: Чтобы определить указатель только для чтения, задайте INSENSITIVE в DECLARE CURSOR, но не задавайте условие FOR UPDATE как часть оператора SELECT данного указателя.

sqlcode: -228

sqlstate: 42620

SQL0231W (новое сообщение)

SQL0231W Текущее положение указателя “<имя-указателя>” недопустимо для выборки (FETCH) текущей строки.

Объяснение: Оператор FETCH CURRENT или FETCH RELATIVE 0 выполнялся для указателя с прокруткой “<имя-указателя>”. Такая операция недопустима, так как указатель не ставится на позицию строки таблицы результатов. Оператор FETCH текущей строки недопустим после оператора FETCH BEFORE или FETCH AFTER или после оператора FETCH, который привел к SQLCODE +100.

Оператор невозможно обработать. Позиция указателя не меняется.

Действия пользователя: Перед попыткой считывания текущей строки установите указатель на строку таблицы результатов.

sqlcode: +231

sqlstate: 02000

SQL0243N (новое сообщение)

SQL0243N Указатель с условием SENSITIVE “<имя-указателя>” нельзя определить для заданного оператора SELECT.

Объяснение: Указатель “<имя-указателя>” определен как SENSITIVE, но содержимое оператора SELECT требует построения временной таблицы результатов указателя, а менеджер баз данных не может гарантировать, что изменения, сделанные вне данного указателя, будут видимы. Такая ситуация имеет место, если содержимое запроса производит таблицу результатов только для чтения. Например, если запрос включает объединение, таблица результатов будет

таблицей только для чтения. В этом случае указатель надо определять как INSENSITIVE или ASENSITIVE.

Оператор невозможно обработать.

Действия пользователя: Либо измените содержимое запроса, чтобы таблица не была таблицей только для чтения, либо измените тип указателя на INSENSITIVE или ASENSITIVE.

sqlcode: -243

sqlstate: 36001

SQL0244N (новое сообщение)

SQL0244N Условие SENSITIVITY “<чувствительность>”, заданное в операторе FETCH, недопустимо для указателя “<имя-указателя>”.

Объяснение: Опция SENSITIVE “<чувствительность>”, заданная в операторе FETCH, конфликтует с опцией SENSITIVE, действующей для указателя “<имя-указателя>”. Ниже в списке показано, что может быть задано в операторе FETCH:

DECLARE CURSOR	оператор FETCH
INSENSITIVE	INSENSITIVE
SENSITIVE STATIC	SENSITIVE или INSENSITIVE
SENSITIVE DYNAMIC	SENSITIVE
SENSITIVE	SENSITIVE

ASENSITIVE	INSENSITIVE или SENSITIVE (в зависимости от действующей опции SENSITIVE указателя)
------------	--

Для указателя без прокрутки опцию SENSITIVE задавать нельзя.

Оператор невозможно обработать.

Действия пользователя: Измените или удалите опцию SENSITIVE, заданную в операторе FETCH.

sqlcode: -244

sqlstate: 428F4

SQL0270N (новый код причины 40)

Для сообщения SQL0270N добавлен следующий код причины:

Код причины 40

В разделе "Объяснение": Функцию IDENTITY_VAL_LOCAL нельзя использовать в триггере или функции S

В разделе "Действия пользователя": Удалите вызов функции IDENTITY_VAL_LOCAL из определения триггера

SQL0301N (новый текст объяснения)

Для этого сообщения был расширен раздел Объяснение. Теперь он выглядит так:

Объяснение: Переменную хоста невозможно использовать так, как это задано в этом операторе, так как ее тип данных не соответствует предполагаемому использованию. Эта ошибка может возникнуть или в результате задания неверной

переменной хоста, или в результате некорректного значения SQLTYPE в SQLDA оператора EXECUTE или OPEN.
Для пользовательского структурного типа соответствующий встроенный тип переменной хоста или SQLTYPE может не быть совместимым с параметром функции преобразования TO SQL, определенной в группе преобразования для оператора.
Оператор невозможно обработать.

SQL0303N (новый текст)

Для этого сообщения были расширены разделы Объяснение и Действия пользователя. Теперь они выглядят так:

Объяснение: Встроенный оператор SELECT или VALUES записывает результат выбора в переменную хоста, но тип данных этой переменной несовместим с типом данных элемента соответствующего списка SELECT или списка VALUES. Оба типа должны быть числовыми, символьными или графическими типами. В случае пользовательского типа данных возможно, что эта переменная хоста, определена со связанным типом данных, который не совместим с типом результата функции преобразования SQL FROM, определенной в группе преобразования для этого оператора.
Например, если столбец имеет тип данных дата или время, переменная должна быть символьного типа данных соответствующей минимальной длины.
Оператор невозможно обработать.

Действия пользователя: Убедитесь, что определения таблицы активны и что данная переменная хоста имеет правильный тип данных.
Для пользовательского типа данных убедитесь, что соответствующий встроенный тип данных совместим с типом результата функции преобразования FROM SQL, определенной в группе преобразования для этого оператора.

SQL0336N (новое сообщение)

SQL0336N Масштаб десятичного числа должен быть ноль.

Объяснение: Десятичное число используется в контексте, где масштаб должен быть равным нулю. Такая ситуация может встретиться, если десятичное число задается в операторе CREATE или ALTER SEQUENCE для START WITH, INCREMENT, MINVALUE, MAXVALUE или RESTART WITH.

Оператор невозможно обработать.

Действия пользователя: Измените десятичное число, удалив из его дробной части все ненулевые цифры.

sqlcode: -336

sqlstate: 428FA

SQL0348N (новое сообщение)

SQL0348N “<выражение-последовательности>” не может быть задано в этом контексте.

Объяснение: Оператор содержит выражение NEXTVAL или PREVVAL в недопустимом контексте. Выражение NEXTVAL нельзя задавать в следующих контекстах:

- условие join полного внешнего объединения
- значение DEFAULT для столбца в операторе CREATE TABLE или ALTER TABLE
- определение генерируемого столбца в операторе CREATE TABLE или ALTER TABLE
- условие проверочного ограничения
- оператор CREATE TRIGGER, CREATE VIEW, CREATE METHOD или CREATE FUNCTION

- оператор SELECT, который содержит операцию DISTINCT
- условие GROUP BY оператора SELECT
- условие join объединения
- оператор SELECT в сочетании с другим оператором SELECT, где используется операция над множествами UNION, INTERSECT или EXCEPT
- вложенное табличное выражение
- список параметров табличной функции
- условие WHERE оператора SELECT, DELETE или UPDATE
- условие ORDER BY
- список параметров оператора CALL

Выражения NEXTVAL и PREVVAL нельзя задавать в следующих контекстах:

- выражение CASE
- список параметров функции сводки
- подзапрос

Оператор невозможно выполнить.

Действия пользователя: Удалите ссылку на выражение последовательности и повторите обработку оператора.

sqlcode: -348

sqlstate: 428F9

SQL0349N (новое сообщение)

SQL0349N Спецификация выражения NEXTVAL для столбца в позиции “<позиция-столбца>” должна совпадать со спецификацией остальных выражений для этого столбца во всех строках.

Объяснение: Выражение, заданное для столбца в позиции “<позиция-столбца>” условия VALUES оператора INSERT вставки нескольких строк, или выражение VALUES содержит выражение NEXTVAL. Если выражение, содержащее выражение NEXTVAL, используется для задания значения столбца в одном из этих контекстов, это же выражение нужно задать для данного столбца во всех строках. Например, следующий оператор INSERT - правильный:

```
INSERT INTO T1
VALUES(NEXTVAL FOR sequence1 + 5, 'a'),
      (NEXTVAL FOR sequence1 + 5, 'b'),
      (NEXTVAL FOR sequence1 + 5, 'c')
```

А следующий оператор INSERT - неверный:

```
INSERT INTO T1
VALUES(NEXTVAL FOR sequence1 + 5, 'a'),
      (NEXTVAL FOR sequence1 + 5, 'b'),
      (NEXTVAL FOR sequence1 + 4, 'c')
```

Действия пользователя: Исправьте синтаксис и повторите обработку оператора.

sqlcode: -349

sqlstate: 560B7

SQL0357N (новый код причины 3)

SQL0357N Менеджер связей данных DB2 “<имя>” в данный момент недоступен. Код причины = “<код причины>”.

Объяснение:

03 Либо доступ к менеджеру связей данных DB2 в данный момент не разрешен, либо DB2 не может соединиться с менеджером связей данных DB2.

Действия пользователя:

03 Менеджер связей данных DB2 не будет доступен, пока DB2 не сможет убедиться, что он

находится в согласованном состоянии.

Подождите, пока DB2 сделает это асинхронно. Возможно, DB2 не сможет сделать это из-за ситуации, описанной другими кодами причин. Поэтому, если ошибка не будет устранена, посмотрите файл db2diag.log этого экземпляра, чтобы найти и устранить эти причины.

Дополнительную информацию о восстановлении после ошибки для менеджера связей данных DB2 смотрите в руководстве Administration Guide. Проверьте также параметры хоста для соединений с этим Менеджером связей данных DB2.

SQL0358N (новое действие пользователя 26)

Код причины 26

Объяснение: К файлу, на который ссылается значение DATALINK, нельзя получить доступ для связывания.

Это может быть каталог, символическое обозначение связи или файл с битом разрешения для установки ID пользователя (SUID) или группового ID (SGID) или же файл указан как никому не принадлежащий (UID = -2).

Действия пользователя: Связывание каталогов не разрешено. Используйте настоящее имя файла, а не символическую связь.

Если включен SUID или SGID, этот файл нельзя связать с использованием типа DATALINK.

Если у файла нет владельца (uid = -2), этот файл нельзя связать с использованием типа DATALINK с опцией READ PERMISSION DB.

SQL0368N (новое сообщение)

SQL0368N Менеджер связей данных DB2 “<имя-dlm>” не зарегистрирован для этой базы данных.

Объяснение: Менеджер связей данных DB2 “<имя-dlm>” не зарегистрирован для этой базы данных.

Зарегистрированные менеджеры связей данных DB2

игнорируются, если для параметра конфигурации менеджера баз данных DATALINKS задано значение NO. Возможно, данный менеджер связей данных DB2 был отброшен при помощи команды DROP DATALINKS MANAGER. Возможно, что в данный момент регистрируется новый вариант данного менеджера связей

данных DB2 с тем же именем; в этом случае ошибка связана с одним или несколькими вариантами этого менеджера связей данных DB2, которые были отброшены ранее.

Оператор невозможно обработать.

Действия пользователя: Убедитесь, что для параметра DATALINKS конфигурации менеджера баз данных задано значение YES. Значения DATALINK от ранее

отброшенных менеджеров связей данных DB2, следует удалить, воспользовавшись утилитой согласования. Дополнительные подробности смотрите в замечаниях по использованию команды DROP DATALINKS MANAGER в книге *Command Reference*.

sqlcode: -0368

sqlstate: 55022

SQL0408N (новый текст)

Для этого сообщения были расширены разделы Объяснение и Действия пользователя. Теперь они выглядят так:

Объяснение: Тип данных значения, назначаемого столбцу, параметру, переменной SQL или переходной переменной оператором SQL, несовместим с объявленным типом данных объекта назначения. Оба этих типа данных должны быть:

- Числовыми
- Символьными
- Графическими
- Типами "дата" или символьными
- Типами "время" или символьными
- Типами "отметка времени" или символьными
- Связями данных
- Одного особого типа
- Ссылочных типов, где тип назначения значения - подтип типа назначения столбца.
- Одинаковыми пользовательскими структурированными типами.

Или статический тип значения должен быть подтипом статического типа (объявленного) объекта назначения. Если использована переменная хоста, соответствующий встроенный тип переменной хоста должен быть совместим с параметром функции преобразования TO SQL, определенной в группе преобразования для оператора.

Оператор невозможно обработать.

Действия пользователя: Проверьте оператор и, возможно, таблицу назначения, чтобы определить тип данных назначения. Обеспечьте правильность типа данных переменной, выражения или литерального значения для объекта назначения. Для пользовательского структурированного типа проверьте также параметр функции преобразования TO SQL, определенной в группе преобразования для оператора как цель назначения.

SQL0423N (исправленный текст)

Переменная адреса "<положение-переменной>" в настоящий момент не содержит никакого адреса.

Объяснение: Ошибка в переменной адреса.

Или с ним не было связано значение типа большой объект (LOB), или локатор, связанный с этой переменной, освобожден, или указатель набора результатов закрыт.

Если указано значение "<положение-переменной>" порядковый номер ошибочной переменной в заданном наборе переменных. В зависимости от того, когда именно была обнаружена эта ошибка, менеджер баз данных может оказаться не в состоянии определить "<положение-переменной>". Вместо обычного положения "<положение-переменной>" локатора, возвращенное пользовательской функцией с указанным именем, ошибочно.

Действия пользователя: Если это была переменная адреса большого объекта, исправьте программу так, чтобы использованные в этом операторе SQL переменные адресов больших объектов имели к моменту выполнения этого оператора правильные значения адресов больших объектов. Если возвращается указатель, он должен быть открыт перед попыткой его выделить.

sqlcode: -423
sqlstate: 0F001

SQL0590N (новое сообщение)

SQL0590N Имя "<имя>", указанное в контексте "<тег-контекста>", не уникально.

Объяснение: В контексте, определенном "<тегом-контекста>" в качестве параметра, переменной SQL, указателя, метки или условия указано имя "<имя>". Это имя не уникально.

Если "<тег-контекста>" - это "BEGIN...END", это значит, что контекст ошибки - динамический составной оператор SQL. В ином случае контекст ошибки - триггер или подпрограмма, а "<тег-контекста>" - имя триггера или подпрограммы, содержащих составной оператор.

- Если "<имя>" - это имя параметра, оно должно быть уникальным в списке параметров и условия EXPRESSION AS подпрограммы.

- Если "<имя>" - имя переменной SQL, указателя или условия, оно должно быть уникальным в составном операторе.
- Метка должна быть уникальной в составном операторе и отличаться от меток операторов, в которые он вложен.

Действия пользователя: Измените это имя на уникальное.

sqlcode: -590

sqlstate: 42734

SQL0670N (исправленный текст)

В сообщении SQL0670N указывается максимальная длина строки для таблиц, определенных в операторе CREATE TABLE или ALTER TABLE, и в табличном пространстве, в котором созданы эти таблицы. Однако сообщение SQL0670N также применяется для длин строк объявленных временных таблиц, определенных в операторе DECLARE GLOBAL TEMPORARY TABLE, и в пользовательских временных табличных пространствах, в которых созданы эти объявленные временные таблицы. Если для оператора DECLARE GLOBAL TEMPORARY TABLE выдается сообщение об ошибке SQL0670N, это означает, что в операторе DECLARE TEMPORARY TABLE задана слишком большая длина строки для этого пользовательского временного табличного пространства.

Исправленный текст сообщения:

Длина строки в таблице превышает предельное значение "<длина>" байт. (Табличное пространство "<имя табличного пространства>".)

Объяснение: Длина строки таблицы для менеджера баз данных не может превышать:

- 4005 байт для табличного пространства с размером страницы 4 Кбайта.
- 8101 байт для табличного пространства с размером страницы 8 Кбайт.
- 16293 байт для табличного пространства с размером страницы 16 Кбайт.
- 32677 байт для табличного пространства с размером страницы 32 Кбайта.

Длина вычисляется путем суммирования внутренних длин столбцов. Подробную информацию о внутренних длинах столбцов можно найти в описании CREATE TABLE в справочнике SQL Reference.

Имеет место одна из с

- Длина строки таблицы, определенной в операторе CREATE TABLE или ALTER TABLE, превышает максимальную длину для размера страницы данного табличного пространства.

Имя обычного табличного пространства "<имя табличного пространства>" указывает табличное пространство, размер страницы которого использовался для определения предельной длины строки.

- Длина строки таблицы, определенной в операторе DECLARE GLOBAL TEMPORARY TABLE, превышает максимальную длину для размера страницы данного табличного пространства.

Имя пользовательского временного табличного пространства "<имя табличного пространства>" указывает табличное пространство, размер страницы которого использовался для определения предельной длины строки.

Оператор невозможно обработать.

Действия пользователя: В зависимости от причины выполните одно из следующих действий:

- Для оператора CREATE TABLE, ALTER TABLE или DECLARE GLOBAL TEMPORARY TABLE задайте, если возможно, табличное пространство с большим размером страницы.

- В прочих случаях уменьшите длину строки, удалив один или несколько столбцов или уменьшив длину одного или нескольких столбцов.
sqlcode: -670
sqlstate: 54010

SQL0845N (новое сообщение)

SQL0845N Выражение PREVVAL нельзя использовать до того, как выражение NEXTVAL в текущем сеансе сгенерирует значение для последовательности "<имя-последовательности>".

Объяснение: В выражении PREVVAL указана последовательность "<имя-последовательности>", но значение для этой последовательности еще не сгенерировано. Выражение NEXTVAL должно в данном

сеансе сгенерировать значение до того, как выражение PREVVAL будет его использовать.

Действия пользователя: Выполните в сеансе хотя бы одно выражение NEXTVAL для последовательности до выражений PREVVAL той же последовательности.

sqlcode: -845

sqlstate: 51035

SQL0846N (новое сообщение)

SQL0846N Неверная спецификация столбца идентификации или последовательности "<тип-объекта>" "<имя-объекта>".

Объяснение: Для столбца идентификации или объекта последовательности спецификация атрибута в операторе ALTER или CREATE TABLE может быть недопустима по одной из следующих причин:

- Исходный тип данных столбца идентификации или объекта последовательности не поддерживается. Для столбца идентификации или объекта последовательности допускаются следующие типы данных: SMALLINT, INTEGER, BIGINT и DECIMAL с масштабом, равным нулю.

- MINVALUE не должно быть больше MAXVALUE.
- Для CACHE задано недопустимое значение. Это должно быть число типа small integer с минимальным значением 2.

Действия пользователя: Исправьте синтаксис и повторите обработку оператора.

sqlcode: -846

sqlstate: 42815

SQL1179W (исправленный текст)

Для этого сообщения были расширены разделы Объяснение и Действия пользователя. Теперь они выглядят так:

SQL1179W Объект типа "<тип-объекта>" с именем "<имя-объекта>" может потребовать, чтобы вызывающий процесс имел необходимые привилегии для объектов источников данных.

Объект с именем "<имя-объекта>" ссылается на табличную функцию OLE DB или на псевдоним, где на источнике данных расположены сами данные.

При обращении к этому источнику данных на основе пользователя, запустившего операцию, проводятся отбор и агрегация данных.

Если "<тип-объекта>" - SUMMARY TABLE, операция - это обновление данных для таблицы сводки. От пользователя, вызвавшего оператор REFRESH TABLE или SET INTEGRITY для обновления, могут потребоваться необходимые привилегии для обращения к исходным данным на источнике данных.

Если "<тип-объекта>" - PACKAGE, PROCEDURE или VIEW, от любого пользователя пакета, процедуры или производной таблицы могут потребоваться необходимые привилегии для обращения к исходным данным на источнике данных.

В любом случае при попытке обращения к объекту источника данных может произойти ошибка авторизации.

Действия пользователя: Привилегий для производной таблицы, таблицы сводки пакета или процедуры может оказаться недостаточно для операции, которая обращается к данным на источнике данных. Может потребоваться пользовательский доступ к объектам источника данных.

sqlcode: +1179

sqlstate: 01639

SQL1186N (новое сообщение)

SQL1186N Объект “<тип-объекта>” с именем “<имя-объекта>” либо был изменен и стал объектом объединения, либо был создан как объект объединения без указания условия FEDERATED.

Объяснение: Если создана таблица сводки и полная выборка ссылается, прямо или косвенно, на табличную функцию OLE DB или на псевдоним, надо задавать условие FEDERATED.

Если производная таблица (не таблица объединения) изменена и полная выборка теперь ссылается, прямо или косвенно, на табличную функцию OLE DB или на псевдоним, надо задавать условие FEDERATED.

Если производная таблица объединения изменена, но

полная выборка еще ссылается, прямо или косвенно, на табличную функцию OLE DB или на псевдоним, нельзя задавать условие NOT FEDERATED.

Действия пользователя: Чтобы создать таблицы сводки или изменить производную таблицу так, чтобы она стала производной таблицей объединения, надо задать условие FEDERATED.

При изменении производной таблицы объединения, при котором она остается производной таблицей объединения, нельзя задавать условие NOT FEDERATED.

sqlcode: -1186

sqlstate: 429BA

SQL1550N (новый SQLCODE)

SQL1550N Команда SET WRITE SUSPEND завершилась неудачно. Код причины = “<код-причины>”.

Объяснение: Команду SET WRITE SUSPEND нельзя выполнить, пока не будет устранена причина, на которую указывает “<код-причины>”:

- 1 База данных не активирована.
- 2 Идет операция резервного копирования для базы данных назначения. Операции записи нельзя приостановить, пока DB2 не закончит резервное копирование.
- 3 Идет операция восстановления для базы данных назначения. Операции записи для этой базы данных нельзя приостановить, пока DB2 не закончит восстановление.
- 4 Операции записи для этой базы данных уже приостановлены.

Действия пользователя:

- 1 Активируйте эту базу данных при помощи команды ACTIVATE DATABASE, затем повторите команду SET WRITE SUSPEND.
- 2 Подождите завершения резервного копирования, затем повторите команду SET WRITE SUSPEND.
- 3 Подождите завершения восстановления, затем повторите команду SET WRITE SUSPEND.
- 4 Запись в эту базу данных уже приостановлена. Чтобы возобновить операции записи, введите команду SET WRITE RESUME.

sqlcode: -1550

SQL1551N (новый SQLCODE)

SQL1551N Команда SET WRITE RESUME не выполнена, поскольку эта база данных не находится сейчас в состоянии WRITE SUSPEND.

Объяснение: Эта база данных не находится сейчас в состоянии WRITE SUSPEND. Возобновить операции записи можно только для базы данных, для которой они были приостановлены.

Действия пользователя: Никаких действий не требуется, поскольку операции записи для этой базы данных разрешены. Чтобы приостановить операции записи для нее, введите команду SET WRITE SUSPEND.

sqlcode: -1551

SQL1552N (новый SQLCODE)

SQL1552N Команда не выполнена, поскольку база данных находится в состоянии WRITE SUSPEND.

Объяснение: Эта команда недопустима, если для базы данных приостановлены операции записи. База данных находится в состоянии WRITE SUSPEND.

Действия пользователя: Если невыполненная команда - RESTART DATABASE, повторите ее с опцией WRITE RESUME.

Если невыполненная команда - BACKUP или RESTORE, введите команду SET WRITE RESUME FOR DATABASE для возобновления операций записи в эту базу данных. Затем повторите команду BACKUP или RESTORE.

sqlcode: -1552

SQL1553N (новый SQLCODE)

SQL1553N DB2 не может быть остановлена, поскольку одна или несколько баз данных находятся в состоянии WRITE SUSPEND.

Объяснение: Нельзя закрыть базу данных, для которой приостановлены операции записи. База данных находится в состоянии WRITE SUSPEND.

Действия пользователя: Введите команду SET WRITE RESUME, чтобы возобновить операции записи для этой базы данных, затем повторите команду db2stop.

sqlcode: -1553

SQL1704N (новые коды причины)

Код причины 14

Объяснение: Таблица содержит неверный первичный ключ или ограничение уникальности.

Действия пользователя: У таблицы есть индекс, который ошибочно используется для первичного ключа или ограничения уникальности. Отбросьте первичный ключ или ограничение уникальности баз данных, который использовался до текущего выпуска.

Повторите команду перенастройки базы данных в текущий выпуск, а затем пересоздайте первичный ключ или ограничение уникальности.

Код причины 15

Объяснение: У таблицы нет индекса уникальности по столбцу REF IS.

Действия пользователя: Создайте индекс уникальности по столбцу REF IS типизированной таблицы в выпуске менеджера базы данных, который ранее использовался для работы с базой.

Затем повторите команду перенастройки базы данных в текущий выпуск.

Код причины 16

Объяснение: Таблица не зарегистрирована, однако у нее есть столбец DATALINK с управлением связями файлов.

Действия пользователя: Отбросьте таблицу и пересоздайте ее без свойства NOT LOGGED.

Это надо сделать в выпуске менеджера баз данных, который использовался до текущего выпуска.

Затем повторите команду перенастройки базы данных в текущий выпуск.

Код причины 17

Объяснение: Не удалось разместить новую страницу из табличного пространства системного каталога DMS.

Действия пользователя: Восстановите резервную копию в предыдущем выпуске менеджера баз данных.

Добавьте контейнеры в табличное пространство.

Рекомендуется выделить для перенастройки базы данных 70% свободного пространства.

Перейдите назад в текущий выпуск и перенастройте базу данных.

SQL2077W (новое сообщение)

SQL2077W Обработка согласования завершилась успешно на доступных менеджерах связей данных DB2. Обработка согласования отложена на недоступных менеджерах связей данных DB2. Смотрите подробную информацию в файле db2diag.log.

Объяснение: Некоторые или все менеджеры связей данных DB2, упомянутые в данных таблицы, были недоступны в процессе согласования. Обработка согласования завершилась успешно на доступных

менеджерах связей данных DB2. Таблица переведена в состояние отложенного согласования связей данных, поскольку обработка согласования на недоступных менеджерах связей данных DB2 отложена.

Действия пользователя: Эта таблица будет выведена из состояния отложенного согласования связей данных, когда согласование завершится успешно на всех менеджерах связей данных, упомянутых в данных таблицы. Запустите недоступные менеджерах связей данных DB2 и повторите согласование.

SQL2078N (новое сообщение)

SQL2078N **Ошибка добавления или отбрасывания менеджера связей данных DB2. Код причины = “<код-причины>”.**

Объяснение: Менеджер связей данных DB2 не удалось добавить или отбросить по причине, на которую указывает код причины:

- 01** Добавляемый менеджер связей данных DB2 уже зарегистрирован для этой базы данных.
- 02** Отбрасываемый менеджер связей данных DB2 не зарегистрирован для этой базы данных.
- 03** Параметр конфигурации менеджера баз данных DATALINKS имеет значение NO.
- 04** Для этой базы данных уже зарегистрировано максимально разрешенное число менеджеров связей данных DB2.

Действия пользователя: Необходимые действия зависят от кода причины:

- 01** Не пытайтесь добавлять менеджер связей данных DB2 повторно.
- 02** Не пытайтесь отбрасывать незарегистрированный менеджер связей данных DB2.
- 03** Задайте для параметра конфигурации менеджера баз данных DATALINKS значение YES при помощи команды UPDATE DATABASE MANAGER CONFIGURATION и повторите попытку.
- 04** Не пытайтесь добавлять больше менеджеров связей данных DB2, чем разрешается.

SQL2417N (новое сообщение)

SQL2417N **Архивирование журнала не допускается для баз данных без возможности восстановления.**

Объяснение: Команду архивирования журнала можно использовать только для баз данных, для которых

задана возможность восстановления. База данных допускает восстановление, если задан параметр LOGRETAIN или USEREXIT.

Действия пользователя: Переведите базу данных в режим с возможностью восстановления и повторите команду.

SQL2426N (новое сообщение)

SQL2426N **Инкрементное резервное копирование для этой базы данных не разрешено. Активируйте отслеживание модификаций и выполните полное резервное копирование этой базы данных.**

Объяснение: Инкрементное резервное копирование не будет разрешено, пока для этой базы данных не будет активировано отслеживание модификаций и не снята полная резервная копия. Полная резервная копия потребуется для восстановления последующих инкрементных копий.

Действия пользователя: Чтобы разрешить инкрементное резервное копирование для этой базы данных, активируйте отслеживание модификаций этой базы данных, введя следующую команду:

```
UPDATE DB CFG FOR имя-базы-данных USING TRACKMOD ON
```

Затем выполните полное резервное копирование базы данных.

SQL2571N (новое сообщение)

SQL2571N **Невозможно выполнить автоматическое инкрементное восстановление. Код причины : “<код-причины>”.**

Объяснение: В процессе автоматического инкрементного восстановления произошла ошибка. Утилита не смогла завершить восстановление. Утилита прекращает работу. Это сообщение появляется после восстановления начальных определений; причину ошибки указывают следующие коды причины:

- 1** Образ резервной копии, соответствующий заданной отметке времени, не найден в хронологии базы данных.

2 Ошибка при попытке определения, какие табличные пространства надо восстанавливать.

3 Требуемый образ резервной копии не найден в хронологии базы данных.

Действия пользователя: Введите команду RESTORE INCREMENTAL ABORT, чтобы очистить все ресурсы, возможно, созданные при обработке. Выполните инкрементное восстановление по этому образу резервной копии вручную.

SQL2572N (новое сообщение)

SQL2572N Попытка инкрементного восстановления образа в неверном порядке. При восстановлении табличного пространства “<имя-табличного-пространства>” произошла ошибка, поскольку перед восстановлением данного образа должен быть восстановлен образ с отметкой времени “<значение-отметки-времени>”.

Объяснение: При восстановлении инкрементных образов резервных копий надо соблюдать следующий порядок:

1. Восстановите сначала последний образ, чтобы указать DB2, до какого состояния вы хотите восстановить базу данных.
2. Восстановите полный образ базы данных или табличного пространства, предшествующий набору инкрементных образов.

3. Восстановить набор инкрементных (разностных) образов в хронологическом порядке их создания.
4. Восстановите последний образ второй раз.

Каждое табличное пространство в образе резервной копии содержит сведения о предыдущих образах, которые надо восстановить перед успешным восстановлением данного образа. Образ с отметкой времени, указанной в сообщении, необходимо восстановить прежде, чем восстанавливать образ, который вызвал данное сообщение. Возможно, перед указанным образом надо восстановить и другие образы, однако это было первое табличное пространство, для которого встретилась ошибка.

Действия пользователя: Установите правильный порядок набора образов инкрементных копий и продолжите процесс восстановления.

SQL2573N (новое сообщение)

SQL2573N Инкрементный образ резервной копии должен восстанавливаться при помощи операции инкрементного восстановления.

Объяснение: Была предпринята попытка операции восстановления с использованием инкрементного образа резервной копии. Инкрементный образ нельзя восстановить сам по себе; его надо обрабатывать в ходе операции инкрементного восстановления.

Утилита прекращает работу.

Действия пользователя: Чтобы восстановить этот образ резервной копии, повторите команду RESTORE с модификатором INCREMENTAL. Для неинкрементной операции восстановления надо задать неинкрементный образ резервной копии.

SQL2574N (новое сообщение)

SQL2574N Резервная копия, восстанавливаемая в ходе операции инкрементного восстановления, не должна быть новее образа назначения.

Объяснение: Образ назначения - это первый образ, восстанавливаемый в ходе операции инкрементного восстановления. Этот образ содержит определения табличных пространств и другие управляющие структуры для восстанавливаемой базы данных. Утилита RESTORE в ходе операции инкрементного восстановления не может восстановить образ новее образа назначения, поскольку это могло бы повредить базу данных.

Операция инкрементного восстановления завершилась неудачно, так как пыталась восстановить образ резервной копии с отметкой времени больше, чем у образа назначения.

Утилита прекращает работу.

Действия пользователя: Повторите команду, задав образ резервной копии с отметкой времени меньше, чем у образа назначения.

SQL2575N (новое сообщение)

SQL2575N Отметка времени заданного образа резервной копии меньше, чем у последнего восстановленного для табличного пространства “<номер-табличного-пространства>” образа. Отметка времени последнего восстановленного образа - “<отметка-времени>”.

Объяснение: Для инкрементного восстановления образы резервных копий должны восстанавливаться в хронологическом порядке от самого раннего к самому

позднему для каждого табличного пространства. В операции инкрементного восстановления был задан образ резервной копии с отметкой времени, более ранней, чем у предыдущего восстановленного образа заданного табличного пространства.

Утилита прекращает работу.

Действия пользователя: Повторите команду, задав образ резервной копии с отметкой времени позже, чем у последней восстановленной для этого табличного пространства копии.

SQL2576N (новое сообщение)

SQL2576N Табличное пространство “<имя-табличного-пространства>” восстановлено в ходе операции инкрементного восстановления, но в команде RESTORE не задано условие INCREMENTAL.

надо задать условие INCREMENTAL.

Утилита прекращает работу.

Действия пользователя: Повторите команду RESTORE, включив в нее условие INCREMENTAL.

Объяснение: Для инкрементного восстановления табличного пространства в каждой команде RESTORE

SQL4942N (новый текст)

Для этого сообщения были расширены разделы Объяснение и Действия пользователя. Теперь они выглядят так:

Объяснение: Вложенный оператор SELECT производит выборку в переменную хоста "<имя>", но тип данных переменной и соответствующего элемента списка SELECT несовместимы. Если тип данных столбца - дата и время, то тип данных переменной должен быть символьным с соответствующей минимальной длиной. Оба элемента могут также быть или числовыми, или символьными, или графическими. Для пользовательского типа данных переменная хоста может быть определена с соответствующим встроенным типом данных, который несовместим с типом результата функции преобразования FROM SQL, определенной в группе преобразования для этого оператора. Функцию невозможно завершить.

Действия пользователя: Убедитесь, что определения таблицы активны и что эта переменная хоста имеет правильный тип данных. Для пользовательского структурированного типа возможно, что переменная хоста определена со связанным встроенным типом данных, который несовместим с типом результата функции преобразования SQL FROM, определенной в группе преобразования для этого оператора. В случае пользовательского типа данных возможно, что эта переменная хоста, определена со связанным типом данных, который не совместим с типом результата функции преобразования SQL FROM, определенной в группе преобразования для этого оператора.

SQL5012N (новое сообщение)

SQL5012N Тип переменной хоста “<переменная-хоста>” - не точный числовой тип.

• Тип переменной хоста - десятичный, но масштаб - не ноль. Числа десятичного типа данных масштаба ноль не должны содержать знаков после запятой.

Объяснение: Задана переменная хоста “<переменная-хоста>”, недопустимая в том контексте, где она используется. Переменная хоста “<переменная-хоста>” задана как часть условия ABSOLUTE или RELATIVE в операторе FETCH, или в условии ROWS оператора FETCH или INSERT. Эту переменную хоста нельзя использовать по одной из следующих причин:

Оператор невозможно обработать.

Действия пользователя: Измените переменную хоста, чтобы ее тип стал точным числовым типом.

sqlcode: -5012

sqlstate: 42618

• Тип этой переменной хоста - не точный числовой тип. Точными числовыми типами считаются десятичные типы данных с масштабом 0и целые типы данных.

SQL6583N (новое сообщение)

SQL6583N Определение ключа разделения несовместимо с режимом загрузки многораздельной базы данных “<режим-загрузки>”.

PARTITION_AND_LOAD, а модификатор identityoverride не задан.

Действия пользователя: Либо выберите режим загрузки PARTITION_AND_LOAD, либо задайте модификатор identityoverride, либо уберите столбец идентификации из определения ключа разделения.

Объяснение: В качестве части ключа разделения задан столбец идентификации, но не выбран режим загрузки

SQL2005N (новое сообщение)

SQL2005N Предел внутреннего идентификатора “<предел>” превышен для объекта типа “<тип-объекта>”.

Объяснение: Внутренний идентификатор однозначно идентифицирует объекты типа “<тип-объекта>”. Превышен предел для внутренних идентификаторов этого типа объектов. Это могло произойти во время выполнения оператора CREATE DISTINCT TYPE, CREATE FUNCTION, CREATE PROCEDURE или CREATE SEQUENCE, или оператора ALTER TABLE или

CREATE TABLE, определяющего столбец идентификации.

Оператор невозможно обработать.

Действия пользователя: Отбросьте ненужные объекты типа “<тип-объекта>”.

sqlcode: -20005

sqlstate: 54035

SQL20117N (изменен код причины 1)

Для сообщения SQL20117N изменен следующий код причины:

Код причины 1

В разделе "Объяснение": В спецификации окна задан RANGE или ROWS без ORDER BY.

В разделе "Действия пользователя": Добавьте условие окна ORDER BY для каждой спецификации окна, где задан RANGE или ROWS.

SQL20121N (новое сообщение)

SQL20121N Для указателя “<имя-указателя>” можно указать либо WITH RETURN, либо SCROLL.

Объяснение: Для указателя “<имя-указателя>” задано и WITH RETURN, и SCROLL, что не разрешено.

Оператор невозможно обработать.

Действия пользователя: Исправьте оператор DECLARE CURSOR, либо задав NO SCROLL (можно удалить ключевое слово SCROLL), либо удалив условие WITH RETURN.

sqlcode: -20121

sqlstate: 428F3

SQL20133N (новое сообщение)

SQL20133N Операцию “<имя-операции>” нельзя выполнить над внешней подпрограммой “<имя-подпрограммы>”. Эту операцию можно выполнять только над подпрограммами SQL.

Объяснение: Вы пытались выполнить операцию “<имя-операции>” над внешней подпрограммой “<имя-подпрограммы>”. Однако эту операцию можно

выполнять только над подпрограммами SQL. Операция не выполнена.

Действия пользователя: Задайте для операции имя подпрограммы SQL.

sqlcode: -20133

sqlstate: 428F7

SQL20134N (новое сообщение)

SQL20134N Файл архива SQL (SAR) для подпрограммы “<имя-подпрограммы>” не может быть создан на сервере.

Объяснение: Создание архива SQL (SAR) для подпрограммы “<имя-подпрограммы>” завершилось неудачно, поскольку DB2 не смогла найти библиотеку или файл связывания для указанной подпрограммы. Файлы связывания доступны только для подпрограмм

SQL, созданных в DB2 Версии 7.1, пакет исправления 2 или более новой.

Действия пользователя: Пересоздайте процедуру на сервере с DB2 Версии 7.1, пакет исправления 2 или более новой, затем повторите операцию.

sqlcode: -20134

sqlstate: 55045

SQL20135N (новое сообщение)

SQL20135N Заданный архив SQL не соответствует среде назначения. Код причины = “<код-причины>”.

Объяснение: Заданный архив SQL не соответствует среде назначения по одной из следующих причин:

- 1 Операционная система среды назначения не совпадает с операционной системой, где был создан архив SQL.
- 2 Тип и уровень базы данных среды назначения не совпадает с типом и уровнем базы данных, где был создан архив SQL.

Действия пользователя: Обеспечьте соответствие среды, где создан архив SQL, среде назначения и повторите команду. Если среды не соответствуют друг другу, надо вручную создать подпрограмму SQL в среде назначения.

sqlcode: -20135

sqlstate: 55046

SQL20143N (новое сообщение)

SQL20143N Функция шифрования или дешифровки завершилась неудачно, поскольку не задано значение ENCRYPTION PASSWORD.

Объяснение: Не задано значение ENCRYPTION PASSWORD.

Действия пользователя: Выполните оператор SET ENCRYPTION PASSWORD, чтобы задать значение ENCRYPTION PASSWORD. Длина пароля должна быть не меньше 6 и не больше 127 байт.

sqlcode: -20143

sqlstate: 51039

SQL20144N (новое сообщение)

SQL20144N Пароль шифрования недопустим, поскольку его длина меньше 6 или больше 127 байт.

Объяснение: Для шифрования данных длина пароля должна быть от 6 до 127 байт.

Действия пользователя: Задайте пароль длиной от 6 до 127 байт.

sqlcode: -20144

sqlstate: 428FC

SQL20145N (новое сообщение)

SQL20145N Ошибка функции дешифровки. Пароль расшифровки не соответствует паролю, использованному для шифрования данных.

Объяснение: Для расшифровки данных надо использовать тот же пароль, что и для их шифрования.

Действия пользователя: Используйте для шифрования и дешифровки данных один и тот же пароль.

sqlcode: -20145

sqlstate: 428FD

SQL20146N (новое сообщение)

SQL20146N Ошибка функции дешифровки. Данные не были зашифрованы.

Объяснение: Данные должны быть результатом функции ENCRYPT.

Действия пользователя: Задайте данные - результат функции ENCRYPT.

sqlcode: -20146

sqlstate: 428FE

SQL20147N (новое сообщение)

SQL20147N Ошибка функции ENCRYPT. Многопроходное шифрование не поддерживается.

Объяснение: Уже зашифрованные данные нельзя шифровать вторично.

Действия пользователя: Задайте для шифрования незашифрованные данные.

sqlcode: -20147

sqlstate: 55048

SQL20148N (новое сообщение)

SQL20148N Последним оператором SQL составного тела подпрограммы “<имя-подпрограммы>” с конкретным именем “<конкретное-имя>” должен быть оператор RETURN.

Объяснение: Оператор RETURN должен быть последним оператором SQL составного тела строковой или табличной функции SQL. В теле подпрограммы не

должно быть других операторов RETURN.

Действия пользователя: Убедитесь, что в составном теле только один оператор RETURN, и он является последним оператором SQL.

sqlcode: -20148

sqlstate: 429BD

SQL20153N (новое сообщение)

SQL20153N Образ разделения базы данных находится в приостановленном состоянии.

Объяснение: Базу данных нельзя использовать, пока ее образ разделения находится в приостановленном состоянии.

Действия пользователя: Выполните команду db2inidb с одной из следующих опций, чтобы возобновить ввод-вывод для этого образа разделения:

- db2inidb <имя-базы-данных> as mirror
- db2inidb <имя-базы-данных> as snapshot

- db2inidb <имя-базы-данных> as standby

В многоузловой среде, прежде, чем использовать базу данных, надо запустить db2inidb на каждом узле. В многоузловой базе данных db2inidb может работать одновременно на нескольких узлах.

sqlcode: -20153

sqlstate: 55040

Новые значения SQLSTATE

Таблица 14. Новые значения SQLSTATE и текст

Значение SQLSTATE	Смысл
02502	Обнаружена дыра удаления или изменения.
23522	Диапазон значений для столбца идентификации или последовательности исчерпан.
24510	Попытка выполнения операции UPDATE или DELETE в дыре или изменения.
24512	Таблица результатов не согласуется с базовой таблицей.
24513	FETCH NEXT, PRIOR, CURRENT или RELATIVE недопустимы, поскольку положение указателя неизвестно.
36001	Указатель SENSITIVE нельзя определить для заданного оператора SELECT.
42620	Задан SCROLL только для чтения с условием UPDATE.
42837	Столбец нельзя изменить, поскольку его атрибуты несовместимы с текущими атрибутами этого столбца.
42872	Условия оператора FETCH несовместимы с определением указателя.
428F3	SCROLL и WITH RETURN - взаимоисключающие условия.
428F4	Заданное в операторе FETCH условие SENSITIVITY недопустимо для указателя.

Таблица 14. Новые значения SQLSTATE и текст (продолжение)

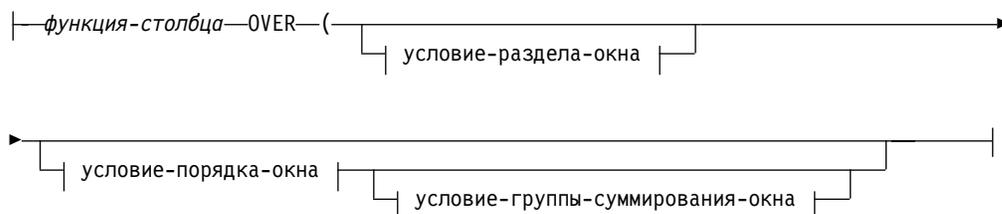
Значение SQLSTATE	Смысл
428F7	Попытка применить операцию, допустимую только для подпрограмм SQL, к внешней подпрограмме.
428F9	Выражение последовательности недопустимо в данном контексте.
428FA	Масштабом десятичного числа должен быть ноль.
428FC	Недопустимая длина пароля шифрования.
428FD	Пароль расшифровки не соответствует паролю, использованному для шифрования данных.
428FE	Данные не являются результатом функции ENCRYPT.
429BA	Со ссылкой на псевдоним или табличную функцию OLE DB надо использовать ключевое слово FEDERATED.
51035	Выражение PREVAL нельзя использовать, поскольку значение для этой последовательности в данном сеансе еще не генерировалось.
51039	Не задано значение ENCRYPTION PASSWORD.
55035	Таблицу нельзя отбросить, поскольку она защищена.
55045	Файл архива SQL (SAR) для подпрограммы не может быть создан, потому что на сервере отсутствует необходимый компонент.
55046	Заданный архив SQL не соответствует среде назначения.
55048	Зашифрованные данные нельзя шифровать.
560B1	Ошибка процедуры: набор результатов допускал прокрутку, но указатель не стоял перед первой строкой.
560B7	Для многострочного оператора INSERT использование выражения последовательности должно быть одинаковым для каждой строки.

Глава 29. SQL Reference

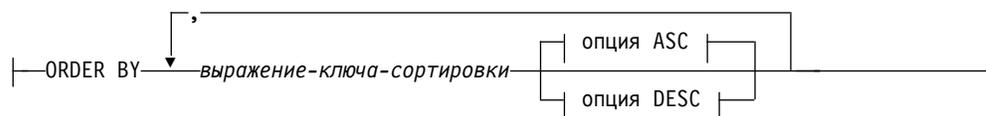
Функции OLAP

Следующее исправление надо внести в подраздел "OLAP Functions" раздела "Expressions" Главы 3.

aggregation-function:



условие-порядка-окна:



опция ASC:



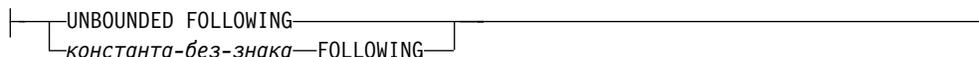
опция DESC:



условие-группы-суммирования-окна:



конец-группы:



В описании условия-порядка-окна:

NULLS FIRST

При упорядочении окна пустые значения считаются идущими в порядке сортировки перед всеми непустыми значениями.

NULLS LAST

При упорядочении окна пустые значения считаются идущими в порядке сортировки после всех непустых значений.

В описании условия-группы-суммирования-окна:

условие-группы-суммирования-окна

Группа суммирования строки R - это набор строк, определяемых относительно R в упорядочении строк раздела R. Это условие задает группу суммирования. Если это условие не задано, по умолчанию используется условие RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW, генерирующее результат суммирования с накоплением.

ROWS

Указывает, что группа суммирования определяется через число строк.

RANGE

Указывает, что группа суммирования определяется через расстояние от ключа сортировки.

начало-группы

Задаёт начальную точку для группы суммирования. Конец группы суммирования - текущая строка. Условие начало-группы эквивалентно условию группа-между типа "BETWEEN начало-группы AND CURRENT ROW".

группа-между

Задаёт начало и конец группы суммирования в виде ROWS или RANGE.

конец-группы

Задаёт конечную точку для группы суммирования. Начало группы суммирования - текущая строка. Условие *конец-группы* эквивалентно условию *группа-между* типа "BETWEEN CURRENT ROW AND *конец-группы*".

UNBOUNDED PRECEDING

Включить весь раздел до текущей строки. Это условие можно задать с ROWS или с RANGE. Его можно также задать с несколькими выражениями-ключей-сортировки в условии-порядка-окна.

UNBOUNDED FOLLOWING

Включить весь раздел после текущей строки. Это условие можно задать с ROWS или с RANGE. Его можно также задать с несколькими выражениями-ключей-сортировки в условии-порядка-окна.

CURRENT ROW

Задаёт начало или конец группы суммирования на основе текущей строки. Если задано условие ROWS, границей группы суммирования будет текущая строка. Если задано условие RANGE, граница группы суммирования будет включать набор строк с теми же значениями *выражений-ключей-сортировки*,

что и текущая строка. Это условие нельзя задавать в *границе-группы-2*, если в *границе-группы-1* задано значение FOLLOWING.

значение PRECEDING

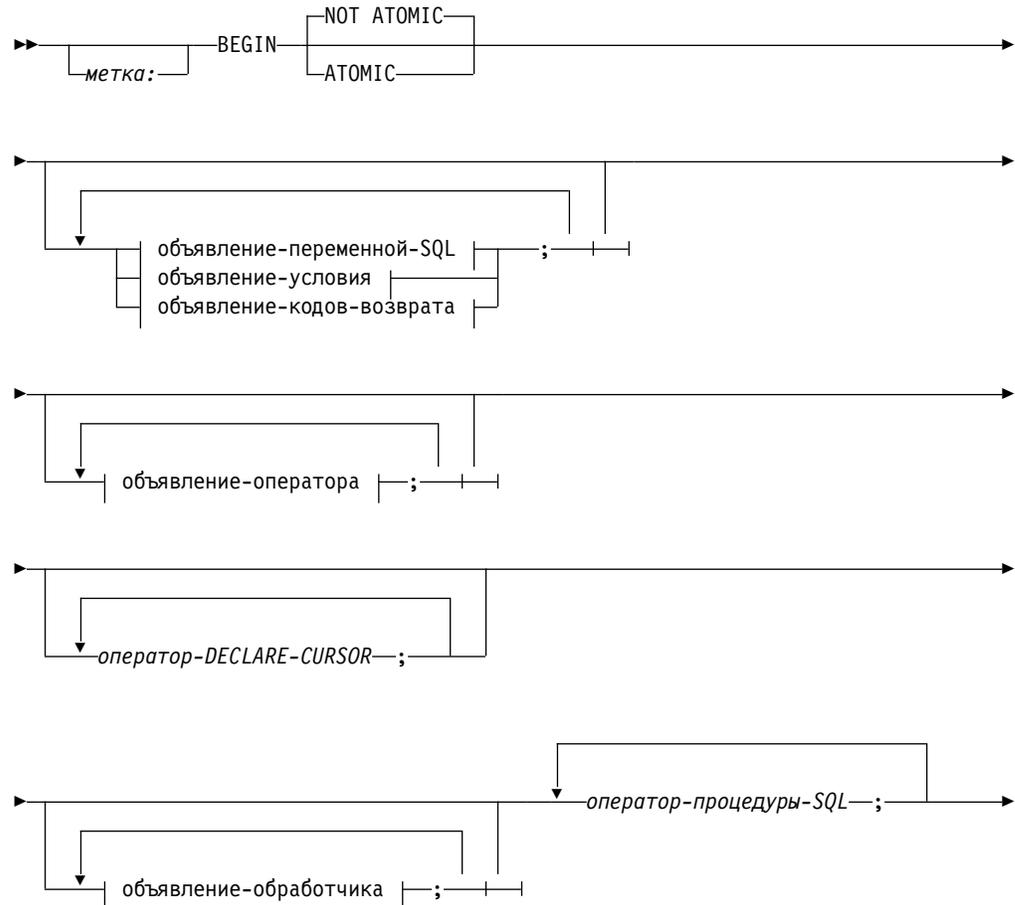
Задаёт диапазон или число строк, идущих перед текущей строкой. Если задано условие ROWS, *значение* - это положительное целое число, обозначающее число строк. Если задано условие RANGE, тип данных *значения* должен быть совместим с типом данных выражения-ключа-сортировки условия-порядка-окна. Может быть только одно выражение-ключа-сортировки и тип данных этого выражения-ключа-сортировки должен допускать вычитание. Это условие нельзя задавать в *границе-группы-2*, если в *границе-группы-1* задано CURRENT ROW или *значение* FOLLOWING.

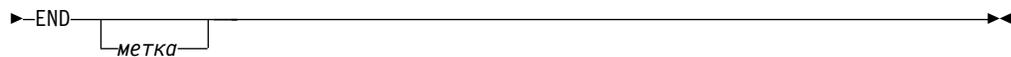
значение FOLLOWING

Задаёт диапазон или число строк, идущих после текущей строки. Если задано условие ROWS, *значение* - это положительное целое число, обозначающее число строк. Если задано условие RANGE, тип данных *значения* должен быть совместим с типом данных выражения-ключа-сортировки условия-порядка-окна. Может быть только одно выражение-ключа-сортировки и тип данных этого выражения-ключа-сортировки должен допускать сложение.

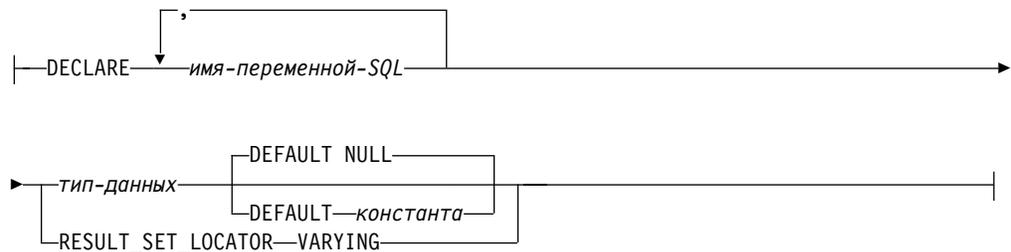
Процедуры и составные операторы SQL

Исправленная синтаксическая диаграмма составного оператора:

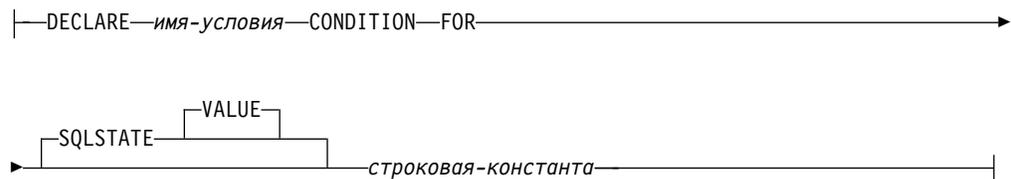




объявление-переменной-SQL:



объявление-условия:



объявление-оператора:

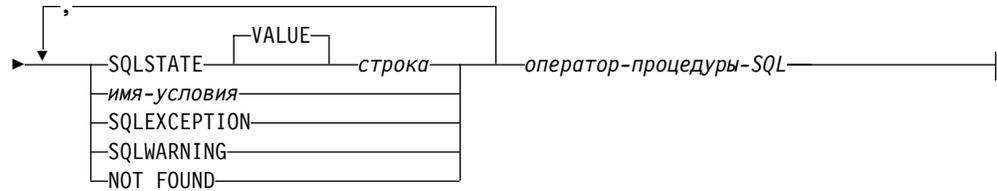


объявление-кодов-возврата:



объявление-обработчика:





В объявлении оператора объявляется список из одного или нескольких имен, локальных для этого составного оператора. Имя оператора не может совпадать с другим именем оператора в том же составном операторе.

LCASE и UCASE (Unicode)

В базе данных Unicode весь набор символов Unicode переводится в верхний (или в нижний) регистр в зависимости от значения свойств Unicode этих символов. Версии символов ASCII двойной ширины и римские цифры теперь преобразуются правильным образом.

WEEK_ISO

Измените описание этой функции на следующее:

Схема - SYSFUN.

Для переданного значения аргумента возвращает номер недели в году в виде целого значения от 1 до 53. Неделя начинается с понедельника и всегда содержит 7 дней. Неделя номер 1 - это первая неделя года, содержащая четверг, или, что то же самое, неделя года, содержащая 4 января. Поэтому возможно, что до трех дней в начале года будут относиться к последней неделе прошлого года. И наоборот, до 3 дней в конце года могут относиться к первой неделе следующего года. Аргумент должен иметь тип данных дата, отметка времени или правильное строковое представление даты или отметки времени, но не CLOB или LONG VARCHAR.

Результат функции имеет тип INTEGER. Результат может иметь пустое значение; если аргумент имеет пустое значение, результат будет иметь пустое значение.

Пример:

Ниже показаны примеры результатов функций WEEK_ISO и DAYOFWEEK_ISO.

DATE	WEEK_ISO	DAYOFWEEK_ISO
1997-12-28	52	7
1997-12-31	1	3
1998-01-01	1	4
1999-01-01	53	5
1999-01-04	1	1
1999-12-31	52	5
2000-01-01	52	6
2000-01-03	1	1

Правила именования и неявные спецификаторы имен объектов

В этот раздел главы 3 добавьте следующее примечание:

- В следующих именах в контексте процедур SQL можно использовать только символы, разрешенные в обычных идентификаторах, даже если эти имена введены в ограничителях:
- имя-условия
 - метка

- имя-параметра
- имя-процедуры
- имя-переменной-SQL
- имя-оператора

Queries (Запросы)

оператор select/условие fetch-first

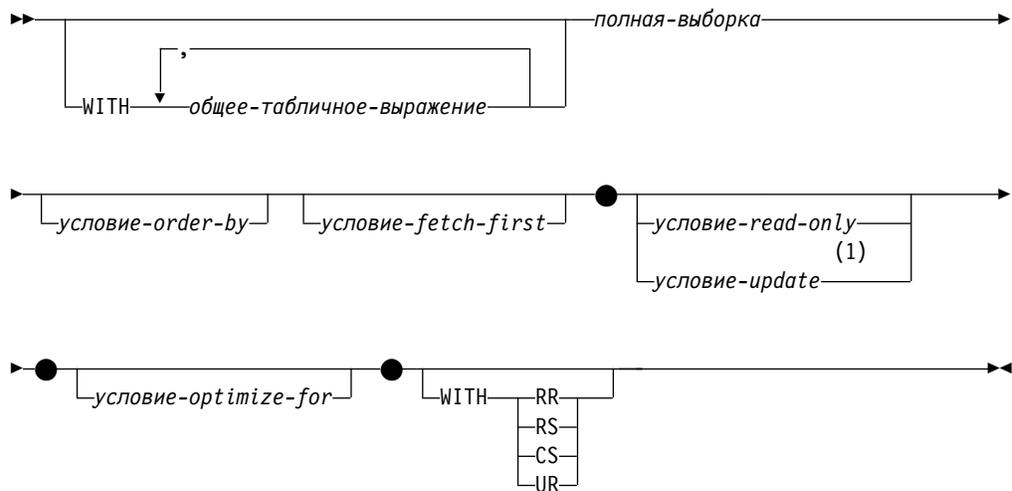
Последний абзац описания условия-выбрать-первым:

Specification of the fetch-first-clause in a select-statement makes the cursor not deletable (read-only). This clause cannot be specified with the FOR UPDATE clause.

является неверным; его нужно удалить.

оператор select/синтаксическая диаграмма

Синтаксическую диаграмму надо изменить на:



Примечания:

- 1 Условия UPDATE и ORDER BY нельзя задавать вместе в одном операторе SELECT.

К описанию под синтаксической диаграммой добавьте следующий абзац:

Необязательное условие WITH задает уровень изоляции, с которым выполняется оператор SELECT.

- RR - многократное чтение
- RS - стабильность чтения
- CS - стабильность на уровне указателя
- UR - чтение неприятого

По умолчанию для оператора используется уровень изоляции пакета, в котором был связан этот оператор.

Библиотеки, используемые оператором CREATE WRAPPER в Linux

В Linux используются библиотеки LIBDRDA.SO и LIBSQLNET.SO, а не LIBDRDA.A и LIBSQLNET.A, как, возможно, указано в вашей документации.

Теперь поддерживается возможность изменения ключа разделения

Теперь поддерживается изменение ключа разделения. В Главе 6 перечисленные ниже утверждения **не** касаются случая, когда DB2_UPDATE_PART_KEY=ON:

Примечание: Если DB2_UPDATE_PART_KEY=OFF, эти ограничения остаются в силе.

Оператор: ALTER TABLE

Правила

- Столбец ключа разделения таблицы не может быть изменен (SQLSTATE 42997).
- Допускающий пустые значения столбец ключа разделения не может быть включен в качестве столбца внешнего ключа, если отношение определено с условием ON DELETE SET NULL (SQLSTATE 42997).

Оператор: CREATE TABLE

Правила

- Столбец ключа разделения таблицы не может быть изменен (SQLSTATE 42997).
- Допускающий пустые значения столбец ключа разделения не может быть включен в качестве столбца внешнего ключа, если отношение определено с условием ON DELETE SET NULL (SQLSTATE 42997).

Оператор: DECLARE GLOBAL TEMPORARY TABLE PARTITIONING KEY (имя-столбца,...)

Примечание: Столбец ключа разделения таблицы не может быть изменен (SQLSTATE 42997).

Оператор: SET переходная-переменная

Правила

- Если оператор используется в триггере BEFORE UPDATE, имя-столбца, заданное в качестве переходной-переменной, не может быть столбцом ключа разделения (SQLSTATE 42997).

Оператор: UPDATE

Сноски

- 108 Столбец ключа разделения не может быть изменен (SQLSTATE 42997). Чтобы изменить ключ разделения, необходимо удалить строку данных и вставить ее снова.

Длинные ключи индекса для баз данных Unicode

CREATE TABLE

Первичные ключи и ключи уникальности, в которые входят столбцы переменной длины, могут иметь размер больше 255, если переменная реестра

DB2_INDEX_2BYTEVARLEN имеет значение ON. Внешние ключи могут быть определены на столбцах переменной длины, размер которых больше 255 байт.

CREATE INDEX

Индексы могут быть определены на столбцах переменной длины, размер которых больше 255 байт, если переменная реестра DB2_INDEX_2BYTEVARLEN имеет значение ON.

ALTER TABLE

Если при определении индекса переменная реестра DB2_INDEX_2BYTEVARLEN имеет значение ON, длина столбцов переменной длины, входящих в этот индекс, включая первичные ключи и ключи уникальности, может быть изменена на значение более 255 байт. Реально длина столбца переменной длины, входящего во внешний ключ, теперь может быть изменена на длину более 255 байт независимо от значения этой переменной реестра. Однако данные длиной более 255 байт можно вставить в таблицу, только если столбец соответствующего первичного ключа имеет длину более 255 байт, что возможно, только если этот первичный ключ создан, когда эта переменная реестра имела значение ON.

Включение новых встроенных скалярных функций SQL

FixPak 2 и FixPak 3 Версии 7.1 содержит новые встроенные скалярные функции SQL. Описание этих функций смотрите в обновлениях справочника *SQL Reference*. Новые функции не включаются автоматически в каждой базе данных при обновлении кода сервера базы данных до нового уровня. Чтобы включить новые функции, системный администратор должен выполнить команду **db2updv7**, указав каждую базу данных на сервере. Эта команда создает в базе данных запись, гарантирующую, что никакой объект базы данных, созданный до выполнения этой команды, не использует существующую функцию с сигнатурой, которая может совпасть с сигнатурой новой функции.

ABS или ABSVAL

ABS или ABSVAL

►► ABS (выражение) ◄◄
ABSVAL

Схема - SYSIBM.

Эта функция впервые стала доступна в FixPak 2 Версии 7.1.

Примечание: Версия SYSFUN функции ABS (или ABSVAL) по-прежнему доступна.

Возвращает абсолютную величину аргумента.

Аргументом является выражение, возвращающее значение любого встроенного числового типа данных.

У результата функции такой же тип данных и атрибут длины, как у аргумента. Если значение аргумента может быть пустым или если в конфигурации базы данных для параметра DFT_SQLMATHWARN задано значение YES, значение результата может быть пустым; если значение аргумента - пустое, то и значение результата - пустое.

Например:
ABS(-51234)

возвращает INTEGER со значением 51234.

MULTIPLY_ALT

MULTIPLY_ALT

►—MULTIPLY_ALT—(точное-числовое-выражение; точное-числовое-выражение)—◄

Схема - SYSIBM.

Эта функция впервые стала доступна в FixPak 2 Версии 7.1.

Скалярная функция MULTIPLY_ALT возвращает десятичное значение произведения двух аргументов. Это альтернатива операции умножения, особенно когда сумма точностей аргументов превосходит 31.

Аргументы могут быть любого встроенного точного числового типа (DECIMAL, BIGINT, INTEGER или SMALLINT).

Тип результата функции - DECIMAL. Точность и масштаб результата определяются так (p и s означают точность и масштаб первого аргумента, а p' и s' - точность и масштаб второго аргумента).

- Точность - MIN(31, p + p')
- Масштаб:
 - 0, если масштаб обоих аргументов - 0
 - MIN(31, s+s'), если p+p' не больше 31
 - MAX(MIN(3, s+s'), 31-(p-s+p'-s')), если p+p' больше 31.

Значение результата может быть пустым, если значение хотя бы одного из аргументов может быть пустым, или если в конфигурации базы данных для параметра DFT_SQLMATHWARN задано значение yes; значение результата пусто, если значение одного из аргументов пусто.

Если выполняются десятичные расчеты с масштабом хотя бы 3 и сумма точностей превосходит 31, функция MULTIPLY_ALT предпочтительнее операции умножения. В этом случае внутренние вычисления выполняются так, чтобы избежать переполнений. Потом окончательный результат приводится к типу результата, используя, если нужно, усечение, чтобы соответствовать масштабу. Обратите внимание на то, что если масштаб - 3, переполнение окончательного результата все же возможно.

Ниже приведен пример, сравнивающий типы результата функции MULTIPLY_ALT и операции умножения.

Тип аргумента 1	Тип аргумента 2	Результат функции MULTIPLY_ALT	Результат операции умножения
DECIMAL(31,3)	DECIMAL(15,8)	DECIMAL(31,3)	DECIMAL(31,11)
DECIMAL(26,23)	DECIMAL(10,1)	DECIMAL(31,19)	DECIMAL(31,24)
DECIMAL(18,17)	DECIMAL(20,19)	DECIMAL(31,29)	DECIMAL(31,31)

Тип аргумента 1	Тип аргумента 2	Результат функции MULTIPLY_ALT	Результат операции умножения
DECIMAL(16,3)	DECIMAL(17,8)	DECIMAL(31,9)	DECIMAL(31,11)
DECIMAL(26,5)	DECIMAL(11,0)	DECIMAL(31,3)	DECIMAL(31,5)
DECIMAL(21,1)	DECIMAL(15,1)	DECIMAL(31,2)	DECIMAL(31,2)

Пример:

Перемножаем два значения, тип данных первого аргумента - DECIMAL(26,3), второго - DECIMAL(9,8). Тип данных результата - DECIMAL(31,7).

```
values multiply_alt(98765432109876543210987,654;5,43210987)
1
```

```
-----
536504678578875294857887,5277415
```

Обратите внимание на то, что полное произведение этих двух чисел - 536504678578875294857887,52774154498, но последние 4 цифры были отброшены, чтобы соответствовать масштабу типа данных результата. Использование с теми же значениями операции умножения приведет к арифметическому переполнению, так как тип данных результата - DECIMAL(31,11) - поддерживает только 20 знаков слева от десятичной точки, а у значения результата - 24 знака.

ROUND

ROUND

►—ROUND—(выражение1; выражение2)—►

Схема - SYSIBM.

Эта функция впервые стала доступна в FixPak 2 Версии 7.1.

Примечание: Версия SYSFUN функции ROUND по-прежнему доступна.

Функция ROUND возвращает *выражение1*, округленное до *выражение2* цифр после разделителя целой и дробной части, если *выражение2* положительно, или до *выражение2* цифр до разделителя целой и дробной части - если оно отрицательно.

Если *выражение1* положительно, пятерка округляется в большую сторону до следующего положительного числа. Например, ROUND(3,5;0) = 4. Если *выражение1* отрицательно, пятерка округляется в меньшую сторону до следующего отрицательного числа. Например, ROUND(-3,5;0) = -4.

выражение1

Выражение, возвращающее значение любого встроеного числового типа данных.

expression2

Выражение, возвращающее короткое или длинное целое. Если значение *выражения2* неотрицательно, оно указывает, до скольких цифр справа от разделителя целой и дробной части округлять. Когда значение *выражения2* отрицательно, его абсолютная величина указывает, до скольких цифр слева от разделителя целой и дробной части округлять.

Если *выражение2* неотрицательно, *выражение1* округляется до абсолютного значения *выражения2* цифр справа от разделителя целой и дробной части. Если значение *выражения2* больше чем масштаб *выражения1*, значение не меняется, только точность значения результата будет больше на 1. Например, ROUND(748,58;5) = 748,58, где точность теперь 6, а масштаб остался 2.

Если *выражение2* отрицательно, *выражение1* округляется до абсолютного значения *выражения2*+1 цифр слева от разделителя целой и дробной части.

Если абсолютное значение отрицательного *выражения2* больше, чем число цифр слева от разделителя целой и дробной части, результат будет 0. Например, ROUND(748,58;-4) = 0.

Тип данных и атрибут длины результата те же, что и у первого аргумента, только точность увеличивается на один, если тип *выражения1* - DECIMAL или NUMERIC и точность меньше 31. Например, если тип аргумента DECIMAL(5,2), тип результата будет DECIMAL(6,2). Если тип аргумента DECIMAL(31,2), тип результата останется DECIMAL(31,2). Масштаб остается такой же, как у первого аргумента.

Значение результата может быть пустым, если значение хотя бы одного из аргументов может быть пустым, или если в конфигурации базы данных для параметра DFT_SQLMATHWARN задано значение yes. Если значение одного из аргументов пусто, значение результата будет пустым.

Примеры:

Вычислим округления числа 873.726 до 2, 1, 0, -1, -2, -3 и -4 десятичных знаков соответственно.

```
VALUES (ROUND(873,726; 2),
        ROUND(873,726; 1),
        ROUND(873,726; 0),
        ROUND(873,726;-1),
        ROUND(873,726;-2),
        ROUND(873,726;-3),
        ROUND(873,726;-4) )
```

Этот пример возвращает:

1	2	3	4	5	6	7
873,730	873,700	874,000	870,000	900,000	1000,000	0,000

Вычисляем как положительные, так и отрицательные числа.

```
VALUES (ROUND(3,5; 0),
        ROUND(3,1; 0),
        ROUND(-3,1; 0),
        ROUND(-3,5;0) )
```

Этот пример возвращает:

1	2	3	4
4,0	3,0	-3,0	-4,0

Language Elements/DATALINK Assignments (Элементы языка/Присваивания DATALINK)

Абзац в этом разделе изменен на следующий:

Учтите, что размер параметра URL или результата функции совпадает с размером входных или выходных параметров и ограничен длиной столбца DATALINK. Однако в некоторых случаях с возвращаемым значением URL передается элемент доступа. В случаях, когда это возможно, выходная переменная должна иметь достаточный размер, чтобы вместить элемент доступа. Поэтому задаваемая на входе действительная длина комментария и URL в полной форме не должна превосходить длину DATALINK. Если она превосходит эту длину, возникает ошибка.

Функция ENCRYPT (новая скалярная функция)

ENCRYPT

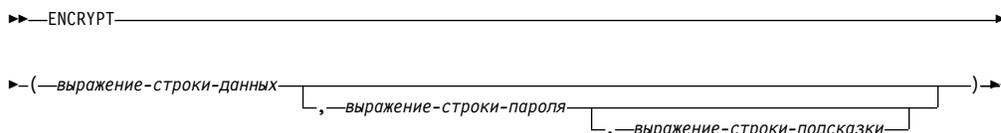


Схема - SYSIBM.

Эта функция впервые стала доступна в FixPak 3 Версии 7.1.

Функция ENCRYPT возвращает значение, полученное в результате шифрования *выражения-строки-данных*. Используемый для шифрования пароль - это или значение *выражение-строки-пароля*, или значение ENCRYPTION PASSWORD (заданное с помощью оператора SET ENCRYPTION PASSWORD).

выражение-строки-данных

Выражение, возвращающее значение типа CHAR или VARCHAR, которое должно быть зашифровано. Атрибут длины для типа данных *выражение-строки-данных* не должен превосходить 32663, если аргумент *выражение-строки-подсказки* не задан, и 32631, если аргумент *выражение-строки-подсказки* задан (SQLSTATE 42815).

выражение-строки-пароля

Выражение, возвращающее значение типа CHAR или VARCHAR длиной не менее 6 и не более 127 байт (SQLSTATE 428FC). Это значение представляет собой пароль, используемый для шифрования *выражение-строки-данных*. Если аргумент пароля не задан или имеет пустое значение, для шифрования данных используется значение ENCRYPTION PASSWORD, заданное для этого сеанса (SQLSTATE 51039).

выражение-строки-подсказки

Выражение, возвращающее значение типа CHAR или VARCHAR длиной до 32 байт, помогающее владельцу данных вспомнить пароль (например, 'океан') в качестве подсказки для пароля ('Тихий'). Если задано значение подсказки, подсказка включается в результат и ее можно получить с помощью функции GETHINT. Если этот аргумент не задан или имеет пустое значение, подсказка не включается в результат.

Тип данных результата этой функции - VARCHAR FOR BIT DATA.

Атрибут длины результата:

- Если задан необязательный параметр подсказки, атрибут длины нешифрованных данных + 8 байт + число байт до следующей 8-байтной границы + 32 байта для подсказки.

- Если подсказка не задана, атрибут длины нешифрованных данных + 8 байт + число байт до следующей 8-байтной границы.

Если значение первого аргумента может быть пустым, значение результата может быть пустым; если первый аргумент имеет пустое значение, то и результат будет иметь пустое значение.

Учтите, что зашифрованный результат имеет большую длину, чем значение *выражение-строки-данных*. Поэтому присваивая зашифрованные значения, убедитесь, что размер места назначения достаточен, чтобы вместить все зашифрованные значения.

Примечания:

- **Алгоритм шифрования:** Внутренний алгоритм шифрования использует блочный шифр RC2 с дополнением, 128-битный секретный ключ строится на основе пароля с помощью алгоритма хэширования MD2.
- **Пароли шифрования и данные:** За поддержку паролей отвечает пользователь. После того, как данные зашифрованы, для их дешифрования можно использовать только пароль, с которым они были зашифрованы (SQLSTATE 428FD). Будьте осторожны при использовании переменных CHAR для задания значений пароля, так как они могут быть дополнены пробелами. Результат шифрования может содержать пустой символ-ограничитель и другие специальные символы.

Определение столбца таблицы: При определении столбцов и типов данных, которые будут содержать зашифрованные данные, всегда вычисляйте атрибут длины, как указано ниже. Для зашифрованных данных без подсказки:

Длина столбца зашифрованных данных = максимальная длина нешифрованных данных + 8 байт + число байт до следующей 8-байтной границы.

Для зашифрованных данных с подсказкой:

Длина столбца зашифрованных данных = максимальная длина нешифрованных данных + 8 байт + число байт до следующей 8-байтной границы + 32 байта для подсказки.

Все операции присваивания или преобразования в тип меньшей длины, чем указанная выше, могут привести в результате к невозможности расшифровать данные и к **потере данных**. Пробелы (возможные значения зашифрованных данных) могут быть усечены при сохранении данных в слишком коротком столбце.

Пример вычислений длины столбца

Максимальная длина нешифрованных данных	6 байт
8 байт	8 байт
Число байт до следующей 8-байтной границы	2 байта

Длина столбца зашифрованных данных	16 байт
Максимальная длина нешифрованных данных	32 байта
8 байт	8 байт
Число байт до следующей 8-байтной границы	8 байт

Длина столбца зашифрованных данных	48 байт

- **Управление зашифрованными данными:** Шифрованные данные могут быть дешифрованы только на серверах, которые поддерживают функции дешифрации, соответствующие функции ENCRYPT. Поэтому репликацию столбцов, содержащих зашифрованные данные, следует производить только на серверы, поддерживающие функцию DECRYPT_BIN или DECRYPT_CHAR.

ENCRYPT

Дополнительную информацию об использовании этой функции смотрите в разделах “Функция DECRYPT_BIN и DECRYPT_CHAR (новая скалярная функция)” и “Функция GETHINT (новая скалярная функция)” на стр. 299.

Примеры:

Пример 1: В этом примере для хранения пароля шифрования используется значение ENCRYPTION PASSWORD.

```
SET ENCRYPTION PASSWORD = 'Ben123';  
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832');
```

Пример 2: В этом примере пароль шифрования задается явно.

```
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832', 'Ben123', '');
```

Пример 3: Сохраняется подсказка 'океан', помогающая пользователю вспомнить пароль шифрования 'Тихий'.

```
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832', 'Тихий', 'океан');
```

Функция DECRYPT_BIN и DECRYPT_CHAR (новая скалярная функция)

DECRYPT_BIN и DECRYPT_CHAR



Схема - SYSIBM.

Эта функция впервые стала доступна в FixPak 3 Версии 7.1.

Функции DECRYPT_BIN и DECRYPT_CHAR возвращают значение, полученное в результате дешифрования *шифрованных-данных*. Используемый для дешифрования пароль - это или значение *выражения-строки-пароля*, или значение ENCRYPTION PASSWORD (заданное с помощью оператора SET ENCRYPTION PASSWORD). Функции DECRYPT_BIN и DECRYPT_CHAR могут дешифровать только значения, зашифрованные функцией ENCRYPT (SQLSTATE 428FE).

шифрованные-данные

Выражение, возвращающее значение CHAR FOR BIT DATA или VARCHAR FOR BIT DATA - полная строка шифрованных данных, созданная функцией ENCRYPT.

выражение-строки-пароля

Выражение, возвращающее значение типа CHAR или VARCHAR длиной не менее 6 и не более 127 байт (SQLSTATE 428FC). Это должен быть тот же пароль, который использовался для шифрования данных, иначе возникнет ошибка дешифрования (SQLSTATE 428FD). Если аргумент пароля не задан или имеет пустое значение, для дешифрования данных используется значение ENCRYPTION PASSWORD, заданное для этого сеанса (SQLSTATE 51039).

Результат функции DECRYPT_BIN имеет тип VARCHAR FOR BIT DATA. Результат функции DECRYPT_CHAR имеет тип VARCHAR. Если *шифрованные-данные* содержат подсказку, эта функция не возвращает подсказку. Атрибут длины результата равен атрибуту длины типа данных *шифрованных-данных* минус 8 байт. Реальная длина возвращаемого функцией значения будет совпадать с длиной исходной строки, которая была зашифрована. Если *шифрованные-данные* содержат дополнительные байты после зашифрованной строки, эти байты не возвращаются этой функцией. Если значение первого аргумента может быть пустым, значение результата может быть пустым; если первый аргумент имеет пустое значение, то и результат будет иметь пустое значение.

Если данные дешифруются в другой системе, использующей кодовую страницу, отличающуюся от кодовой страницы, в которой выполнялось шифрование, длина дешифрованного значения при преобразовании его в кодовую страницу базы данных может увеличиться. В таких ситуациях значение *шифрованные-данные* должно быть преобразовано в строку VARCHAR с большим числом байт.

Дополнительную информацию об использовании этой функции смотрите в разделах “Функция ENCRYPT (новая скалярная функция)” на стр. 296 и “Функция GETHINT (новая скалярная функция)”.

Примеры:

Пример 1: В этом примере для хранения пароля шифрования используется значение ENCRYPTION PASSWORD.

```
SET ENCRYPTION PASSWORD = 'Ben123';
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832');
SELECT DECRYPT_CHAR(SSN)
FROM SSN;
```

Возвращается значение '289-46-8832'.

Пример 2: В этом примере пароль шифрования передается явно.

```
SELECT DECRYPT_CHAR(SSN, 'Ben123')
FROM SSN;
```

Возвращается значение '289-46-8832'.

Функция GETHINT (новая скалярная функция)

GETHINT

►►—GETHINT—(—*шифрованные-данные*—)—————►►

Схема - SYSIBM.

Эта функция впервые стала доступна в FixPak 3 Версии 7.1.

Функция GETHINT возвращает подсказку пароля, если она содержится в *шифрованных-данных*. Подсказка пароля - это фраза, помогающая владельцу данных вспомнить пароль (например, 'океан' в качестве подсказки для пароля 'Тихий').

шифрованные-данные

GETHINT

Выражение, возвращающее значение CHAR FOR BIT DATA или VARCHAR FOR BIT DATA - полная строка зашифрованных данных, созданная функцией ENCRYPT (SQLSTATE 428FE).

Результат функции имеет тип VARCHAR(32). Значение результата может быть пустым; если параметр подсказки не добавлен в *зашифрованные-данные* функцией ENCRYPT или если первый аргумент имеет пустое значение, результат будет иметь пустое значение.

Дополнительную информацию об использовании этой функции смотрите в разделах “Функция DECRYPT_BIN и DECRYPT_CHAR (новая скалярная функция)” на стр. 298 и “Функция ENCRYPT (новая скалярная функция)” на стр. 296.

Пример:

В этом примере сохраняется подсказка 'океан', помогающая пользователю вспомнить пароль шифрования 'Тихий'.

```
INSERT INTO EMP (SSN) VALUES ENCRYPT('289-46-8832', 'Тихий', 'океан');  
SELECT GETHINT(SSN)  
FROM EMP;
```

Возвращается значение 'океан'.

CREATE SEQUENCE (новый оператор SQL)

CREATE SEQUENCE

Оператор CREATE SEQUENCE создает последовательность на сервере прикладных программ.

Область действия

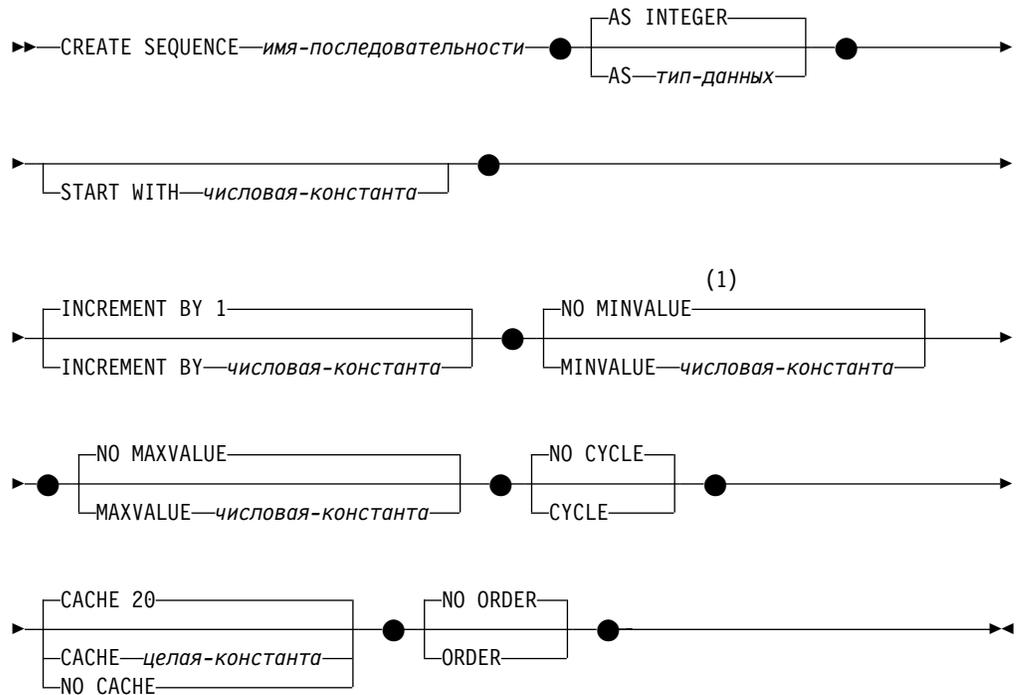
Этот оператор может быть встроен в прикладную программу или выполняться как динамический оператор SQL. Это выполняемый оператор, который может быть динамически подготовлен. Но если применяется опция связывания DYNAMICRULES BIND, этот оператор нельзя подготовить динамически (SQLSTATE 42509).

Авторизация

Привилегии ID авторизации этого оператора, должны содержать, как минимум, одну из следующих привилегий:

- Привилегию CREATEIN для неявно или явно заданной схемы
- Полномочия SYSADM или DBADM

Синтаксис команды



Примечания:

- 1 Эти параметры можно задавать без пробелов: NOMINVALUE, NOMAXVALUE, NOCYCLE, NOCACHE и NOORDER. Варианты параметров, записанные в одно слово, можно задавать вместо вариантов из двух слов.

Параметры команды

имя-последовательности

Имя последовательности. Сочетание имени и неявно или явно заданного имени схемы не должно указывать на существующую последовательность на текущем сервере (SQLSTATE 42710).

CREATE SEQUENCE

Имя-последовательности без спецификатора - это идентификатор SQL. Имя со спецификатором - это спецификатор, после которого идут точка и идентификатор SQL. Спецификатор - это имя схемы.

Если для имени последовательности явно задано имя схемы, это имя схемы не может начинаться с 'SYS'; в противном случае возникает ошибка (SQLSTATE 42939).

AS *тип-данных*

Задаёт тип данных, который должен использоваться для значения последовательности. Этот тип данных может быть любым точным числовым типом (SMALLINT, INTEGER, BIGINT или DECIMAL) с масштабом 0 или пользовательским особым типом, исходным типом которого является точный числовой тип с масштабом 0 (SQLSTATE 42815). Тип по умолчанию - INTEGER.

START WITH *числовая-константа*

Задаёт первое значение для этой последовательности. Оно может быть любым положительным или отрицательным значением, которое можно присваивать столбцу типа данных, связанного с этой последовательностью (SQLSTATE 42820); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 428FA). Значение по умолчанию - MINVALUE для возрастающих последовательностей и MAXVALUE для убывающих последовательностей.

При достижении последовательностью ее максимального или минимального циклическая генерация не обязательно будет продолжена с этого начального значения. Условие START WITH может использоваться для запуска последовательности со значения, выходящего за границы цикла генерации. Диапазон циклической генерации определяется значениями MINVALUE и MAXVALUE.

INCREMENT BY *числовая-константа*

Задаёт разность между последовательными значениями последовательности. Она может быть любым положительным или отрицательным значением, которое можно присваивать столбцу типа данных, связанного с этой последовательностью (SQLSTATE 42820) и которое не превышает значения большой целой константы (SQLSTATE 42815); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 428FA).

Если это отрицательное значение, последовательность значений будет убывающей. Если это положительное значение, последовательность значений будет возрастающей. Если это значение равно 0 или если оно выходит за диапазон, определяемый значениями MINVALUE и MAXVALUE, будет генерироваться одно и то же значение, но последовательность все равно будет считаться возрастающей последовательностью. Значение по умолчанию - 1.

MINVALUE *или NO MINVALUE*

Задаёт минимальное значение, по достижении которого для убывающей последовательности прекращается генерация значений или начинается следующий цикл генерации значений; для возрастающей последовательности это значение, с которого начинается следующий цикл генерации после достижения максимального значения.

MINVALUE *числовая-константа*

Задаёт минимальное числовое значение. Оно может быть любым положительным или отрицательным значением, которое можно присваивать столбцу типа данных, связанного с этой последовательностью (SQLSTATE 42820); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 428FA); это значение не должно быть больше максимального значения (SQLSTATE 42815).

NO MINVALUE

Для возрастающей последовательности минимальное значение будет равно значению START WITH или 1, если значение START WITH не задано. Для убывающей последовательности минимальное значение будет равно минимальному значению типа данных, связанного с этой последовательностью. Эта опция принимается по умолчанию.

MAXVALUE или NO MAXVALUE

Задаёт максимальное значение, по достижении которого для возрастающей последовательности прекращается генерация значений или начинается следующий цикл генерации значений, а для убывающей последовательности - это значение, с которого начинается следующий цикл генерации после достижения минимального значения.

MAXVALUE *числовая-константа*

Задаёт числовое значение, представляющее собой максимальное значение. Это значение может быть любым положительным или отрицательным значением, которое можно присваивать столбцу типа данных, связанного с этой последовательностью (SQLSTATE 42820); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 428FA); это значение не должно быть меньше минимального значения (SQLSTATE 42815).

NO MAXVALUE

Для возрастающей последовательности максимальное значение будет равно максимальному значению типа данных, связанного с этой последовательностью. Для убывающей последовательности максимальное значение будет равно значению START WITH или -1, если значение START WITH не задано. Это значение по умолчанию.

CYCLE или NO CYCLE

Задаёт, нужно ли продолжать генерировать значения последовательности после достижения максимального или минимального значения последовательности. Граница последовательности достигается, когда следующее значение попадает точно на граничное значение или когда оно выходит за границу, задаваемую таким значением.

CYCLE

Задаёт, что после достижения максимального или минимального значения генерация значений должна быть продолжена. Если используется эта опция, для возрастающей последовательности после достижения ее максимального значения генерируется ее минимальное значение, а для убывающей последовательности после достижения ее минимального значения генерируется ее максимальное значение. Минимальное и максимальное значения для последовательности определяют диапазон, используемый для циклической генерации значений.

Если задан параметр CYCLE, для последовательности могут быть сгенерированы повторяющиеся значения.

NO CYCLE

Задаёт, что после достижения максимального или минимального значения генерация значений будет прекращена. Эта опция принимается по умолчанию.

CACHE или NO CACHE

Задаёт, нужно ли для ускорения доступа хранить в памяти некоторое число заранее вычисленных значений. Это опция влияет на производительность и настройку.

CREATE SEQUENCE

CACHE *целая-константа*

Задаёт максимальное число значений последовательности, которые заранее вычисляются и хранятся в памяти. Если при генерации значений последовательности эти значения вычисляются заранее и хранятся в кэше, это уменьшает затраты на операции синхронного ввода-вывода в журнал.

При возникновении системной ошибки все хранящиеся в кэше и ещё неиспользованные в принятых операторах значения последовательности будут потеряны (то есть вообще не будут использованы). Заданное в опции **CACHE** значение - это максимальное число значений последовательности, которые могут быть потеряны в случае системной ошибки.

Минимальное значение - 2 (SQLSTATE 42815). Значение по умолчанию - **CACHE** 20.

NO CACHE

Задаёт, что значения последовательности не будут вычисляться заранее. Это гарантирует, что никакие значения не будут потеряны в случае системной ошибки, перезагрузки или деактивации базы данных. Если задана эта опция, значения последовательности не хранятся в кэше. В этом случае при каждом запросе нового значения последовательности будет выполняться операция синхронного ввода-вывода в журнал.

NO ORDER или **ORDER**

Задаёт, нужно ли генерировать значения последовательности в порядке запросов.

ORDER

Задаёт, что значения последовательности должны генерироваться в порядке запросов.

NO ORDER

Задаёт, что значения последовательности не нужно генерировать в порядке запросов. Эта опция принимается по умолчанию.

Замечания по использованию

- Можно определить последовательность, содержащую одинаковые значения, то есть всегда возвращающую одно и то же значение. Для этого можно задать одинаковые значения **MINVALUE** и **MAXVALUE** или задать нулевое значение **INCREMENT**. В любом случае чтобы разрешить **NEXTVAL** генерировать несколько раз одно и то же значение, необходимо задать опцию **CYCLE**. Последовательность с одинаковыми значениями может использоваться в качестве числовой глобальной переменной. Для изменения значения, генерируемого последовательностью с одинаковыми значениями, можно использовать оператор **ALTER SEQUENCE**.
- С помощью оператора **ALTER SEQUENCE** можно вручную сделать последовательность циклической. Если явно или неявно задана опция **NO CYCLE**, последовательность можно перезапустить или расширить, используя оператор **ALTER SEQUENCE**, чтобы вызвать продолжение генерации значений после того, как было достигнуто максимальное или минимальное значение для этой последовательности.
- При кэшировании значений последовательности некоторое число ее значений может храниться в памяти для быстрого доступа. Когда прикладная программа обращается к последовательности, для которой следующее значение может быть взято из кэша, это значение будет получено быстро. Однако если прикладная программа обращается к последовательности, следующее значение которой не может быть взято из кэша, для получения этого значения может потребоваться ожидание выполнения операций ввода-вывода для всей области хранения

CREATE SEQUENCE

значений. При выборе значения опции CACHE нужно выбирать между производительностью и потребностями прикладной программы.

- Владелец новой последовательности обладает для нее привилегиями ALTER и USAGE. Владелец последовательности может предоставлять для нее только привилегию USAGE и только всем пользователям (PUBLIC).

Примера

Пример 1: Создать последовательность с именем org_seq:

```
CREATE SEQUENCE org_seq
  START WITH 1
  INCREMENT BY 1
  NO MAXVALUE
  NO CYCLE
  CACHE 24
```

ALTER SEQUENCE (новый оператор SQL)

ALTER SEQUENCE

ALTER SEQUENCE

Оператор ALTER SEQUENCE изменяет атрибуты последовательности:

- Изменяя приращение между последующими значениями последовательности
- Задавая новые минимальные или максимальные значения
- Изменяя число значений последовательности, сохраняемых в кэше
- Задавая, должна ли последовательность генерироваться циклически или нет
- Задавая, должны ли значения последовательности генерироваться в порядке запросов
- Перезапуская последовательность

Область действия

Этот оператор может быть встроен в прикладную программу или выполняться как динамический оператор SQL. Это выполняемый оператор, который можно подготовить динамически. Но если применяется опция связывания DYNAMICRULES BIND, этот оператор нельзя подготовить динамически (SQLSTATE 42509).

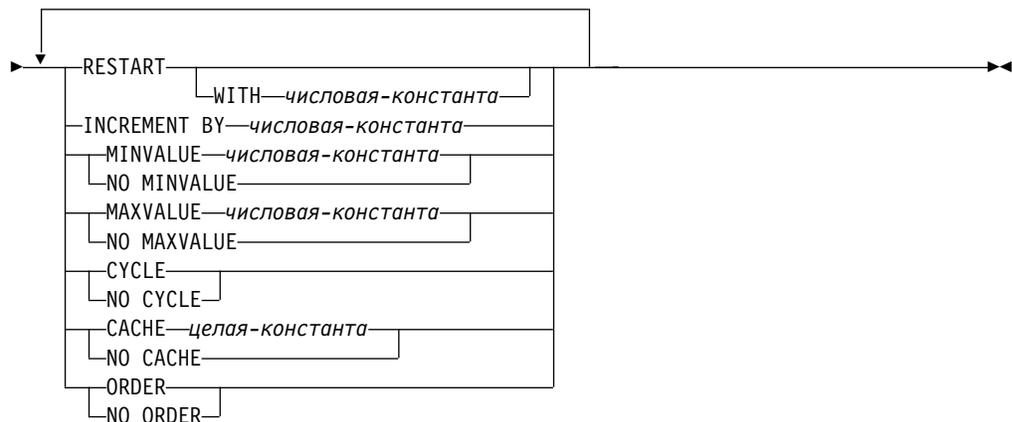
Авторизация

Привилегии ID авторизации этого оператора, должны включать, как минимум, одну из следующих привилегий:

- Привилегии определившего эту последовательность
- Привилегия ALTERIN для неявно или явно заданной схемы
- Полномочия SYSADM или DBADM

Синтаксис команды

►► ALTER SEQUENCE—*имя-последовательности*—————►



Параметры команды

имя-последовательности

Задаёт конкретную последовательность. Комбинация этого имени и неявно или явно заданного имени схемы должна задавать существующую последовательность на текущем сервере. Если в явно или неявно заданной схеме нет последовательности с таким именем, возникает ошибка (SQLSTATE 42704).

RESTART

Перезапускает последовательность. Если *числовая-константа* не задана,

последовательность перезапускается со значения, заданного неявно или явно в качестве начального значения в операторе CREATE SEQUENCE, которым была создана эта последовательность.

WITH *числовая-константа*

Перезапускает последовательность с конкретного значения. Это значение может быть любым положительным или отрицательным значением, которое можно присваивать столбцу типа данных, связанного с этой последовательностью (SQLSTATE 42820); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 42894).

INCREMENT BY

Задаёт разность между последовательными значениями. Она может быть любым положительным или отрицательным значением, которое можно присваивать столбцу типа данных, связанного с этой последовательностью (SQLSTATE 42820), и которое не превышает значения большой целой константы (SQLSTATE 42815); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 428FA).

Если это отрицательное значение, последовательность значений будет убывающей. Если это положительное значение, последовательность значений будет возрастающей. Если это значение равно 0 или если оно выходит за диапазон, определяемый значениями MINVALUE и MAXVALUE, будет генерироваться одно и то же значение, но последовательность все равно будет считаться возрастающей последовательностью.

MINVALUE *или NO MINVALUE*

Задаёт минимальное значение, по достижении которого для убывающей последовательности прекращается генерация значений или начинается следующий цикл генерации значений; для возрастающей последовательности это значение, с которого начинается следующий цикл генерации после достижения максимального значения.

MINVALUE *числовая-константа*

Задаёт минимальное числовое значение. Оно может быть любым положительным или отрицательным значением, которое можно присваивать столбцу типа данных, связанного с этой последовательностью (SQLSTATE 42820); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 428FA); это значение не должно быть больше максимального значения (SQLSTATE 42815).

NO MINVALUE

Для возрастающей последовательности минимальное значение будет равно значению START WITH или 1, если значение START WITH не задано. Для убывающей последовательности минимальное значение будет равно минимальному значению типа данных, связанного с этой последовательностью. Эта опция принимается по умолчанию.

MAXVALUE *или NO MAXVALUE*

Задаёт максимальное значение, по достижении которого для возрастающей последовательности прекращается генерация значений или начинается следующий цикл генерации значений; для убывающей последовательности это значение, с которого начинается следующий цикл генерации после достижения минимального значения.

MAXVALUE *числовая-константа*

Задаёт максимальное числовое значение. Оно может быть любым положительным или отрицательным значением, которое можно присваивать столбцу типа данных, связанного с этой последовательностью (SQLSTATE 428FA); справа от разделителя целой и дробной части не должно быть

ALTER SEQUENCE

ненулевых знаков (SQLSTATE 428FA); это значение не должно быть меньше минимального значения (SQLSTATE 42815).

NO MAXVALUE

Для возрастающей последовательности максимальное значение будет равно максимальному значению типа данных, связанного с этой последовательностью. Для убывающей последовательности максимальное значение будет равно значению START WITH или -1, если значение START WITH не задано. Это значение по умолчанию.

CYCLE или NOCYCLE

Задаёт, нужно ли продолжать генерировать значения последовательности после достижения максимального или минимального значения последовательности. Граница последовательности достигается, когда следующее значение попадает точно на граничное значение или когда оно выходит за границу, задаваемую таким значением; в последнем случае, если разрешены циклы, следующим значением будет значение START WITH.

CYCLE

Задаёт, что после достижения максимального или минимального значения генерация значений должна быть продолжена. Если используется эта опция, для возрастающей последовательности после достижения ее максимального значения генерируется ее минимальное значение, а для убывающей последовательности после достижения ее минимального значения генерируется ее максимальное значение. Минимальное и максимальное значения для последовательности определяют диапазон, используемый для циклической генерации значений.

Если задан параметр CYCLE, для последовательности могут быть сгенерированы повторяющиеся значения.

NO CYCLE

Задаёт, что после достижения максимального или минимального значения генерация значений будет прекращена. Эта опция принимается по умолчанию.

CACHE или NO CACHE

Задаёт, нужно ли для ускорения доступа хранить в памяти некоторое число заранее вычисленных значений. Это опция влияет на производительность и настройку.

CACHE *целая-константа*

Задаёт максимальное число значений последовательности, которые заранее вычисляются и хранятся в памяти. Когда значения вычисляются заранее и хранятся в кэше, это уменьшает затраты на операции синхронного ввода-вывода в журнал при генерации значений последовательности.

При возникновении системной ошибки все хранящиеся в кэше и еще неиспользованные в принятых операторах значения последовательности будут потеряны (то есть вообще не будут использованы). Заданное в опции CACHE значение - это максимальное число значений последовательности, которые могут быть потеряны в случае системной ошибки.

Минимальное значение - 2 (SQLSTATE 42815). Значение по умолчанию - CACHE 20.

NO CACHE

Задаёт, что значения последовательности не будут вычисляться заранее. Это гарантирует, что никакие значения не будут потеряны в случае системной ошибки, перезагрузки или деактивации базы данных. Если задана эта опция,

значения последовательности не хранятся в кэше. В этом случае при каждом запросе нового значения последовательности будет выполняться операция синхронного ввода-вывода в журнал.

NO ORDER или ORDER

Задаёт, нужно ли генерировать значения последовательности в порядке запросов.

ORDER

Задаёт, что значения последовательности должны генерироваться в порядке запросов.

NO ORDER

Задаёт, что значения последовательности не нужно генерировать в порядке запросов. Эта опция принимается по умолчанию.

После перезапуска последовательности или задании опции **CYCLE**, значения последовательности могут совпадать со значениями, генерируемыми этой последовательностью ранее.

Замечания по использованию

- Оператор **ALTER SEQUENCE** влияет только на будущие значения последовательности.
- Тип данных последовательности нельзя изменить. Вместо этого отбросьте последовательность и заново создайте ее, задав для новой последовательности требуемый тип данных.
- При изменении последовательности все ее значения в кэше будут потеряны.

Примера

Пример 1: Причиной задания **RESTART** без числового значения может быть необходимость перезапустить последовательность со значения **START WITH**. В этом примере целью является генерация чисел с 1 до числа строк в таблице и затем вставка этих чисел в столбец, добавленный в таблицу с помощью временных таблиц. Другой вариант использования - получение результата с перенумерованными строками:

```
ALTER SEQUENCE org_seq
RESTART
SELECT NEXTVAL for org_seq, org.*
FROM org
```

GRANT (привилегии для последовательности) (новый оператор SQL)

GRANT (привилегии для последовательности)

GRANT (привилегии для последовательности)

Эта форма оператора GRANT предоставляет привилегии для пользовательской последовательности.

Область действия

Этот оператор может быть встроен в прикладную программу или выполняться как динамический оператор SQL. Это выполняемый оператор, который можно подготовить динамически. Но если применяется опция связывания DYNAMICRULES BIND, этот оператор нельзя подготовить динамически (SQLSTATE 42509).

Авторизация

Привилегии ID авторизации этого оператора, должны включать, как минимум, одну из следующих привилегий:

- Привилегии владельца последовательности
- Полномочия SYSADM или DBADM

Синтаксис команды

```
►► GRANT USAGE ON SEQUENCE имя-последовательности TO PUBLIC ◀◀
```

Параметры команды

USAGE

Предоставляет привилегию USAGE для последовательности.

ON SEQUENCE *имя-последовательности*

Задаёт последовательность, для которой нужно предоставить привилегию USAGE. *Имя-последовательности* вместе с неявно или явно заданным именем схемы, должно однозначно задавать существующую последовательность на текущем сервере. Если в заданной схеме нет последовательности с таким именем, возникает ошибка (SQLSTATE 42704).

TO PUBLIC

Предоставить привилегию USAGE всем пользователям.

Примера

Пример 1: Предоставляет всем пользователям привилегию для последовательности с именем MYNUM

```
GRANT USAGE ON SEQUENCE MYNUM TO PUBLIC
```

SET ENCRYPTION PASSWORD (новый оператор SQL)

SET ENCRYPTION PASSWORD

Оператор SET ENCRYPTION PASSWORD задает пароль, который будет использоваться функциями шифрования и дешифрования. Этот пароль не связан с аутентификацией DB2, он используется только для шифрования данных.

Этот оператор не контролируется программой управления транзакциями.

Область действия

Этот оператор может быть включен в прикладную программу или задаваться интерактивно. Это выполняемый оператор, который можно подготовить динамически.

Авторизация

Для выполнения этого оператора не требуются полномочия.

Синтаксис команды

```

▶▶ SET ENCRYPTION PASSWORD [ ] [ ]
                                     |
                                     |----- переменная-хоста -----|
                                     |----- строковая-константа -----|
                                     |-----|
  
```

Параметры команды

ENCRYPTION PASSWORD может использоваться встроенными функциями ENCRYPT и DECRYPT для шифрования с паролем. Пароль может иметь длину от 6 до 127 байт, включительно. Все символы должны задаваться в нужном регистре, так как их преобразование в верхний регистр не производится.

переменная-хоста

Переменная типа CHAR или VARCHAR. Длина содержимого этой переменной хоста должна быть от 6 до 127 байт включительно (SQLSTATE 428FC). Для нее нельзя задать пустое значение. Все символы должны задаваться в нужном регистре, так как их преобразование в верхний регистр не производится.

строковая-константа

Константа типа символьная строка. Ее длина должна быть от 6 до 127 байт включительно (SQLSTATE 428FC).

- Исходное значение ENCRYPTION PASSWORD - пустая строка ('').
- Эта переменная-хоста или строковая-константа передается серверу баз данных с помощью обычных механизмов DB2.

Замечания по использованию

- Дополнительную информацию об использовании этого оператора смотрите в разделах “Функция ENCRYPT (новая скалярная функция)” на стр. 296 и “Функция DECRYPT_BIN и DECRYPT_CHAR (новая скалярная функция)” на стр. 298.

Примера

Пример 1: Следующий оператор задает ENCRYPTION PASSWORD.

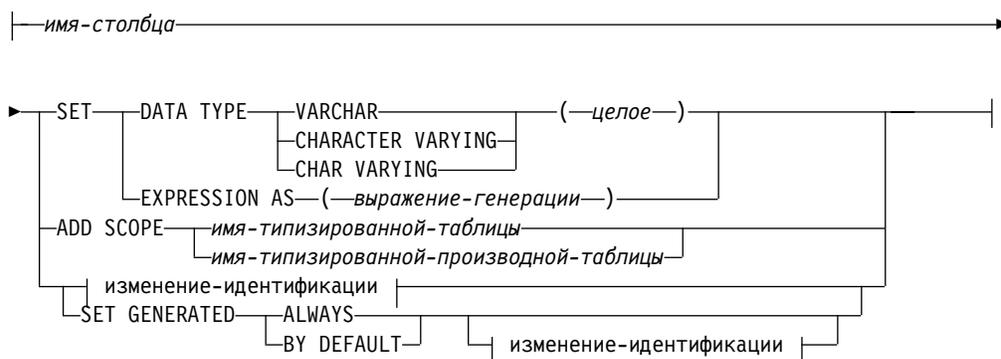
```
SET ENCRYPTION PASSWORD = 'bubbalu'
```

Изменения для оператора ALTER TABLE

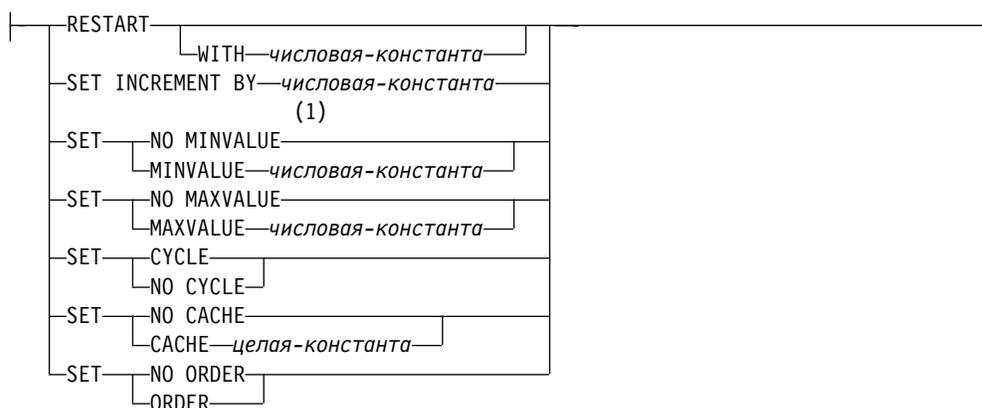
Изменения для оператора ALTER TABLE: Изменения фрагментов синтаксиса:

SET ENCRYPTION PASSWORD

изменение-столбца:



изменение-идентификации:



Примечания:

- 1 Эти параметры можно задавать без пробелов: NOMINVALUE, NOMAXVALUE, NOCYCLE, NOCACHE и NOORDER. Все эти варианты параметров, записанных в одно слово, можно задавать вместо вариантов из двух слов.

Добавьте следующие параметры:

SET GENERATED

Задаёт, должны ли значения для столбца генерироваться всегда или только когда требуется значение по умолчанию.

ALWAYS

При вставке строки в таблицу для этого столбца всегда будет генерироваться значение. Столбец должен быть уже определен как генерируемый столбец (SQLSTATE 42837).

BY DEFAULT

При вставке строки в таблицу для этого столбца будет генерироваться значение, если значение не задано. Столбец должен быть уже определен как генерируемый столбец (SQLSTATE 42837).

SET INCREMENT BY *числовая-константа*

Задаёт разность между последовательными значениями столбца идентификации. Столбец должен быть уже определен с атрибутом IDENTITY

SET ENCRYPTION PASSWORD

(SQLSTATE 42837). Это значение - любое положительное или отрицательное значение, которое можно присваивать этому столбцу (SQLSTATE 42820) и которое не превышает значения большой целой константы (SQLSTATE 42815); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 42894).

Если это отрицательное значение, последовательность значений для этого столбца идентификации будет убывающей. Если это положительное значение, последовательность значений для этого столбца идентификации будет возрастающей. Если это значение равно 0 или если оно выходит за диапазон, определяемый значениями MINVALUE и MAXVALUE, DB2 будет генерировать одно и то же значение, но последовательность все равно будет считаться возрастающей.

SET MINVALUE *числовая-константа* **или** **NO MINVALUE**

Задаёт минимальное значение, по достижении которого для убывающего столбца идентификации прекращается генерация значений или начинается следующий цикл генерации значений, а для возрастающего столбца идентификации - это значение, с которого начинается следующий цикл генерации после достижения максимального значения. Столбец должен быть уже определен с атрибутом IDENTITY (SQLSTATE 42837).

MINVALUE *числовая-константа*

Задаёт минимальное значение. Оно может быть любым положительным или отрицательным значением, которое можно присваивать этому столбцу (SQLSTATE 42820); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 42894); это значение должно быть меньше максимального значения (SQLSTATE 42815).

NO MINVALUE

Для возрастающей последовательности минимальное значение будет равно значению START WITH или 1, если значение START WITH не задано. Для убывающей последовательности минимальное значение будет равно минимальному значению типа данных этого столбца.

SET MAXVALUE *числовая-константа* **или** **NO MAXVALUE**

Задаёт максимальное значение, по достижении которого для возрастающего столбца идентификации прекращается генерация значений или начинается следующий цикл генерации значений; для убывающего столбца идентификации это значение, с которого начинается следующий цикл генерации после достижения минимального значения. Столбец должен быть уже определен с атрибутом IDENTITY (SQLSTATE 42837).

MAXVALUE *числовая-константа*

Задаёт числовую константу - максимальное значение. Это значение может быть любым положительным или отрицательным значением, которое можно присваивать этому столбцу (SQLSTATE 42820); справа от разделителя целой и дробной части не должно быть ненулевых знаков (SQLSTATE 42894); это значение должно быть больше минимального значения (SQLSTATE 42815).

NO MAXVALUE

Для возрастающей последовательности максимальное значение будет равно максимальному значению типа данных этого столбца. Для убывающей последовательности максимальное значение будет равно значению START WITH или -1, если значение START WITH не задано.

SET CYCLE **или** **NO CYCLE**

Задаёт, нужно ли продолжать генерировать значения столбца идентификации

SET ENCRYPTION PASSWORD

после генерации максимального или минимального значения. Столбец должен быть уже определен с атрибутом IDENTITY (SQLSTATE 42837).

CYCLE

Задает, что после достижения максимального или минимального значения генерация значений этого столбца должна быть продолжена. Если используется эта опция, для возрастающего столбца идентификации после достижения его максимального значения генерируется его минимальное значение, а для убывающего столбца идентификации после достижения его минимального значения генерируется его максимальное значение. Минимальное и максимальное значения для столбца идентификации определяют диапазон, используемый для циклической генерации значений.

Если задан параметр CYCLE, для столбца идентификации могут быть сгенерированы повторяющиеся значения. Хотя это не обязательное требование, но если нужны уникальные значения, для обеспечения уникальности значений используйте определенный для этого столбца идентификации индекс уникальности с одним столбцом. Если для столбца идентификации существует такой индекс и генерируется неуникальное значение, возникает ошибка (SQLSTATE 23505).

NO CYCLE

Задает, что после достижения максимального или минимального значения генерация значений столбца идентификации будет прекращена.

SET CACHE *целая-константа* или NO CACHE

Задает, нужно ли для ускорения доступа хранить в памяти некоторое число заранее вычисленных значений. Это опция влияет на производительность и настройку. Столбец должен быть уже определен с атрибутом IDENTITY (SQLSTATE 42837).

CACHE *целая-константа*

Задает максимальное число значений последовательности идентификации, которые заранее вычисляются и хранятся в памяти. Если при генерации значений столбца идентификации эти значения вычисляются заранее и хранятся в кэше, это уменьшает затраты на операции синхронного ввода-вывода в журнал.

Если для столбца идентификации требуется новое значение, но в кэше нет неиспользованных значений, для получения этого значения может потребоваться ожидание выполнения операций ввода-вывода в журнал. Однако если для столбца идентификации требуется новое значение и в кэше есть неиспользованное значение, это значение будет получено быстрее, так как не будут выполняться операции ввода-вывода в журнал.

При остановке менеджера баз данных (например, при деактивации базы данных, при возникновении системной ошибки или завершении работы системы) все хранящиеся в кэше и еще неиспользованные в принятых операторах значения последовательности будут потеряны (то есть вообще не будут использованы). Заданное в опции CACHE значение - это максимальное число значений для столбца идентификации, которые могут быть потеряны в случае системной ошибки.

Минимальное значение - 2 (SQLSTATE 42615).

NO CACHE

Задает, что значения для столбца идентификации не будут вычисляться заранее.

Если задана эта опция, значения столбца идентификации не хранятся в кэше. В этом случае при каждом запросе нового значения столбца идентификации будет выполняться операция синхронного ввода-вывода в журнал.

SET ORDER или NO ORDER

Задаёт, нужно ли генерировать значения столбца идентификации в порядке запросов. Столбец должен быть уже определен с атрибутом IDENTITY (SQLSTATE 42837).

ORDER

Задаёт, что значения столбца идентификации должны генерироваться в порядке запросов.

NO ORDER

Задаёт, что значения столбца идентификации не нужно генерировать в порядке запросов.

RESTART или RESTART WITH *числовая-константа*

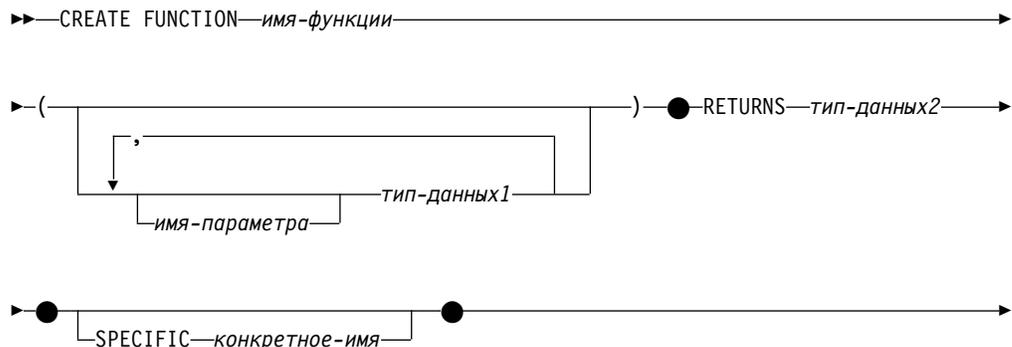
Перезадаёт последовательность, связанную со столбцом идентификации. Если не задано условие WITH *числовая-константа*, последовательность значений для столбца идентификации перезапускается со значения, которое было явно или неявно задано в качестве начального значения при создании этого столбца идентификации. *Числовая-константа* - это точная числовая константа, она может принимать любое положительное или отрицательное значение, которое можно присваивать этому столбцу (SQLSTATE 42820); справа от десятичной точки не должно быть ненулевых знаков (SQLSTATE 42894). Столбец должен быть уже определен с атрибутом IDENTITY (SQLSTATE 42837). Значение *числовой-константы* будет использовано в качестве следующего значения столбца идентификации.

Изменения раздела Compound SQL (Embedded) (Составные операторы SQL (встроенные))

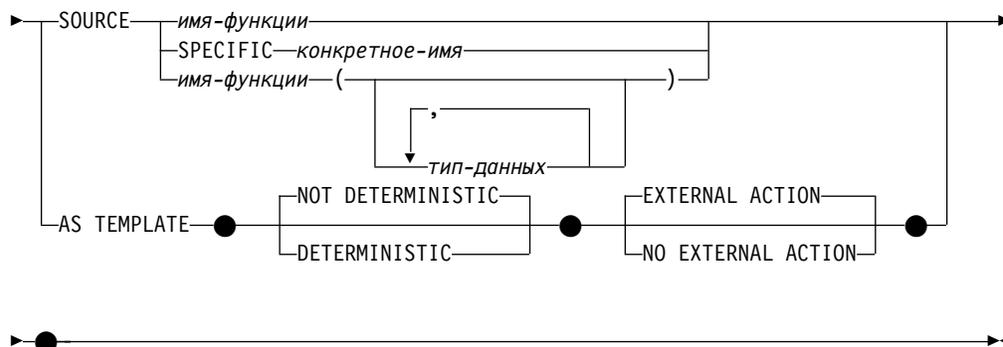
В составном операторе SQL ATOMIC не разрешен подготовленный оператор COMMIT.

Изменения описания оператора CREATE FUNCTION (с источником или шаблоном)

Синтаксическая диаграмма изменена на следующую



SET ENCRYPTION PASSWORD



Добавьте следующий текст в раздел "Description" (Описание):

DETERMINISTIC или **NOT DETERMINISTIC**

Это необязательное условие указывает, будет ли функция всегда возвращать одинаковый результат для данных значений аргументов (**DETERMINISTIC**) или же на результат функции влияют какие-то значения состояния (**NOT DETERMINISTIC**). Функция типа **DETERMINISTIC** при последовательных вызовах ее с одними и теми же входными данными должна всегда возвращать одну и ту же таблицу. Если задано условие **NOT DETERMINISTIC**, не будут выполняться оптимизации, использующие преимущества того, что для одинаковых входных данных всегда получается один и тот же результат.

Условие **NOT DETERMINISTIC** должно быть задано (явно или неявно), если функция обращается к специальному регистру или вызывает другую недетерминированную функцию (SQLSTATE 428C2).

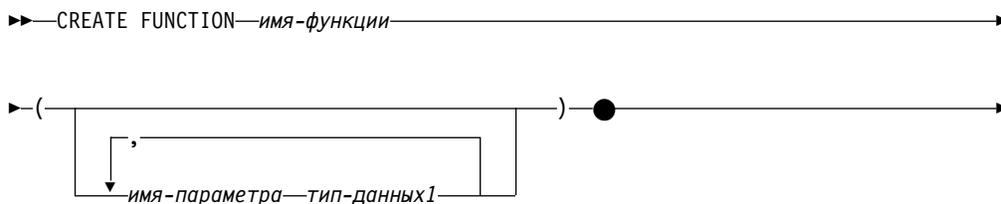
NO EXTERNAL ACTION или **EXTERNAL ACTION**

Это необязательное условие указывает, выполняет ли эта функция действия, изменяющие состояние объектов, которыми не управляет менеджер баз данных. Если задано условие **NO EXTERNAL ACTION**, система может использовать определенные оптимизации, исходя из того, что эта функция не влияет на внешние объекты.

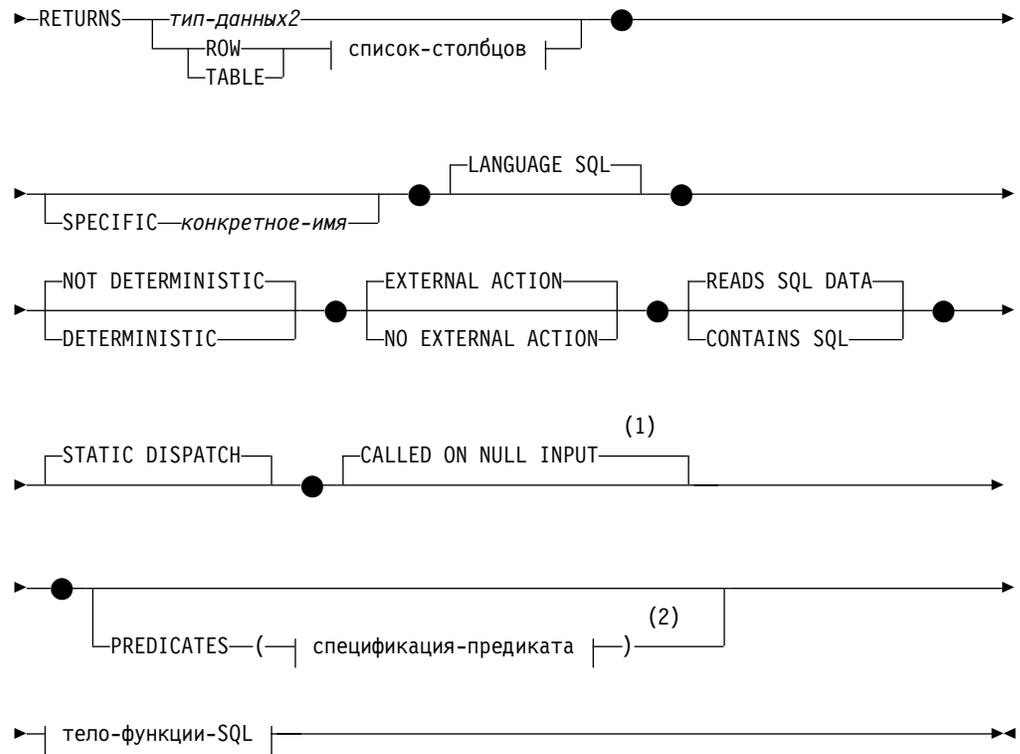
Условие **EXTERNAL ACTION** должно быть задано (явно или неявно), если функция вызывает другую функцию с внешним действием (SQLSTATE 428C2).

Изменения описания оператора **CREATE FUNCTION** (скалярная SQL, табличная или строковая)

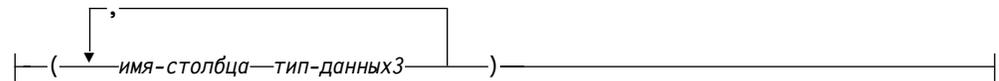
Синтаксическая диаграмма изменена на:



SET ENCRYPTION PASSWORD



список-столбцов:



тело-функции-SQL:



Примечания:

- 1 Вместо условия CALLED ON NULL INPUT можно задать NULL CALL
- 2 Допустимо, только если в RETURNS задан скалярный результат (тип-данных2)

Измените следующие параметры:

LANGUAGE SQL

Указывает, что функция написана на языке SQL.

Этот блок параметров заменяет блок параметров "RETURN expression, NULL, WITH common-table-expression, fullselect".

тело-функции-SQL

Задаёт тело функции. В теле-функции-SQL можно использовать имена параметров. Чтобы избежать неоднозначности, можно добавить к именам параметров в качестве спецификатора имя функции.

SET ENCRYPTION PASSWORD

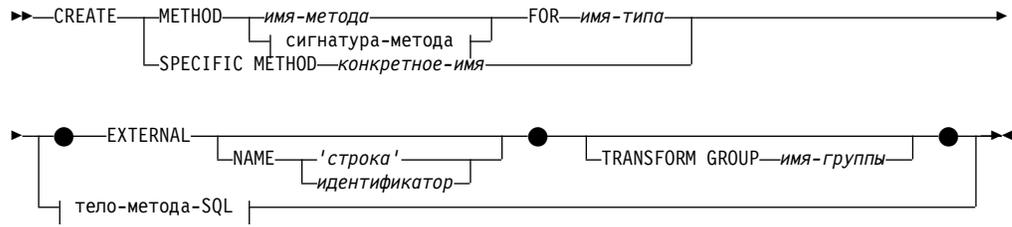
Если тело-функции-SQL представляет собой динамический составной оператор, оно должно содержать по крайней мере один оператор RETURN, который должен быть выполнен при вызове функции (SQLSTATE 42632). Если это табличная функция или строковая функция, она может содержать только один оператор RETURN, который должен быть последним оператором динамического составного оператора (SQLSTATE 429BD).

Дополнительную информацию смотрите в разделах “Составной оператор (динамический)” на стр. 341 и “оператор RETURN” на стр. 348.

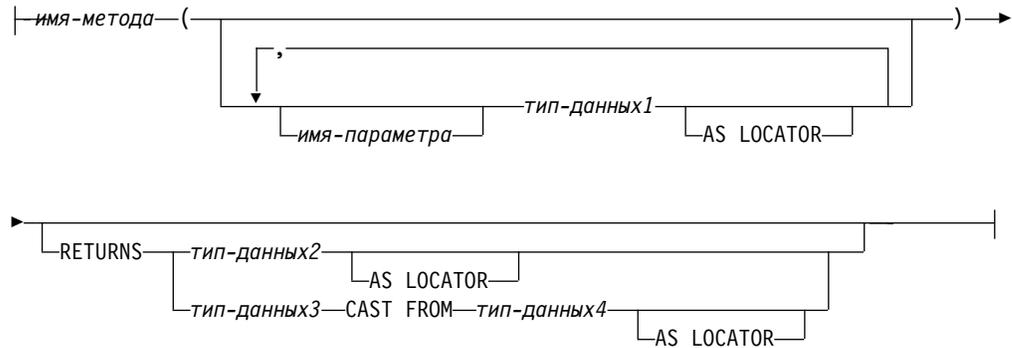
Изменения оператора CREATE METHOD

Синтаксическая диаграмма изменена на:

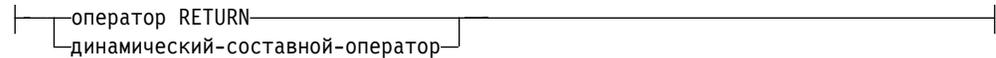
Синтаксис команды



сигнатура-метода:



тело-метода-SQL:



Раздел "RETURN scalar-expression or NULL" заменен на следующий:

тело-метода-SQL

Тело-метода-SQL определяет, как реализован метод, если в CREATE TYPE для него задано LANGUAGE SQL.

Тело-метода-SQL должно соответствовать следующим частям объявления метода:

- DETERMINISTIC или NOT DETERMINISTIC (SQLSTATE 428C2)
- EXTERNAL ACTION или NO EXTERNAL ACTION (SQLSTATE 428C2)
- CONTAINS SQL или READS SQL DATA (SQLSTATE 42985)

В теле-метода-SQL можно использовать имена параметров. Объект метода передается реализации метода как неявный первый параметр с именем SELF.

Дополнительную информацию смотрите в разделах "Составной оператор (динамический)" на стр. 341 и "оператор RETURN" на стр. 348.

Изменения оператора CREATE TRIGGER

SET ENCRYPTION PASSWORD

Синтаксис команды

► CREATE TRIGGER *имя-триггера* NO CASCADE BEFORE
AFTER

► INSERT
DELETE
UPDATE ON *имя-таблицы*
OF *имя-столбца*

► REFERENCING (1) (2)
OLD AS *внутриоператорное-имя*
NEW AS *внутриоператорное-имя*
OLD_TABLE AS *идентификатор*
NEW_TABLE AS *идентификатор*

► FOR EACH ROW (3) MODE DB2SQL | действие-триггера |
FOR EACH STATEMENT

действие-триггера:

WHEN (*критерий-поиска*) оператор-процедуры-SQL

Примечания:

- 1 Каждое из условий OLD и NEW можно задавать только один раз.
- 2 Каждое из условий OLD_TABLE и NEW_TABLE можно задавать только один раз и только для триггеров AFTER.
- 3 Условие FOR EACH STATEMENT нельзя задавать для триггеров BEFORE.

Описание "triggered-action" ("действия-триггера") замените на следующее:

действие-триггера

Задаёт действие, выполняемое при активации триггера. Это действие состоит из *оператора-процедуры-SQL* и необязательного условия для выполнения этого оператора.

WHEN (*критерий-поиска*)

Задаёт критерий, который может принимать значения true, false или unknown. *критерий-поиска* позволяет определить, нужно ли выполнять действие триггера в конкретном случае.

Действие триггера выполняется только в том случае, когда заданный критерий поиска имеет значение true. Если условие WHEN не задано, связанный с ним оператор-процедуры-SQL будет выполняться во всех случаях.

оператор-процедуры-SQL

Оператор-процедуры-SQL может содержать динамический составной оператор или любой из управляющих операторов SQL, перечисленных в разделе “Составной оператор (динамический)” на стр. 341.

Если это триггер BEFORE, оператор-процедуры-SQL может также содержать полную выборку или оператор SET переменная (SQLSTATE 42987).

Если это триггер AFTER, оператор-процедуры-SQL может также содержать один из следующих операторов (SQLSTATE 42987):

- оператор SQL INSERT
- оператор SQL UPDATE с поиском
- оператор SQL DELETE с поиском
- оператор SET переменная
- полную выборку ¹

В операторе-процедуры-SQL нельзя использовать неопределенные переходные переменные (SQLSTATE 42703) или объявленные временные таблицы (SQLSTATE 42995).

В операторе-процедуры-SQL триггера BEFORE нельзя использовать таблицу сводки, определенную с условием REFRESH IMMEDIATE (SQLSTATE 42997).

В операторе-процедуры-SQL триггера BEFORE нельзя использовать генерируемые столбцы (кроме столбца идентификации) в новой переходной переменной (SQLSTATE 42989).

Раздел Notes (Примечания) изменен следующим образом:

- Результат полной выборки, заданной в операторе-процедуры-SQL не доступен внутри и вне триггера.
- **Неработоспособные триггеры:** *Неработоспособный триггер* - это триггер, который более недоступен и поэтому никогда не активируется. Триггеры становятся неработоспособными, если:
 - Создатель триггера был лишен привилегии, необходимой для выполнения триггера.
 - Отброшен объект (такой как таблица, производная таблица или алиас), от которого зависит действие триггера.
 - Производная таблица, от которой зависит действие триггера, стала неработоспособной.
 - Отброшен алиас, представляющий рабочую таблицу триггера.

Практически неработоспособный триггер - это триггер, определение которого было отброшено в результате последовательного применения правил для операторов DROP или REVOKE. Например, если отброшена производная таблица, все триггеры, в определении оператора-процедуры-SQL которых используется эта производная таблица, становятся неработоспособными.

1. Перед полной выборкой может задаваться общее-табличное-выражение.

SET ENCRYPTION PASSWORD

Если триггер стал неработоспособными, все пакеты с операторами, выполняющими действия, которые активировали этот триггер, отмечаются как дефектные. При повторном связывании пакета (явном или неявном) **неработоспособный триггер полностью игнорируется**. Аналогично, прикладные программы с динамическими операторами SQL, выполняющие действия, которые активировали этот триггер, также будут полностью игнорировать все неработоспособные триггеры.

Имя такого триггера все еще можно задавать в операторах DROP TRIGGER и COMMENT ON TRIGGER.

Неработоспособный триггер можно создать заново с помощью оператора CREATE TRIGGER, используя текст определения этого неработоспособного триггера. Текст этого определения триггера хранится в столбце TEXT таблицы SYSCAT.TRIGGERS. Обратите внимание на то, что не нужно явно отбрасывать неработоспособный триггер, чтобы создать его заново. Если введен оператор CREATE TRIGGER с тем же *именем-триггера*, что и неработоспособный триггер, этот неработоспособный триггер будет заменен и будет выдано предупреждение (SQLSTATE 01595).

Неработоспособные триггеры обозначаются символом X в столбце VALID производной таблицы каталога SYSCAT.TRIGGERS.

- **Ошибки при выполнении триггеров:** Если при выполнении оператора SQL триггера возникли ошибки, которые не считаются серьезными, возвращается SQLSTATE 09000. Если возникла серьезная ошибка, возвращается SQLSTATE этой ошибки. Для ошибок, не считающихся серьезными, поле SQLERRMC области SQLCA будет содержать имя триггера, SQLCODE, SQLSTATE и все уместившиеся в нем элементы сообщения об ошибке.

Оператор-процедуры-SQL может содержать оператор SIGNAL SQLSTATE или функцию RAISE_ERROR. В обоих этих случаях возвращается SQLSTATE, заданный в операторе SIGNAL SQLSTATE или условии RAISE_ERROR.

Изменения для оператора DROP

Добавьте следующую опцию:

►►—SEQUENCE—*имя-последовательности*—RESTRICT—◄◄

Добавьте следующие параметры:

SEQUENCE *имя-последовательности* **RESTRICT**

Указывает конкретную последовательность, которую нужно отбросить.

Имя-последовательности вместе с неявно или явно заданным именем схемы должно задавать существующую последовательность на текущем сервере. Если в явно или неявно заданной схеме нет последовательности с таким именем, возникает ошибка (SQLSTATE 42704).

Ключевое слово RESTRICT запрещает отбрасывать последовательность, если она используется в определении какого-либо столбца таблицы (для столбца IDENTITY).

Примечание:

- Созданные системой последовательности для столбцов IDENTITY нельзя отбросить с помощью команды DROP последовательность.

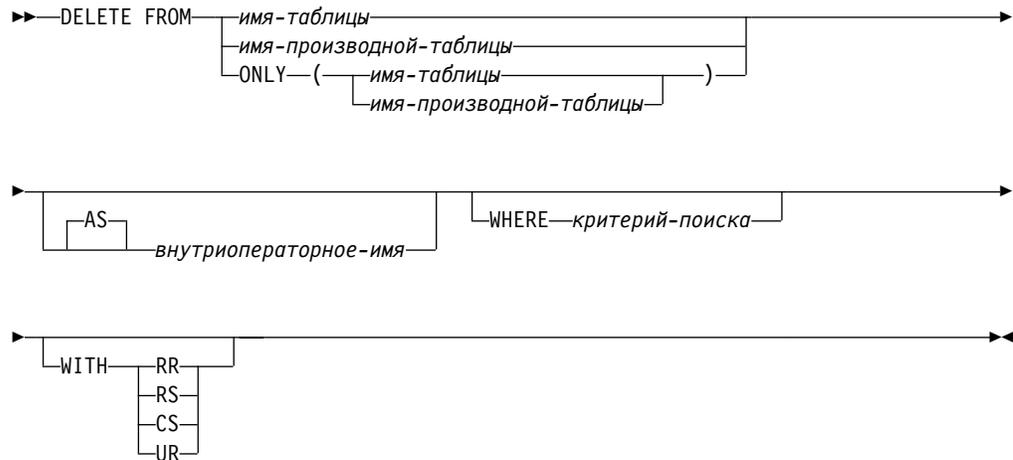
- При отбрасывании последовательности отбрасываются также все привилегии для этой последовательности.

Таблицу, в которой показаны зависимости объектов друг от друга (Таблица 27), нужно изменить так:

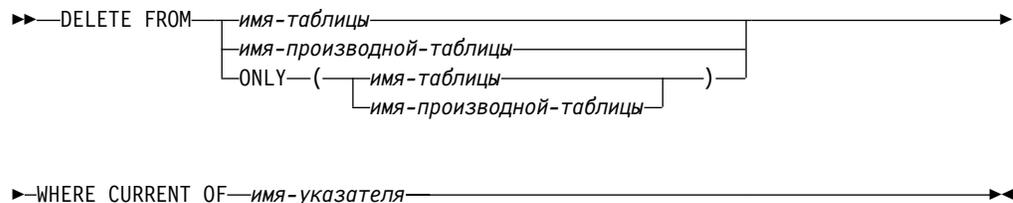
Новая строка: DROP SEQUENCE. Элемент таблицы на пересечении этой новой строки "DROP SEQUENCE" и столбца "PACKAGE" будет содержать "A". Остальные элементы в этой новой строке будут содержать "-".

Изменения для оператора DELETE

Синтаксическая диаграмма оператора DELETE с поиском изменена на следующую:



DELETE с указанием позиции:



Добавьте следующий текст в раздел описания:

WITH

Задаёт уровень изоляции, используемый при поиске удаляемых строк.

RR

Множественное чтение

RS

Стабильность чтения

CS

Стабильность на уровне указателя

SET ENCRYPTION PASSWORD

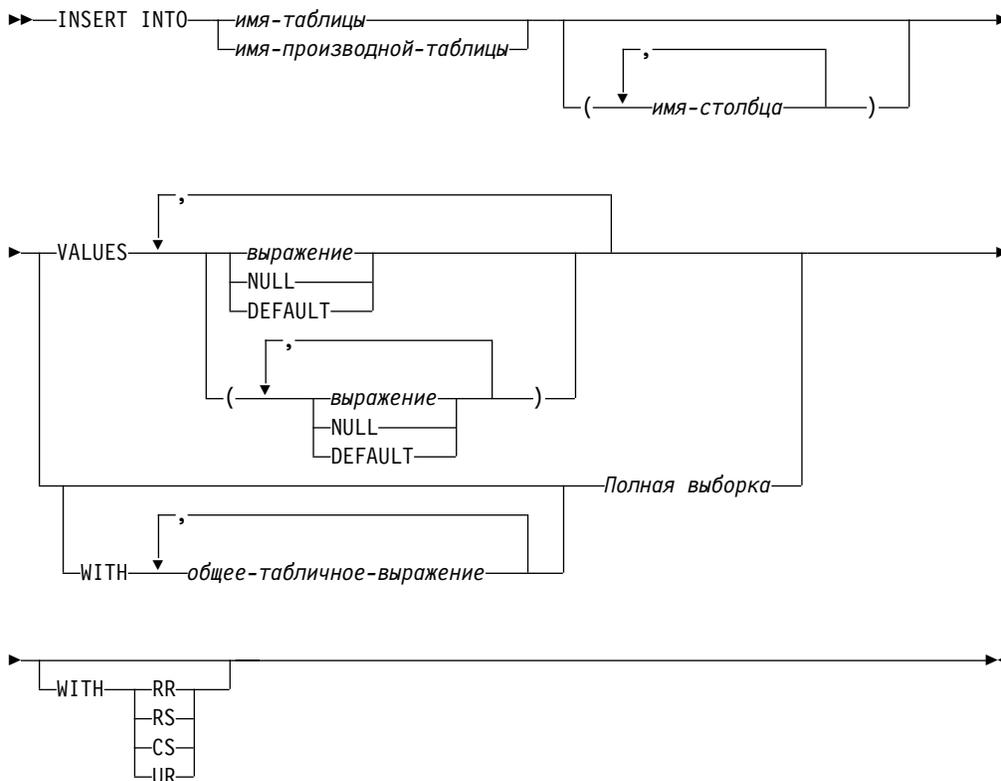
UR

Чтение неприятого

По умолчанию для оператора используется уровень изоляции пакета, в котором был связан этот оператор.

Изменения для оператора INSERT

Синтаксическая диаграмма изменена на:



Добавьте следующий текст в раздел описания:

WITH

Задает уровень изоляции, используемый при выполнении полной выборки.

RR

Множественное чтение

RS

Стабильность чтения

CS

Стабильность на уровне указателя

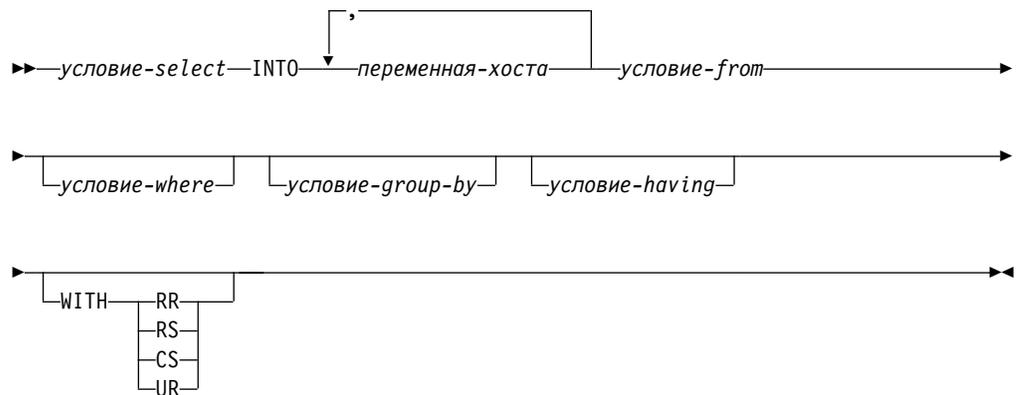
UR

Чтение неприятого

По умолчанию для оператора используется уровень изоляции пакета, в котором был связан этот оператор.

Изменения для оператора SELECT INTO

Синтаксическая диаграмма изменена на:



Добавьте следующий текст в раздел описания:

WITH

Задаёт уровень изоляции, используемый при выполнении оператора SELECT INTO.

RR

Многократное чтение

RS

Стабильность чтения

CS

Стабильность на уровне указателя

UR

Чтение неприятого

По умолчанию для оператора используется уровень изоляции пакета, в котором был связан этот оператор.

Изменения для оператора SET переходная-переменная

Этот раздел изменен следующим образом:

SET переменная

SET переменная

Оператор SET переменная присваивает значения локальным переменным или новым переходным переменным. Он контролируется программой управления транзакциями.

Область действия

Это оператор можно использовать только как оператор SQL в динамических составных операторах, триггерах, функциях SQL или методах SQL.

Авторизация

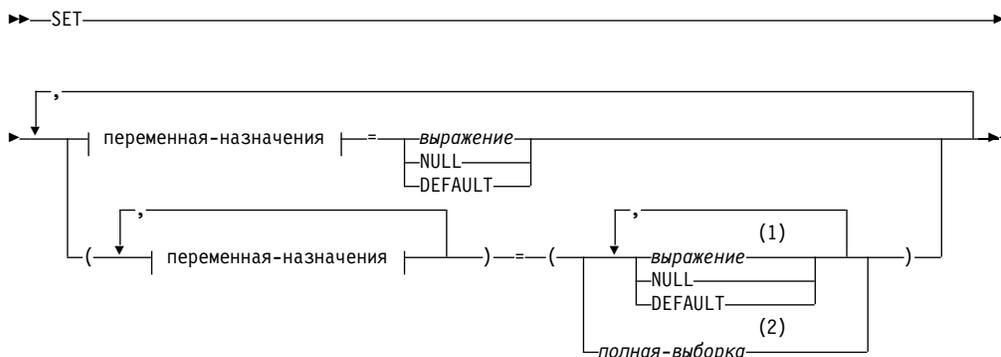
Чтобы можно было использовать переходные переменные, привилегии ID авторизации создателя триггера, должны включать, как минимум, одну из следующих привилегий:

- Привилегии UPDATE для столбцов, указанных в левой части присваивания, и SELECT для всех столбцов, указанных в правой части присваивания.
- Привилегию CONTROL для таблицы (рабочей таблицы триггера)
- Полномочия SYSADM или DBADM.

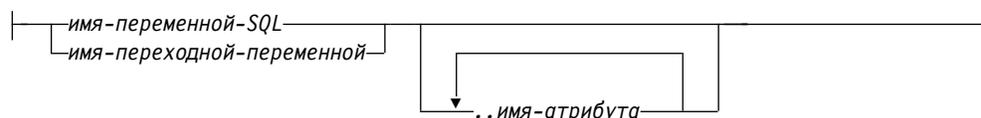
Чтобы выполнять этот оператор с *полной-выборкой-строк* в правой части присваивания, привилегии ID авторизации создателя триггера или владельца динамического составного оператора, должны также включать по крайней мере одну из следующих привилегий для каждой используемой в операторе таблицы или производной таблицы:

- Привилегия SELECT
- Привилегия CONTROL
- Полномочия SYSADM или DBADM.

Синтаксис команды



переменная-назначения:



Примечания:

- 1 Число выражений и ключевых слов NULL и DEFAULT должно совпадать с числом *переменных-назначения*.

- 2 Число столбцов в списке столбцов выборки должно совпадать с числом переменных-назначения.

Параметры команды

переменная-назначения

Задаёт переменную назначения для присваивания. *Переменную-назначения*, представляющая одну и ту же переменную, нельзя задаваться несколько раз (SQLSTATE 42701).

имя-переменной-SQL

Задаёт переменную SQL - назначение присваивания. Переменные SQL должны быть объявлены до их использования. Переменные SQL могут быть определены в динамическом составном операторе.

имя-переходной-переменной

Задаёт столбец, который нужно изменить в переходной строке. *Имя-переходной-переменной* должно задавать столбец в рабочей таблице триггера; в качестве необязательного спецификатора можно задать внутриоператорное имя, указывающее на новое значение (SQLSTATE 42703).

..имя-атрибута

Задаёт атрибут структурированного типа, для которого задается значение (указанный в *присваивании атрибутов*). Заданное *имя-переменной-SQL* или *имя-переходной-переменной* должно быть определено с пользовательским структурированным типом (SQLSTATE 428DP). *Имя-атрибута* должно быть атрибутом этого структурированного типа (SQLSTATE 42703).

Присваивание, для которого не задано условие *..имя-атрибута*, называют обычным присваиванием.

выражение

Задаёт новое значение *переменной-назначения*. Это может быть любое выражение, описанное в главе 2 справочника SQL Reference. Это выражение не может содержать функцию столбца, за исключением случая, когда оно входит в скалярную полную выборку (SQLSTATE 42903). В контексте оператора CREATE TRIGGER *выражение* может содержать ссылки на переходные переменные OLD и NEW, при этом для задания конкретной переходной переменной в качестве спецификатора должно быть задано *внутриоператорное имя* (SQLSTATE 42702).

NULL

Задаёт пустое значение; может использоваться только для столбцов, допускающих пустые значения (SQLSTATE 23502). Ключевое слово NULL нельзя задать в присваивании атрибута (SQLSTATE 429B9), если оно не может быть преобразовано в тип данных этого атрибута.

DEFAULT

Задаёт, что должно использоваться значение по умолчанию.

Если *переменная-назначения* - это столбец, вставляемое значение зависит от того, как определен этот столбец в таблице.

- Если в определении столбца задано условие WITH DEFAULT, в качестве этого значения используется значение по умолчанию, заданное для этого столбца.
- Если в определении столбца задано условие IDENTITY, это значение генерируется менеджером баз данных.
- Если в определении столбца не заданы условия WITH DEFAULT, IDENTITY или NOT NULL, это значение будет равно NULL.

SET переменная

- Если в определении столбца задано условие NOT NULL, но не задано условие IDENTITY или условие WITH DEFAULT или задано условие DEFAULT NULL, для этого столбца нельзя задавать ключевое слово DEFAULT (SQLSTATE 23502).

Если *переменная-назначения* - это переменная SQL, то вставляемое значение - это значение по умолчанию, заданное или подразумеваемое в объявлении этой переменной.

полная-выборка-строки

Полная выборка, возвращающая одну строку с числом столбцов, соответствующим числу переменных-назначения, заданных для присваивания. Значения присваиваются каждой соответствующей переменной-назначения. Если результат полной-выборки-строки не содержит строки, присваиваются пустые значения. В контексте оператора CREATE TRIGGER *полная-выборка-строки* может содержать ссылки на переходные переменные OLD и NEW, при этом для задания конкретной переходной переменной в качестве спецификатора должно быть задано *внутриоператорное имя* (SQLSTATE 42702). Если результат содержит несколько строк, возникает ошибка (SQLSTATE 21000).

- Число значений для присваивания, задаваемых выражениями, ключевыми словами NULL и DEFAULT или *полной-выборкой-строки*, должно совпадать с числом переменных-назначения, заданных для присваивания (SQLSTATE 42802).
- В одном операторе SET переходная переменная нельзя присваивать значения переменной SQL и переходной переменной (SQLSTATE 42997).
- Значения присваиваются переменным-назначения в соответствии с правилами, описанными в главе 2 справочника SQL Reference.

Если оператор используется в триггере BEFORE UPDATE и переменная реестра DB2_UPDATE_PART_KEY=OFF, *переходная-переменная*, заданная в качестве *переменной-назначения*, не может быть столбцом ключа разделения (SQLSTATE 42997).

Замечания по использованию

- Если оператор содержит несколько присваиваний, все *выражения* и *полные-выборки-строки* вычисляются перед выполнением присваиваний. Поэтому ссылки на переменные-назначения в выражении или полной выборке строки всегда дают значение переменной-назначения, которое она имела до выполнения всех присваиваний в отдельном операторе SET.
- При обновлении столбца идентификации, который определен с особым типом данных, все вычисления производятся в исходном типе данных и затем результат преобразуется в этот особый тип перед фактическим присваиванием значения столбцу.²
- Чтобы система DB2 генерировала значение в операторе SET для столбца идентификации, используйте ключевое слово DEFAULT:

```
SET NEW.EMPNO = DEFAULT
```

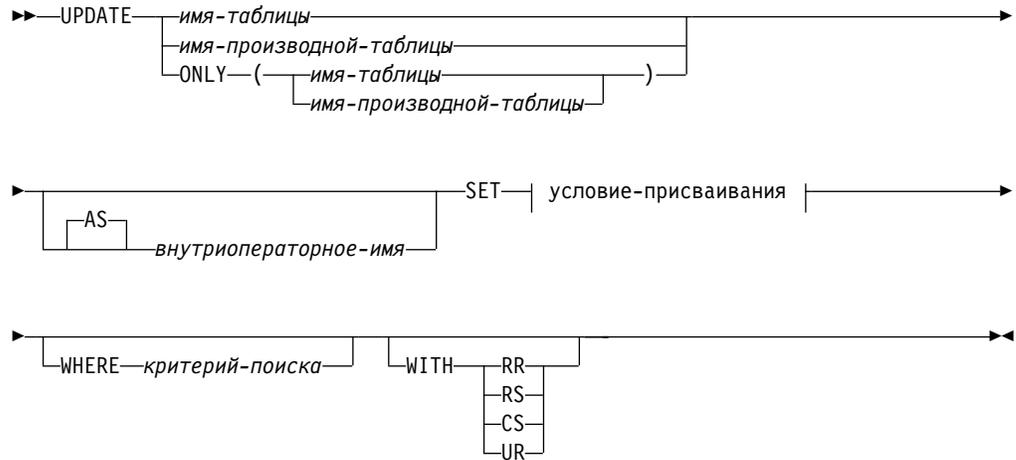
В этом примере NEW.EMPNO определен как столбец идентификации; используемое для его обновления значение генерируется DB2.

Примеры для этого оператора остались без изменений.

2. Перед вычислениями не выполняется преобразование предыдущего значения в исходный тип данных.

Изменения для оператора UPDATE

Синтаксическая диаграмма оператора UPDATE с поиском изменена:



Добавьте следующий текст в раздел описания:

WITH

Задаёт уровень изоляции, используемый при выполнении оператора UPDATE.

RR

Множественное чтение

RS

Стабильность чтения

CS

Стабильность на уровне указателя

UR

Чтение неприятого

По умолчанию для оператора используется уровень изоляции пакета, в котором был связан этот оператор.

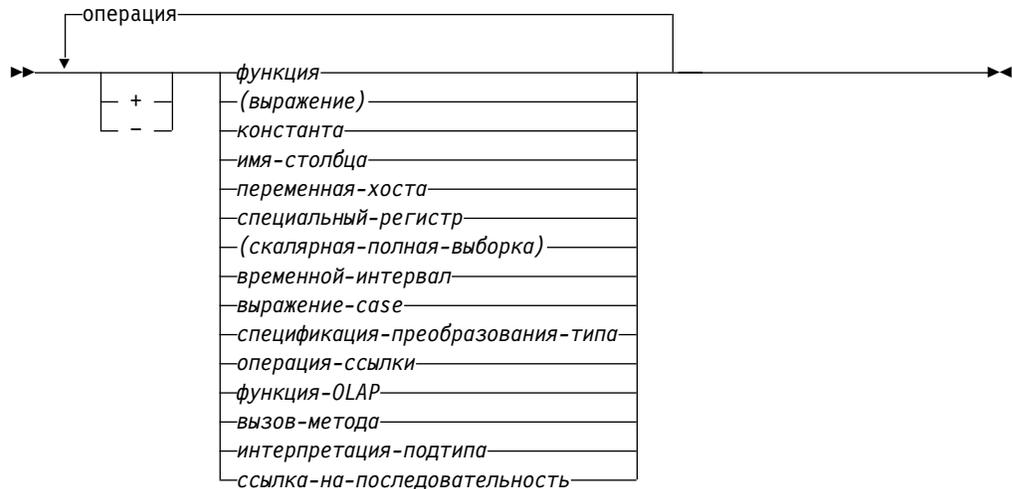
Глава 3. Language Elements (Элементы языка)

Expressions (Выражения)

Синтаксическая диаграмма

Синтаксическая диаграмма была изменена на:

SET переменная



операция:



Примечания:

1 В качестве синонима CONCAT можно использовать ||.

Справочник по последовательностям

В конце раздела Expressions (Выражения) после "Subtype Treatment" (Использование подтипа) надо добавить следующую информацию.

ссылка-на-последовательность:



выражение-nextval:



выражение-prevval:



NEXTVAL FOR имя-последовательности

Выражение NEXTVAL возвращает следующее значение для последовательности, задаваемой именем-последовательности.

PREVVAL FOR *имя-последовательности*

Выражение PREVVAL возвращает самое последнее значение для заданной последовательности, сгенерированное для предыдущего оператора в текущем сеансе. К этому значению можно повторно обращаться с помощью выражений PREVVAL, задавая имя последовательности. В одном операторе может быть несколько экземпляров выражений PREVVAL с тем же именем последовательности; все они вернут одно и то же значение.

Выражение PREVVAL можно использовать, только если выражение NEXTVAL с тем же именем последовательности уже было использовано в текущем сеансе пользователя (в текущей или одной из предыдущих транзакций) (SQLSTATE 51035).

Примечание:

- Когда имя последовательности задано в выражении NEXTVAL, для нее генерируется новое значение. Однако если в запросе есть несколько экземпляров выражения NEXTVAL с тем же именем последовательности, счетчик последовательности увеличивается только один раз для каждой строки результата.
- К самому последнему сгенерированному значению последовательности можно повторно обращаться с помощью выражения PREVVAL с именем этой последовательности. В одном операторе может быть несколько экземпляров выражений PREVVAL с тем же именем последовательности.
- Одно и то же значение последовательности может использоваться в качестве уникального ключа в двух различных таблицах - получите значение последовательности для первой строки с помощью выражения NEXTVAL (при этом будет сгенерировано новое значение последовательности) и используйте выражение PREVVAL для другой строки (этот экземпляр PREVVAL возвращает значение последовательности, сгенерированное предыдущим выражением NEXTVAL), как показано ниже:

```
INSERT INTO order(orderno, custno)
VALUES (NEXTVAL FOR order_seq, 123456);
```

```
INSERT INTO line_item (orderno, partno, quantity)
VALUES (PREVVAL FOR order_seq, 987654, 1);
```

- Примеры мест, где могут быть заданы выражения NEXTVAL и PREVVAL:
 - оператор-выборки или оператор SELECT INTO - внутри условия-выборки, если оператор не содержит ключевого слова DISTINCT, условия GROUP BY, условия ORDER BY, ключевого слова UNION, ключевого слова INTERSECT или ключевого слова EXCEPT
 - оператор INSERT - условие VALUES
 - оператор INSERT - внутри условия-выборки в полной выборке
 - оператор SET переменная-хоста - внутри условия-выборки в полной выборке в выражении
 - оператор UPDATE - внутри условия-выборки в полной выборке в выражении в условии SET (оператор UPDATE с поиском или с указанием позиции)
 - оператор VALUES INTO - внутри условия-выборки в полной выборке в выражении

SET переменная

- Примеры мест, где нельзя задавать выражения NEXTVAL и PREVVVAL (SQLSTATE 428F9):
 - условие JOIN полного внешнего объединения
 - значение DEFAULT для столбца в операторе CREATE TABLE или ALTER TABLE
 - определение генерируемого столбца в операторе CREATE TABLE или ALTER TABLE
 - условие проверочного ограничения
 - оператор CREATE TRIGGER
 - оператор CREATE VIEW
 - оператор CREATE METHOD
 - оператор CREATE FUNCTION.
- Кроме того, выражение NEXTVAL нельзя задавать (SQLSTATE 428F9) в:
 - выражении CASE
 - списке параметров функции сводки
 - подзапросе
 - операторе SELECT, который содержит операцию DISTINCT
 - условии JOIN объединения
 - условии GROUP BY оператора SELECT
 - операторе SELECT, который объединен с другим оператором SELECT с помощью оператора множества UNION, INTERSECT или EXCEPT
 - вложенном табличном выражении
 - списке параметров табличной функции
 - условии WHERE оператора SELECT, DELETE или UPDATE
 - условии ORDER BY
 - списке параметров оператора CALL.
- Когда для последовательности генерируется значение, это значение используется; затем, когда значение требуется в следующий раз, генерируется новое значение. Это так, даже если оператор, содержащий выражение NEXTVAL, завершается с ошибкой. Если оператор INSERT содержит выражение NEXTVAL в списке VALUES для столбца и при выполнении этого оператора в некоторый момент возникла ошибка (это может быть ошибка генерации следующего значения последовательности или ошибка, связанная со значением для другого столбца), операция вставки будет неудачной, но значение, сгенерированное для этой последовательности будет считаться использованным. В некоторых случаях повторное выполнение того же оператора INSERT может быть успешным. Например, предположим, что ошибка вызвана тем, что для столбца, для которого используется NEXTVAL, существует индекс уникальности, и сгенерированное значение последовательности уже существует в этом индексе. Возможно, что следующего сгенерированного для последовательности значения еще не будет в этом индексе и поэтому повторное выполнение операции INSERT будет успешным.
- Если при генерации значения для последовательности превышено максимальное значение для этой последовательности (или минимальное значение для убывающей последовательности) и

циклическая генерация не разрешена, возникнет ошибка (SQLSTATE 23522). В этом случае пользователь должен использовать оператор ALTER для этой последовательности, чтобы изменить ее и расширить диапазон допустимых значений или разрешить циклическую генерацию, или отбросить (DROP) эту последовательность и создать (CREATE) новую, с другим типом данных, имеющим больший диапазон значений.

Например, последовательность может быть определена с типом данных SMALLINT; со временем она исчерпает все возможные значения. Чтобы переопределить эту последовательность как INTEGER, нужно отбросить и заново создать эту последовательность с новым определением.

Примеры:: В этих примерах подразумевается, что существуют таблица с именем "order" и последовательность с именем "order_seq", созданная следующим образом:

```
CREATE SEQUENCE order_seq
  START WITH 1
  INCREMENT BY 1
  NOMAXVALUE
  NOCYCLE
  CACHE 24
```

- Некоторые примеры, как при помощи выражения NEXTVAL генерировать значения созданной выше последовательности "order_seq":

```
INSERT INTO order(orderno, custno)
  VALUES (NEXTVAL FOR order_seq, 123456);
```

или

```
UPDATE order
  SET orderno = NEXTVAL FOR order_seq
  WHERE custno = 123456;
```

или

```
VALUES NEXTVAL FOR order_seq INTO :hv_seq;
```

Глава 4. Functions (Функции)

Процедуры

GET_ROUTINE_SAR

GET_ROUTINE_SAR

►► GET_ROUTINE_SAR (—*sarblob*—, —тип—, —строка-имени-подпрограммы—) ►►

Схема - SYSFUN.

Эта процедура впервые стала доступна в FixPak 3 Версии 7.1.

Процедура GET_ROUTINE_SAR получает информацию, необходимую для установки той же подпрограммы на другом сервере баз данных, работающем в том же уровне той же операционной системы. Информация получается в виде одной строки BLOB, представляющей файл архива SQL. Вызывающий процедуру GET_ROUTINE_SAR должен обладать полномочиями DBADM.

sarblob

Выходной аргумент типа BLOB(3M), в котором возвращается содержимое файла SAR для этой подпрограммы.

тип

Входной аргумент типа CHAR(2), задающий тип подпрограммы; используется одно из следующих значений:

- **P** для процедуры.
- **SP** для конкретного имени процедуры.

строка-имени-подпрограммы

Входной аргумент типа VARCHAR(257), задающий полное имя подпрограммы. Если схема не задана, для подпрограммы по умолчанию используется схема CURRENT SCHEMA.

Примечание: *Строка-имени-подпрограммы* не должна содержать символ двойной кавычки (").

Полное имя подпрограммы используется для определения подпрограммы, информацию для которой нужно получить. Найденная подпрограмма должна быть подпрограммой SQL, иначе возникнет ошибка (SQLSTATE 428F7). Если не используется конкретное имя, может быть найдено несколько подпрограмм и возникнет ошибка (SQLSTATE 42725). В этом случае для получения информации для подпрограммы нужно использовать ее конкретное имя.

Файл SAR должен содержать файл связывания, который может быть недоступен на сервере. Если файл связывания не удалось найти и сохранить в файле SAR, возникает ошибка (SQLSTATE 55045).

PUT_ROUTINE_SAR

PUT_ROUTINE_SAR

►► PUT_ROUTINE_SAR ►►

► (—*sarblob* —, —*новый_владелец*—, —*флаг_использования_регистров*—) —►

Схема - SYSFUN.

Эта процедура впервые стала доступна в FixPak 3 Версии 7.1.

Процедура PUT_ROUTINE_SAR передает на сервер необходимый файл для создания на нем подпрограммы SQL и затем определяет эту подпрограмму. Вызывающий процедуру PUT_ROUTINE_SAR должен обладать полномочиями DBADM.

sarblob

Входной аргумент типа BLOB(3M), в котором передается содержимое файла SAR для этой подпрограммы.

новый_владелец

Входной аргумент типа VARCHAR(128), задающий имя авторизации, используемое для проверки полномочий подпрограммы. *Новый_владелец* должен обладать привилегиями, необходимыми для определяемой подпрограммы. Если *новый_владелец* не задан, используется имя владельца исходной подпрограммы.

флаг_использования_регистров

Входной аргумент типа INTEGER, указывающий, используются ли для определения подпрограммы специальные регистры CURRENT SCHEMA и CURRENT PATH. Если эти специальные регистры не используются, в качестве значений схемы по умолчанию и пути SQL используются те же значения, что и при определении исходной подпрограммы. Возможные значения аргумента *флага_использования_регистров*:

- 0** Не использовать специальные регистры текущей среды
- 1** Использовать специальные регистры CURRENT SCHEMA и CURRENT PATH.

Если задано значение 1, специальный регистр CURRENT SCHEMA используется для неспецифицированных имен объектов в определении подпрограммы (включая имя самой подпрограммы), а специальный регистр CURRENT PATH - для неспецифицированных имен подпрограмм и типов данных в определении этой подпрограммы. Если *флаг_использования_регистров* не задан, предполагается значение 0.

Для находящейся в *sarblob* информации идентификации выполняется проверка соответствия входных данных среде; если это не так, выдается ошибка (SQLSTATE 55046). Затем процедура PUT_ROUTINE_SAR использует содержимое *sarblob*, чтобы определить нужную подпрограмму на сервере.

Содержимое аргумента *sarblob* выделяется в отдельные файлы, составлявшие файл архива SQL. Совместно используемая библиотека и файлы связывания записываются в файлы во временном каталоге. Затем среда настраивается таким образом, чтобы выполняющийся оператор определения подпрограммы знал, что компиляция и связывание не требуются и что доступны положения совместно используемой библиотеки и файлов связывания. Затем содержимое файла DDL используется для динамического выполнения оператора определения подпрограммы.

Примечание: В данной схеме можно одновременно устанавливать только одну процедуру.

процедуры

При выполнении этого оператора могут возникнуть ошибки, как и при выполнении оператора определения подпрограммы через другие интерфейсы. В процессе определения подпрограммы отмечается наличие совместно используемой библиотеки и файлов связывания и пропускаются шаги прекомпиляции, компиляции и компоновки. Файлы связывания используются при выполнении связывания и содержимое обоих файлов копируется в обычный каталог для подпрограмм SQL.

Примечание: При возникновении ошибок выполнения операций GET ROUTINE или PUT ROUTINE (или соответствующих им процедур) всегда возвращается код ошибки (SQLSTATE 38000) и текст диагностики, содержащий информацию о причине ошибки. Например, если заданное для GET ROUTINE имя процедуры не соответствует ни одной из процедур SQL, возвращается текст диагностики "100, 02000", где "100" и "02000" - соответственно SQLCODE и SQLSTATE, чем указывается причина ошибки. SQLCODE и SQLSTATE в этом примере указывают, что для заданного имени процедуры не найдена строка в таблицах каталога.

Скалярные функции

IDENTITY_VAL_LOCAL

►—IDENTITY_VAL_LOCAL—(—)—————►

Схема - SYSIBM.

Эта процедура впервые стала доступна в FixPak 3 Версии 7.1.

Функция IDENTITY_VAL_LOCAL - это недетерминированная функция, возвращающая самое последнее значение, присвоенное столбцу идентификации в результате выполнения оператора вставки одной строки (INSERT) с условием VALUES. У этой функции нет входных параметров.

Тип данных результата - DECIMAL(31,0), независимо от типа данных соответствующего столбца идентификации.

Возвращаемое этой функцией значение - это значение, присвоенное столбцу идентификации таблицы, указанной в самом последнем операторе вставки одной строки INSERT. В этом операторе INSERT должно использоваться условие VALUES для таблицы, содержащей столбец идентификации. Кроме того, этот оператор INSERT должен быть выдан на том же уровне обработки³; это значит, что это значение доступно локально на том уровне обработки, на котором оно было присвоено, пока оно не будет замещено новым присвоенным значением.

Присвоенное значение - это или значение, заданное пользователем (если этот столбец идентификации определен как GENERATED BY DEFAULT), или значение идентификации, сгенерированное DB2.

В следующих ситуациях эта функция возвращает пустое значение:

- Если на текущем уровне обработки не был выполнен оператор вставки одной строки INSERT с условием VALUES для таблицы со столбцом идентификации.

3. При запуске триггера, функции или хранимой процедуры инициируется новый уровень обработки.

- Если после выполнения последнего присвоившего значение оператора INSERT была выполнена операция COMMIT или ROLLBACK единицы работы ⁴.

На результат этой функции не влияют:

- Оператор вставки одной строки INSERT с условием VALUES для таблицы, не содержащей столбцов идентификации.
- Оператор вставки нескольких строк INSERT с условием VALUES.
- Оператор INSERT с полной выборкой.
- Оператор ROLLBACK TO SAVEPOINT.

Примечания:

- Выражения в условии VALUES оператора INSERT вычисляются перед присваиванием значений столбцам назначения этого оператора INSERT. Поэтому вызов функции IDENTITY_VAL_LOCAL в условии VALUES оператора INSERT возвращает самое последнее значение столбца идентификации, присвоенное в предыдущем операторе INSERT. Функция вернет пустое значение, если на том же уровне обработки, на котором вызвана функция IDENTITY_VAL_LOCAL, не было выполнено предыдущих операторов вставки одной строки (INSERT) с условием VALUES для таблицы со столбцом идентификации.
- Значение столбца идентификации для таблицы, для которой определен триггер, можно узнать внутри триггера, используя временную переходную переменную триггера для столбца идентификации.
- При вызове функции IDENTITY_VAL_LOCAL внутри условия триггера операции вставки она возвращает пустое значение.
- Для таблицы может существовать несколько триггеров BEFORE или AFTER для операций вставки. В этих случаях каждый триггер выполняется отдельно и значения идентификации, присваиваемые одним действием триггера, не могут быть получены в других действиях триггеров при помощи функции IDENTITY_VAL_LOCAL. Это верно, даже если несколько действий триггеров по смыслу определены на одном уровне.
- В общем случае не рекомендуется использовать функцию IDENTITY_VAL_LOCAL в теле триггера BEFORE операции вставки. При вызове функции IDENTITY_VAL_LOCAL внутри действия триггера BEFORE операции вставки она возвращает пустое значение. Значение для столбца идентификации таблицы, для которой определен триггер, нельзя получить с помощью вызова функции IDENTITY_VAL_LOCAL внутри действия триггера BEFORE операции вставки. Однако значение столбца идентификации можно узнать внутри действия триггера, используя переходную переменную триггера для столбца идентификации.
- При вызове функции IDENTITY_VAL_LOCAL внутри действия триггера AFTER операции вставки⁵ она возвращает значение, присвоенное столбцу идентификации таблицы, указанной в самом последнем вызванном в том же действии триггера операторе вставки одной строки INSERT с условием VALUES для таблицы, содержащей столбец идентификации. Если в том же действии триггера перед вызовом функции IDENTITY_VAL_LOCAL не были выполнены операторы вставки одной строки INSERT с условием VALUES для таблицы со столбцом идентификации, функция IDENTITY_VAL_LOCAL вернет пустое значение.
- Поскольку результаты функции IDENTITY_VAL_LOCAL не детерминированы, результат вызова этой функции в операторе SELECT указателя может быть другим для каждого оператора FETCH.

4. Интерфейсы, выполняющие автоматическое принятие после каждого оператора, будут возвращать пустое значение при вызове этой функции в отдельном операторе (если не отключен режим автоматического принятия).

5. Это относится и к триггерам AFTER операции вставки типа FOR EACH ROW, и к таким триггерам типа FOR EACH STATEMENT.

IDENTITY_VAL_LOCAL

- Присвоенное значение - это значение, действительно присвоенное столбцу идентификации (то есть это значение, которое возвращает последующий оператор SELECT). Это значение не обязательно должно совпадать со значением, заданным в условии VALUES оператора INSERT или со значением, сгенерированным DB2. Присвоенное значение может быть значением, заданным в теле триггера BEFORE для операции вставки в операторе задания временной переменной (SET) для временной переменной триггера, связанной со столбцом идентификации.
- После возникновения ошибки выполнения оператора вставки одной строки INSERT с условием VALUES для таблицы со столбцом идентификации эта функция возвращает непредсказуемый результат. Возвращаемое значение может быть значением, которое вернула бы эта функция при вызове ее перед ошибочным оператором INSERT, или это может быть значение, которое было бы присвоено оператором INSERT в случае его успешного выполнения. Действительное возвращаемое значение зависит от места возникновения ошибки и поэтому непредсказуемо.

Примеры:

Пример 1: Присваивает переменной IVAR значение, присвоенное столбцу идентификации в таблице EMPLOYEE. Если это была первая операция вставки в таблицу EMPLOYEE, переменная IVAR должна иметь значение 1.

```
CREATE TABLE EMPLOYEE
(EMPNO  INTEGER GENERATED ALWAYS AS IDENTITY,
 NAME   CHAR(30),
 SALARY DECIMAL(5,2),
 DEPTNO SMALLINT)
```

Пример 2: Функция IDENTITY_VAL_LOCAL, вызванная в операторе INSERT, возвращает значение, присвоенное предыдущим оператором вставки одной строки INSERT с условием VALUES для таблицы со столбцом идентификации. Для этого примера предположим, что есть две таблицы - T1 и T2. Обе эти таблицы содержат столбец идентификации с именем C1. DB2 генерирует для столбца C1 таблицы T1 последовательные значения, начиная с 1, а для столбца C2 таблицы T2 - последовательные значения, начиная с 10.

```
CREATE TABLE T1
(C1 INTEGER GENERATED ALWAYS AS IDENTITY,
 C2 INTEGER),
CREATE TABLE T2
(C1 DECIMAL(15,0) GENERATED BY DEFAULT AS IDENTITY
(START WITH 10),
 C2 INTEGER),
INSERT INTO T1 (C2) VALUES (5),
INSERT INTO T1 (C2) VALUES (6),
SELECT * FROM T1
```

Это дает такой результат:

C1	C2
1	5
2	6

Теперь объявим функцию для переменной IVAR:

```
VALUES IDENTITY_VAL_LOCAL() INTO :IVAR
```

В этот момент функция IDENTITY_VAL_LOCAL вернет в переменную IVAR значение 2, поскольку это самое последнее значение, присвоенное DB2. Следующий оператор INSERT вставляет одну строку в таблицу T2; при этом столбцу C2 присваивается значение 2, возвращенное функцией IDENTITY_VAL_LOCAL.

```
INSERT INTO T2 (C2) VALUES (IDENTITY_VAL_LOCAL());
SELECT * FROM T2
WHERE C1 = DECIMAL(IDENTITY_VAL_LOCAL(),15,0)
```

Это дает такой результат:

C1	C2
----- 10.	----- 2

Если вызвать функцию IDENTITY_VAL_LOCAL после этой операции вставки, она вернет значение 10, сгенерированное DB2 для столбца C1 таблицы T2.

В среде с вложенными уровнями обработки, содержащей триггеры, функция IDENTITY_VAL_LOCAL используется для получения значения идентификации, присвоенного на конкретном уровне, хотя при этом могут также существовать значения, присвоенные на более низких уровнях. Допустим, существуют три таблицы EMPLOYEE, EMP_ACT и ACCT_LOG. Для операций вставки таблицы EMPLOYEE определен триггер AFTER, который выполняет дополнительные операции вставки в таблицы EMP_ACT и ACCT_LOG.

```
CREATE TABLE EMPLOYEE
(EMPNO SMALLINT GENERATED ALWAYS AS IDENTITY (START WITH 1000),
NAME CHAR(30),
SALARY DECIMAL(5,2),
DEPTNO SMALLINT);

CREATE TABLE EMP_ACT
(ACNT_NUM SMALLINT GENERATED ALWAYS AS IDENTITY (START WITH 1),
EMPNO SMALLINT);

CREATE TABLE ACCT_LOG
(ID SMALLINT GENERATED ALWAYS AS IDENTITY (START WITH 100),
ACNT_NUM SMALLINT,
EMPNO SMALLINT);

CREATE TRIGGER NEW_HIRE
AFTER INSERT ON EMPLOYEE
REFERENCING NEW AS NEW_EMP
FOR EACH ROW MODE DB2SQL
BEGIN ATOMIC
INSERT INTO EMP_ACT (EMPNO)
VALUES (NEW_EMP.EMPNO);
INSERT INTO ACCT_LOG (ACNT_NUM EMPNO)
VALUES (IDENTITY_VAL_LOCAL(), NEW_EMP.EMPNO);
END
```

Первый оператор INSERT триггера вставляет строку в таблицу EMP_ACT. Чтобы задать, что значение идентификации для столбца EMPNO таблицы EMPLOYEE должно быть скопировано в столбец EMPNO таблицы EMP_ACT, в этом операторе INSERT используется переходная переменная триггера для столбца EMPNO таблицы EMPLOYEE. Для получения значения, присвоенного столбцу EMPNO таблицы EMPLOYEE, нельзя использовать функцию IDENTITY_VAL_LOCAL. Это связано с тем, что на этом же уровне обработки не был вызван оператор INSERT и поэтому, если вызвать функцию IDENTITY_VAL_LOCAL в условии VALUES оператора INSERT для таблицы EMP_ACT, эта функция вернет пустое значение. Этот оператор INSERT для таблицы EMP_ACT также генерирует новое значение столбца идентификации для столбца ACNT_NUM.

Второй оператор INSERT триггера вставляет строку в таблицу ACCT_LOG. Чтобы задать, что значение идентификации, присвоенное столбцу ACNT_NUM таблицы EMP_ACT предыдущим оператором INSERT действия триггера, должно быть скопировано в столбец ACNT_NUM таблицы ACCT_LOG, в этом операторе INSERT используется вызов функции IDENTITY_VAL_LOCAL. Столбцу EMPNO присваивается то же значение, что и столбцу EMPNO таблицы EMPLOYEE.

В вызывающей программе (то есть на уровне, на котором выдается оператор INSERT для таблицы EMPLOYEE) присвоим переменной IVAR значение, присвоенное столбцу EMPNO таблицы EMPLOYEE исходным оператором INSERT.

IDENTITY_VAL_LOCAL

```
INSERT INTO EMPLOYEE (NAME, SALARY, DEPTNO)
VALUES ('Rupert', 989.99, 50);
```

Содержимое этих трех таблиц после выполнения этого исходного оператора INSERT и всех действий триггеров будет таким:

```
SELECT EMPNO, SUBSTR(NAME,10) AS NAME, SALARY, DEPTNO
FROM EMPLOYEE;
```

EMPNO	NAME	SALARY	DEPTNO
1000	Rupert	989.99	50

```
SELECT ACNT_NUM, EMPNO
FROM EMP_ACT;
```

ACNT_NUM	EMPNO
1	1000

```
SELECT * FROM ACCT_LOG;
```

ID	ACNT_NUM	EMPNO
100	1	1000

Функция IDENTITY_VAL_LOCAL возвращает в качестве результата самое последнее значение, присвоенное столбцу идентификации на том же уровне обработки. После выполнения этого исходного оператора INSERT и всех действий триггеров функция IDENTITY_VAL_LOCAL вернет значение 1000, так как это значение, присвоенное столбцу EMPNO таблицы EMPLOYEE. Следующий оператор VALUES присваивает переменной IVAR значение 1000. Операция вставки в таблицу EMP_ACT (которая выполнялась позже операции вставки в таблицу EMPLOYEE, но на другом уровне вложенности) не влияет на результат вызова функции IDENTITY_VAL_LOCAL.

```
VALUES IDENTITY_VAL_LOCAL() INTO :IVAR;
```

Глава 6. SQL Statements (Операторы SQL)

Составной оператор (динамический)

Составной оператор (динамический)

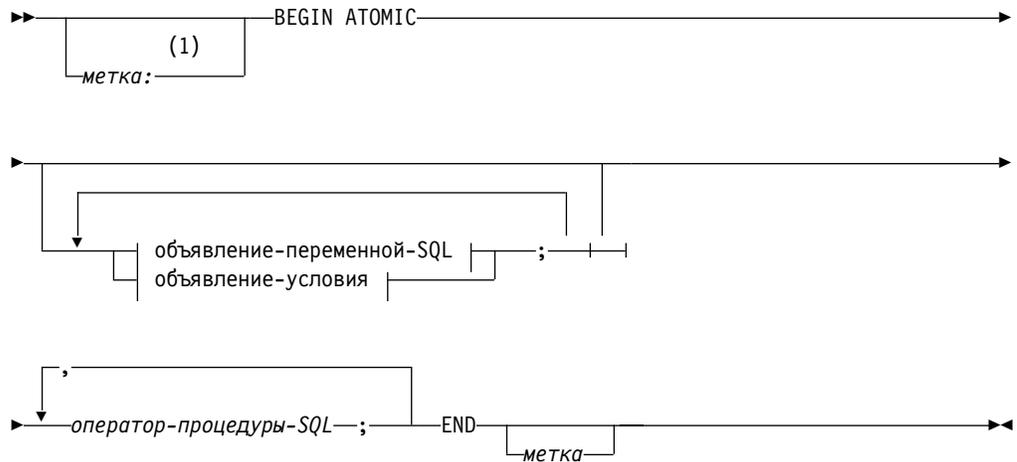
Составной оператор объединяет вместе другие операторы в один выполняемый блок. Внутри динамически подготовленного составного оператора ATOMIC можно объявлять переменные SQL.

Область действия: Этот оператор может быть вставлен в триггер, функцию SQL или метод SQL или выполняться как динамический оператор SQL. Это выполняемый оператор, который можно подготовить динамически.

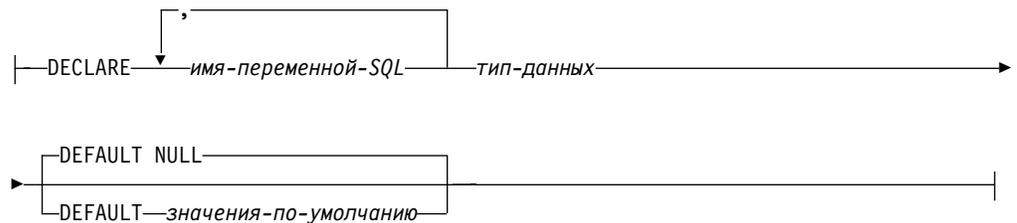
Авторизация: Для запуска динамических составных операторов не требуются привилегии. Однако ID авторизации составного оператора должен иметь привилегии, необходимые для вызова входящих в него операторов SQL.

Синтаксис команды:

динамический-составной-оператор



объявление-переменной-SQL:



объявление-условия:



Составной оператор (динамический)



Примечания:

- 1 Метка может быть задана, только если этот оператор находится в определении функции, метода или триггера.

Параметры команды:

метка

Задаёт метку для этого блока кода. Если задана метка начала, её можно использовать в качестве спецификатора для переменных SQL, объявленных в этом динамическом составном операторе; её можно также использовать в операторе LEAVE. Если задана метка конца, она должна совпадать с меткой начала.

АТОМІС

Ключевое слово АТОМІС указывает, что при возникновении ошибки в этом составном операторе будет выполнен откат для всех выполненных в этом составном операторе операторов SQL, а оставшиеся в этом составном операторе операторы SQL не будут выполняться.

оператор-процедуры-SQL

Внутри динамического составного оператора можно использовать следующие *управляющие-операторы-SQL*:

- оператор FOR
- оператор GET DIAGNOSTICS
- оператор IF
- оператор ITERATE
- оператор LEAVE
- оператор SIGNAL
- оператор WHILE

Операторы SQL, которые могут быть вызваны:

- полная выборка⁶
- UPDATE с поиском
- DELETE с поиском
- INSERT
- оператор SET переменная

объявление-переменной-SQL

Объявляет переменную, локальную для этого динамического составного оператора.

имя-переменной-SQL

Задаёт имя локальной переменной. DB2 преобразует все имена переменных SQL в верхний регистр. Это имя не может:

- Совпадать с именем другой переменной SQL в том же составном операторе.
- Совпадать с именем параметра.

6. Перед полной выборкой может идти общее-табличное-выражение

Составной оператор (динамический)

- Совпадать с именем столбца.

Если оператор SQL содержит идентификатор, совпадающий и с именем переменной SQL, и с именем столбца, DB2 интерпретирует этот идентификатор как имя столбца.

тип-данных

Задаёт тип данных этой переменной.

DEFAULT значения-по-умолчанию или **NULL**

Задаёт значение по умолчанию для этой переменной SQL. Переменная инициализируется при вызове этого динамического составного оператора. Если значение по умолчанию не задано, переменная инициализируется значением NULL.

объявление-условия

Объявляет имя условия и соответствующее значение SQLSTATE.

имя-условия

Задаёт имя условия. Имя условия должно быть уникальным в теле процедуры; на него можно ссылаться только внутри составного оператора, в котором оно определено.

FOR SQLSTATE строковая-константа

Задаёт SQLSTATE для этого условия. *Строковая-константа* должна представлять собой пять символов, заключённых в простые кавычки; она не может быть равна '00000'.

Замечания по использованию:

- Динамический составной оператор компилируется системой DB2 как один оператор. Такой оператор эффективен для коротких алгоритмов, содержащих небольшое число управляющих конструкций, но обрабатывающих значительное число данных. Для более сложных алгоритмов, содержащих сложные вложенные управляющие конструкции, лучше использовать процедуры SQL.

Примера:

Глава 7 "SQL Procedures" теперь называется "Управляющие операторы SQL"

Управляющие операторы - это операторы SQL, позволяющие при написании программ на языке SQL использовать методы структурного программирования. Управляющие операторы SQL можно использовать в теле процедуры, триггера или динамического составного оператора.

В этой главе представлены синтаксис и описание поддерживаемых управляющих операторов SQL и оператора-процедуры-SQL.

Оператор процедуры SQL

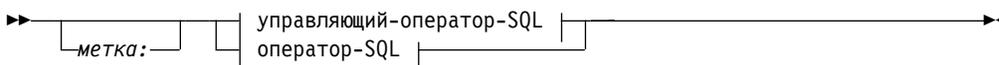
Информация об операторе процедуры SQL изменена следующим образом:

Оператор процедуры SQL

Оператор процедуры SQL

В этой главе представлены синтаксические диаграммы, описания семантики, правила и примеры использования операторов, составляющих тело процедуры SQL, триггера или динамического составного оператора.

Синтаксис команды:



управляющий-оператор-SQL:

оператор <i>ALLOCATE CURSOR</i>	(1)
оператор присваивания	(1)
оператор <i>ASSOCIATE LOCATORS</i>	(1)
оператор <i>CASE</i>	(1)
динамический составной оператор	(2)
оператор <i>FOR</i>	
оператор <i>GET DIAGNOSTICS</i>	
оператор <i>GOTO</i>	(1)
оператор <i>IF</i>	
оператор <i>ITERATE</i>	
оператор <i>LEAVE</i>	
оператор <i>LOOP</i>	(1)
составной оператор процедуры	(1)
оператор <i>REPEAT</i>	(1)
оператор <i>RESIGNAL</i>	(1)
оператор <i>RETURN</i>	
оператор <i>SIGNAL</i>	
оператор <i>WHILE</i>	

Примечания:

- 1 Этот оператор поддерживается только внутри процедуры SQL.
- 2 Этот оператор поддерживается только внутри триггера, функции SQL или метода SQL. Он должен быть самым внешним оператором.

Параметры команды:

метка:

Задаёт метку для оператора процедуры SQL. Эта метка должна быть уникальной в списке операторов процедуры SQL, включая все вложенные составные операторы в этом списке. Обратите внимание, что в составных операторах, не

являющихся вложенными, может использоваться та же метка. Список операторов процедуры SQL может содержать некоторое число управляющих операторов SQL.

В контексте триггера, функции SQL, метода SQL или динамического составного оператора метки можно задавать только для динамических составных операторов и операторов FOR и WHILE.

оператор-SQL

Тело процедуры SQL может содержать все выполняемые операторы SQL, за исключением следующих:

- CONNECT
- CREATE для любого объекта, кроме индекса, таблицы или производной таблицы
- DESCRIBE
- DISCONNECT
- DROP для любого объекта, кроме индекса, таблицы или производной таблицы
- FLUSH EVENT MONITOR
- REFRESH TABLE
- RELEASE (только соединение)
- RENAME TABLE
- RENAME TABLESPACE
- REVOKE
- SET CONNECTION
- SET INTEGRITY

Примечание: В тело процедуры SQL можно включить операторы CALL, но эти операторы CALL могут вызывать только другие процедуры SQL или процедуры C. Операторы CALL в теле процедуры SQL не могут вызывать другие типы хранимых процедур.

Заголовок раздела **Compound Statement** изменен на **Составной оператор процедуры**

Составной оператор процедуры объединяет вместе другие операторы в процедуру SQL. Внутри составного оператора можно объявлять переменные SQL, указатели и обработчики условий.

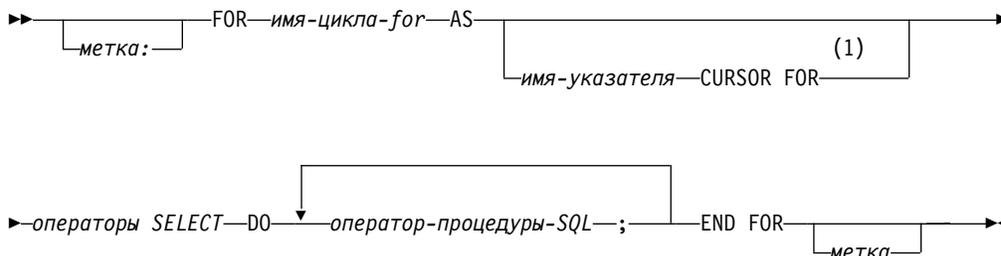
Синтаксическая диаграмма теперь имеет заголовок: составной-оператор-процедуры.

Оператор FOR

Оператор FOR

Оператор FOR выполняет некоторый оператор или группу операторов для каждой строки таблицы.

Синтаксис команды:



Примечания:

1 Эта опция может использоваться только в контексте процедуры SQL.

Параметры команды:

метка

Задаёт метку для этого оператора FOR. Если задана метка начала, её можно использовать в операторах LEAVE и ITERATE. Если задана метка конца, она должна совпадать с меткой начала.

имя-цикла-for

Задаёт метку для неявного составного оператора, создаваемого для реализации этого оператора FOR. Она подчиняется тем же правилам, что метка составного оператора, но её нельзя использовать в операторе ITERATE или LEAVE внутри этого оператора FOR. *Имя-цикла-for* используется в качестве спецификатора для имен столбцов, возвращаемых заданным *оператором-выборки*.

имя-указателя

Задаёт имя указателя, используемого для выборки строк из таблицы результатов оператора SELECT. Если это имя не задано, DB2 генерирует уникальное имя указателя.

оператор-выборки

Задаёт оператор SELECT для этого указателя. Все столбцы в списке выборки должны иметь имена и имена столбцов не должны совпадать.

В триггере, функции, методе или динамическом составном операторе *оператор-выборки* должен состоять только из *полной выборки* с необязательным общим табличным выражением.

оператор-процедуры-SQL

Задаёт оператор (или операторы), который должен быть выполнен для каждой строки таблицы.

- Список выборки должен состоять из уникальных имен столбцов, а таблица, заданная в списке выборки, должна существовать в момент создания процедуры или создаваться в одном из предыдущих операторов процедуры SQL.
- Указатель, заданный в операторе-for, нельзя использовать вне этого оператора-for; его нельзя задавать в операторах OPEN, FETCH или CLOSE.

Примера: В следующем примере оператор-for используется для обработки всей таблицы employee. Для каждой строки этой таблицы переменной SQL fullname

Оператор FOR

присваивается строка, содержащая: фамилию работника, запятую, имя, пробел и инициал отчества. Каждое значение fullname вставляется в таблицу tnames.

```
BEGIN
  DECLARE fullname CHAR(40);
  FOR v1 AS
    SELECT firstnme, midinit, lastname FROM employee
  DO
    SET fullname = lastname || ',' || firstnme || ' ' || midinit;
    INSERT INTO tnames VALUE (fullname);
  END FOR
END
```

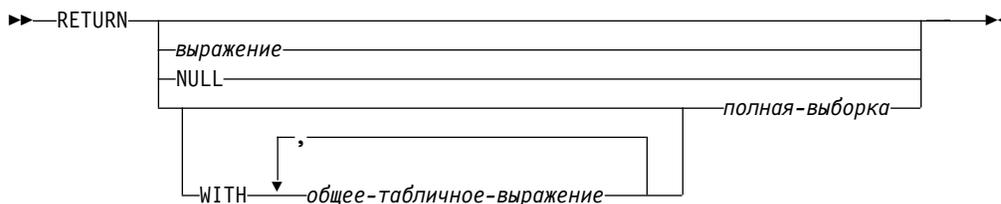
Оператор RETURN

Оператор RETURN

оператор RETURN

Оператор RETURN используется для возврата из подпрограммы. Для функций SQL и методов SQL он возвращает значение результата этой функции или метода. Для процедуры SQL он может возвращать необязательное целое значение состояния.

Синтаксис команды:



Параметры команды:

выражение

Задаёт значение, возвращаемое из подпрограммы:

- Если подпрограмма - это функция или метод, необходимо задать *выражение*, NULL или *полную выборку* (SQLSTATE 42630), а тип данных результата должен быть приводимым к типу данных RETURNS этой подпрограммы (SQLSTATE 42866).
- Для табличной функции нельзя задавать скалярное выражение (кроме скалярной полной выборки) (SQLSTATE 428F1).
- Если подпрограмма - это процедура, *выражение* должно иметь тип данных INTEGER (SQLSTATE 428E2). Процедура не может возвращать NULL или полную выборку.

NULL

Задаёт, что функция или метод возвращает пустое значение типа данных, определенного в условии RETURNS. Ключевое слово NULL нельзя задавать для оператора RETURN процедуры.

WITH *общее-табличное-выражение*

Определяет общее табличное выражение, используемое с задаваемой после него полной выборкой.

полная-выборка

Задаёт строку или строки, возвращаемые для этой функции. Число столбцов в полной выборке должно совпадать с числом столбцов в результате функции (SQLSTATE 42811). Кроме того, статические типы столбцов полной выборки должны быть приводимы к объявленным типам столбцов результата функции с помощью правил присваивания столбцов (SQLSTATE 42866).

Полную-выборку нельзя задавать для оператора RETURN процедуры.

Если подпрограмма - это скалярная функция или метод, полная выборка должна возвращать один столбец (SQLSTATE 42823) и не более одной строки (SQLSTATE 21000).

Если подпрограмма - это функция строки, она должна возвращать не более одной строки (SQLSTATE 21505).

Если подпрограмма - это табличная функция, она может возвращать ноль или большее число строк с одним или более столбцами.

Оператор RETURN

- Выполнение функции SQL или метода SQL должно заканчиваться оператором RETURN (SQLSTATE 42632).
- В табличной функции SQL или функции строки SQL, использующей *динамический-составной-оператор*, оператор RETURN разрешено использовать только в конце составного оператора (SQLSTATE 429BD).

Замечания по использованию:

- Когда процедура возвращает значение, вызывающая программа может получить это значение, используя:
 - оператор GET DIAGNOSTICS, чтобы получить RETURN_STATUS, если эта процедура SQL была вызвана из другой процедуры SQL
 - параметр, связанный с маркером параметра для возвращаемого значения в условии ESCAPE при соответствующем синтаксисе оператора CALL (?=CALL...) в прикладной программе CLI
 - напрямую из SQLCA, возвращенной после выполнения вызова (CALL) процедуры SQL - получив значение SQLERRD[0], если SQLCODE не меньше нуля (предполагая значение -1, если SQLCODE меньше нуля).

Примера: Используется оператор RETURN для возвращения из хранимой процедуры SQL - со значением 0 при успешном выполнении процедуры и со значением -200 в случае ошибки.

```
BEGIN
...
  GOTO FAIL
...
  SUCCESS: RETURN 0
  FAIL: RETURN -200
END
```

Оператор SIGNAL

Оператор SIGNAL SQLSTATE более не используется, вместо него используется этот оператор.

имя-переменной

Указывает переменную SQL, которая должна быть объявлена внутри этого составного оператора. Эта переменная SQL должна быть определена с типом данных CHAR или VARCHAR.

строковая-константа-диагностики

Задаёт символьную строку-константу, содержащую текст сообщения.

строка-диагностики

Выражение типа CHAR или VARCHAR, возвращающее символьную строку длиной до 70 байт, описывающую условие ошибки. Если эта строка длиннее 70 байт, она будет усечена. Эта опция сделана доступной только в области видимости оператора CREATE TRIGGER для совместимости с более ранними версиями DB2. Не рекомендуется использовать ее в обычных случаях.

Замечания по использованию:

- При выполнении оператора SIGNAL SQLCODE присваивается следующее значение:
 - +438, если SQLSTATE начинается с '01' или '02'
 - 438 в остальных случаях
- Если SQLSTATE или условие указывает на исключительную ситуацию (класс SQLSTATE, отличный от '01' или '02'):
 - Выполняется обработка этой исключительной ситуации и управление передается ее обработчику, если обработчик существует в том же составном операторе (или во внешнем составном операторе), что и оператор SIGNAL, и составной оператор содержит обработчик для заданных SQLSTATE, имени-условия или SQLEXCEPTION;
 - Если исключительная ситуация не может быть обработана, управление сразу передается в конец этого составного оператора.
- Если SQLSTATE или условие указывает на предупреждение (класс SQLSTATE '01') или на условие "не найден" (класс SQLSTATE '02'):
 - Выполняется обработка этого предупреждения или условия "не найден" и управление передается обработчику, если обработчик существует в том же составном операторе (или во внешнем составном операторе), что и оператор SIGNAL, и составной оператор содержит обработчик для заданных SQLSTATE, имени-условия, SQLWARNING (если класс SQLSTATE равен '01') или NOT FOUND (если класс SQLSTATE равен '02');
 - Если предупреждение не может быть обработано, обработка продолжается со следующего оператора.
- Значения SQLSTATE состоят из двухсимвольного значения кода класса, после которого идет трехсимвольное значение кода подкласса. Значения кодов классов представляют классы условий успешного и неуспешного выполнения.

В операторе SIGNAL можно использовать любое допустимое значение SQLSTATE. Однако программистам рекомендуется определять новые значения SQLSTATE в диапазонах их значений, зарезервированных для прикладных программ. Это позволит избежать случайного использования значения SQLSTATE, которое может быть определено менеджером баз данных в будущих выпусках.

 - Можно определить классы SQLSTATE, начинающиеся с символов от '7' до '9' или с символов от 'I' до 'Z'. Внутри этих классов можно определить любые подклассы.
 - Классы SQLSTATE, начинающиеся с символов от '0' до '6' или с символов от 'A' до 'H', зарезервированы для менеджера баз данных. Внутри этих классов

Оператор SIGNAL

подклассы, начинающиеся с символов от 'O' до 'H', зарезервированы для менеджера баз данных. Можно определить подклассы, начинающиеся с символов от 'I' до 'Z'.

Примера: Процедура SQL для более старой системы, которая сигнализирует об ошибке прикладной программы, если прикладной программе не известен номер заказчика. Таблица ORDERS (заказы) содержит внешний ключ для таблицы CUSTOMER (заказчик), требуя, чтобы CUSTNO (номер заказчика) существовал, прежде чем можно будет принять заказ.

```
CREATE PROCEDURE SUBMIT_ORDER
  (IN ONUM INTEGER, IN CNUM INTEGER,
   IN PNUM INTEGER, IN QNUM INTEGER)
  SPECIFIC SUBMIT_ORDER
  MODIFIES SQL DATA
  LANGUAGE SQL
  BEGIN
    DECLARE EXIT HANDLER FOR SQLSTATE VALUE '23503'
      SIGNAL SQLSTATE '75002'
      SET MESSAGE_TEXT = 'Неизвестный номер заказчика';
    INSERT INTO ORDERS (ORDERNO, CUSTNO, PARTNO, QUANTITY)
      VALUES (ONUM, CNUM, PNUM, QNUM);
  END
```

Приложение D: Catalog Views (Производные таблицы каталога)

Добавлена новая производная таблица каталога:

SYSCAT.SEQUENCES

Производная таблица SYSCAT.SEQUENCES автоматически генерируется для баз данных, созданных с FixPak 3 или более поздним. Чтобы добавить эта производную таблицу в базу данных, созданную до FixPak 3, выполните команду **db2updv7**. Подробную информацию смотрите в разделе "Изменения в Command Reference" Замечаний по выпуску.

Эта производная таблица каталога обновляется при обычной работе при выполнении операторов SQL определения данных, процедур среды и некоторых утилит. Данные из этой производной таблицы каталога можно получить с помощью обычных средств запросов SQL. Имена столбцов соответствуют типам объектов, заданным при их описании.

Таблица 15. Столбцы производной таблицы каталога SYSCAT.SEQUENCES

Имя столбца	Тип данных	Допускаются пустые значения	Описание
SEQSHEMA	VARCHAR(128)		Схема последовательности.
SEQNAME	VARCHAR(128)		Имя последовательности (генерируется системой DB2 для столбца идентификации).
DEFINER	VARCHAR(128)		Определяющий последовательности.
OWNER	VARCHAR(128)		Владелец последовательности.
SEQID	INTEGER		Внутренний ID последовательности.
SEQTYPE	CHAR(1)		Тип последовательности S - Обычная последовательность
INCREMENT	DECIMAL(31,0)		Значение приращения.
START	DECIMAL(31,0)		Начальное значение.

Таблица 15. Столбцы производной таблицы каталога SYSCAT.SEQUENCES (продолжение)

Имя столбца	Тип данных	Допускаются пустые значения	Описание
MAXVALUE	DECIMAL(31,0)	Да	Максимальное значение.
MINVALUE	DECIMAL(31,0)		Минимальное значение.
CYCLE	CHAR(1)		Выполняется ли циклическая генерация после достижения граничного значения: Y - циклическая генерация выполняется N - циклическая генерация не выполняется
CACHE	INTEGER		Число значений последовательности, заранее вычисляемых и сохраняемых в памяти для ускорения доступа. 0 означает, что значения не будут вычисляться заранее.
ORDER	CHAR(1)		Должны ли значения последовательности генерироваться в порядке запросов: Y - значения последовательности должны генерироваться в порядке запросов N - значения последовательности не обязательно должны генерироваться в порядке запросов
DATATYPEID	INTEGER		Для встроенных типов - внутренний ID этого встроенного типа. Для особых типов - внутренний ID этого особого типа.
SOURCETYPEID	INTEGER		Для встроенного типа этот столбец имеет значение 0. Для особого типа - это внутренний ID встроенного типа, на основе которого определен этот особый тип.
CREATE_TIME	TIMESTAMP		Время создания последовательности.
ALTER_TIME	TIMESTAMP		Время, когда для этой последовательности последний раз выполнялся оператор ALTER SEQUENCE.
PRECISION	SMALLINT		Точность, определенная для последовательности десятичного или числового типа. Значения: 5 для SMALLINT, 10 для INTEGER и 19 для BIGINT.
ORIGIN	CHAR(1)		Происхождение последовательности U - последовательность, сгенерированная пользователем S - последовательность, сгенерированная системой
REMARKS	VARCHAR(254)	Да	Символьная строка, заданная пользователем в операторе COMMENT ON.

SYSCAT.SEQUENCES

Глава 30. Построитель хранимых процедур DB2

Поддержка Java 1.2 для построителя хранимых процедур DB2

Построитель хранимых процедур DB2 поддерживает построение хранимых процедур с использованием возможностей Java 1.2. Кроме этого, построитель хранимых процедур поддерживает языки с двумя направлениями письма, такие как арабский или иврит, используя средства поддержки таких языков в Java 1.2.

Эта поддержка есть только для платформ Windows NT.

Необходимо установить Java 1.2, чтобы построитель хранимых процедур мог распознать и использовать его возможности.

Чтобы установить Java 1.2:

1. JDK 1.2.2 находится на компакт-диске DB2 UDB в каталоге DB2\bid\NT.
ibm-inst-n122p-win32-x86.exe - это программа установки, а
ibm-jdk-n122p-win32-x86.exe - дистрибутив JDK. Скопируйте оба эти файла во временный каталог на жестком диске и затем запустите оттуда программу установки.
2. Установите этот продукт в каталог <DB2PATH>\java\Java12, где <DB2PATH> - путь установки DB2.
3. При установке JDK/JRE не выбирайте JDK/JRE в качестве системной VM.

Успешно установив Java 1.2, запускайте построитель хранимых процедур обычным образом.

Для выполнения хранимых процедур Java с использованием поддержки JDK 1.2 задайте для переменной среды сервера баз данных DB2_USE_JDK12 значение TRUE, используя следующую команду:

```
DB2SET DB2_USE_JDK12=TRUE
```

Задайте также параметр конфигурации JDK11_PATH, чтобы он указывал на каталог, в котором установлена поддержка Java 1.2. Используйте для этого следующую команду:

```
DB2 UPDATE DBM CFG USING JDK11_PATH
```

Чтобы прекратить использование Java 1.2, можно или деинсталлировать JDK/JRE из каталога <DB2PATH>\java\Java12, или просто переименовать подкаталог <DB2PATH>\java\Java12.

Внимание: Не перепутайте <DB2PATH>\java\Java12 с <DB2PATH>\Java12. <DB2PATH>\Java12 - это часть установки DB2, содержащая поддержку JDBC для Java 1.2.

Удаленная отладка хранимых процедур DB2

Чтобы использовать возможность удаленной отладки хранимых процедур Java и C на платформах UNIX и Windows, нужно установить отладчик IBM Distributed Debugger. IBM Distributed Debugger находится на компакт-диске Visual Age for Java Professional Edition. Клиент этого отладчика работает только на платформах Windows. Поддерживаемые платформы серверов: Windows, AIX и Solaris. Для отладки локальных и удаленных хранимых процедур SQL на платформах Windows и UNIX используйте встроенную возможность отладки SQL построителя хранимых процедур. Поддержка для платформы OS/2 в настоящее время недоступна. Дополнительную информацию о построителе хранимых процедур DB2 for OS/390 можно найти на следующем Web-сайте: <http://www-4.ibm.com/software/data/db2/os390/spb/exciting>

Для отладки процедур SQL на платформах OS/390 необходимо также иметь продукт IBM C/C++ Productivity Tools for OS/390 R1. Дополнительную информацию об IBM C/C++ Productivity Tools for OS/390 R1 можно найти на следующем Web-сайте: <http://www.ibm.com/software/ad/c390/pt/>

Построение процедур SQL на платформах Windows, OS/2 или UNIX

Прежде, чем можно будет использовать построитель хранимых процедур для успешного построения процедур SQL в базе данных на платформе Windows, OS/2 или UNIX, необходимо сконфигурировать используемый сервер для процедур SQL. Информацию о том, как сконфигурировать сервер для процедур SQL, смотрите в разделе “Глава 4. Building Java Applets and Applications” на стр. 231.

Использование построителя хранимых процедур DB2 на платформе Solaris

Чтобы использовать построитель хранимых процедур DB2 на платформе Solaris:

1. Загрузите и установите JDK 1.1.8. Его можно загрузить с Web-сайта JavaSoft.
2. В переменной среды JAVA_HOME задайте каталог установки JDK.
3. В параметре конфигурации JDK11_PATH задайте каталог установки JDK.

Чтобы задать JDK11_PATH, используйте команду:

```
DB2 UPDATE DBM CFG USING JDK11_PATH.
```

Известные проблемы и ограничения

- Процедуры SQL в настоящее время не поддерживаются в Windows 98.
- Для хранимых процедур Java имена классов, имена методов и ID JAR не должны содержать символы, не являющиеся символами ASCII.
- В AS/400 для OS/400 V4R4 необходимо применить следующие PTF для V4R4:
 - SF59674
 - SF59878
- В выводимом в панели редактора исходном тексте хранимой процедуры, восстановленном из базы данных, не показываются параметры хранимых процедур с символьным подтипом FOR MIXED DATA или FOR SBCS DATA.
- В настоящее время есть ошибка при получении исходного кода Java из базы данных. При получении кода в нем повреждаются комментарии. Это будет влиять на пользователей построителя хранимых процедур DB2, работающих в не-ASCII кодовых страницах и использующих клиенты и серверы с другими кодовыми страницами.

Использование построителя хранимых процедур DB2 с традиционным китайским языком

Есть ошибка при использовании Java Development Kit или Java Runtime 1.1.8 с традиционным китайским языком. Элементы графического интерфейса построителя хранимых процедур (включая меню, текст редактора, сообщения и т.д.) выводятся неправильно. Чтобы исправить эту ошибку, измените файл `font.properties.zh_TW`, находящийся в одном или в обоих каталогах:

```
sqllib/java/jdk/lib
sqllib/java/jre/lib
```

Измените:

```
monospaced.0=\u7d30\u660e\u9ad4,CHINESEBIG5_CHARSET,NEED_CONVERTED
```

на:

```
monospaced.0=Courier New,ANSI_CHARSET
```

Установки в UNIX (AIX, Sun Solaris, Linux) и построитель хранимых процедур

Если для установок в Sun Solaris используется Java Development Kit или Java Runtime, отличающиеся от установленных в AIX с UDB, необходимо задать в переменной среды `JAVA_HOME` путь установки Java (то есть имя каталога, содержащего каталоги `/bin` и `/lib`). Построитель хранимых процедур не поддерживается в Linux, но его можно использовать на поддерживаемых платформах для построения и выполнения хранимых процедур в системах DB2 UDB для систем Linux.

Поддерживаемые платформы для клиента - AIX, Solaris и NT, для сервера - AIX, Solaris, NT, Linux, OS/2, HP-UX и NUMA-Q.

Построение хранимых процедур SQL в OS/390

Построитель хранимых процедур DB2 поддерживает построение хранимых процедур SQL на сервере DB2 UDB for OS390 V7.

Отладка хранимых процедур SQL

На платформах Windows и UNIX отладка хранимых процедур SQL теперь прямо интегрирована в построитель хранимых процедур DB2. Опция конфигурации менеджера баз данных KEEPDATA для отладки неизолированных (проверенных) процедур SQL может иметь значения YES или NO, но для отладки изолированных (непроверенных) процедур SQL она должна иметь значение YES (по умолчанию). Дополнительную информацию об использовании интегрированного отладчика смотрите в справке построителя хранимых процедур.

Экспорт хранимых процедур Java

Построитель хранимых процедур DB2 теперь поддерживает экспорт хранимых процедур Java. Чтобы экспортировать хранимую процедуру Java:

1. Щелкните правой кнопкой мыши по папке хранимых процедур и выберите Экспорт хранимых процедур Java, чтобы открыть окно Экспорт хранимых процедур Java.
2. Выберите хранимые процедуры, которые вы хотите экспортировать, и переместите их в столбец "Выбранные хранимые процедуры".

3. Выберите нужные опции и нажмите кнопку ОК.

Вставка хранимых процедур в OS/390

Для построителя хранимых процедур DB2 Версии 5 и новее, работающего на OS/390, если вы используете мастер для вставки хранимой процедуры и не задаете опций среды WLM, генерируемый код содержит следующий текст: NO WLM ENVIRONMENT. Эта строка кода заставляет запускать хранимую процедуру в адресном пространстве SPAS, как и требуется. Это исправление устраняет ошибку, существующую в построителе хранимых процедур DB2 версии 6 и выше.

После исправления генерируемый код выглядит так:

```
CREATE PROCEDURE SYSPROC.Proc2 ( )
  RESULT SETS 1
  LANGUAGE SQL
  MODIFIES SQL DATA
  COLLID TEST
  NO WLM ENVIRONMENT
  ASUTIME NO LIMIT
  RUN OPTIONS 'NOTEST(ALL,*,VADTCP&9.112.14.91:*)'
-----
-- SQL Stored Procedure
-----
P1: BEGIN
  -- Declare cursor
  DECLARE cursor1 CURSOR WITH RETURN FOR
    SELECT * FROM SYSIBM.SYSPROCEDURES;

  -- Cursor left open for client application
  OPEN cursor1;

END P1
```

Задание опций построения для хранимых процедур SQL на сервере рабочих станций

Используя построитель хранимых процедур DB2 на платформах UNIX и Windows, вы можете задать опции построения для всех хранимых процедур SQL. В число этих опций входят следующие переменные реестра компилятора и прекомпилятора DB2:

- DB2_SQLROUTINE_PREPOPTS
- DB2_SQLROUTINE_COMPILER_PATH
- DB2_SQLROUTINE_COMPILE_COMMAND
- DB2_SQLROUTINE_KEEP_FILES

Хотя можно задать эти переменные реестра и с помощью команды db2set, использование построителя хранимых процедур позволяет обойтись без физического обращения к серверу баз данных, а также без остановки и перезапуска сервера для вступления изменений в силу.

Чтобы открыть окно Опции построителя хранимых процедур SQL, щелкните правой кнопкой мыши по соединению с базой данных в окне вашего проекта и выберите Опции построителя хранимых процедур SQL. Дополнительную информацию о задании этих опций смотрите в справке по хранимым процедурам DB2.

Автоматическое обновление адресного пространства WLM для построения хранимых процедур в OS/390

После того, как вы успешно постройте в OS/390 хранимую процедуру, которая будет выполняться в WLM, построитель хранимых процедур DB2 автоматически обновляет адресное пространство WLM.

Глава 31. Обновления для Unicode

Введение

Стандарт Unicode - универсальная схема кодирования символов для записи символов и текста. Unicode использует многобайтное представление символов. Unicode представляет логичный способ кодирования многоязычных текстов, поддерживающий международный обмен текстовыми данными и создающий основу для написания программ, работающих в любой языковой среде.

В Unicode есть две схемы кодирования. По умолчанию используется схема кодирования UTF-16 с 16-битным форматом. UCS-2 представляет собой поднабор UTF-16, использующий для представления одного символа два байта. UCS-2 обычно воспринимается как универсальная кодовая страница, способная представлять все необходимые символы всех существующих одно- и двухбайтных кодовых страниц. UCS-2 зарегистрирована в IBM как кодовая страница 1200.

Другой формат кодирования Unicode - байт-ориентированный формат UTF-8, разработанный для облегчения его использования с существующими системами на основе ASCII. UTF-8 использует для хранения одного символа переменное число байтов (обычно 1 - 3, иногда 4). Инвариантные символы ASCII хранятся в виде одиночных байтов. Любой другой символ хранится, как несколько байт. В общем случае данные UTF-8 можно рассматривать как расширение данных ASCII, а не как код для многобайтных кодовых страниц. UTF-8 зарегистрирован в IBM как кодовая страница 1208.

Важно, чтобы прикладные программы учитывали требования к размеру данных при их преобразовании между национальной кодовой страницей, UCS-2 и UTF-8. Например, для 20 символов в UCS-2 потребуется ровно 40 байтов, а в UTF-8 - от 20 до 60 в зависимости от исходной кодовой страницы и использованных символов.

Базы данных Unicode и прикладные программы DB2

В Unix, Windows и OS/2 база данных DB2 UDB, созданная с кодовым набором UTF-8, может использоваться для хранения данных как в формате UCS-2, так и UTF-8. Такая база данных называется базой данных Unicode. Данные SQL с типом CHAR кодируются с использованием UTF-8, а данные SQL с типом GRAPHIC кодируются с использованием UCS-2. Это можно сопоставить с сохранением однобайтных (SBCS) и многобайтных кодовых наборов (MBCS) в столбцах CHAR и сохранением двухбайтных кодовых наборов (DBCS) в столбцах GRAPHIC.

Кодовая страница прикладной программы может не совпадать с кодовой страницей, которую DB2 использует для хранения данных. В других базах данных (не Unicode) при несовпадении этих кодовых страниц менеджер баз данных преобразует символьные и графические (исключительно DBCS) данные при переносе их между клиентом и сервером. В базе данных Unicode менеджер баз данных автоматически выполняет преобразование символьных данных между кодовой страницей клиента и UTF-8, но все графические данные (UCS-2) передаются между клиентом и сервером без преобразования.

Преобразования кодовых страниц, выполняемые менеджером баз данных

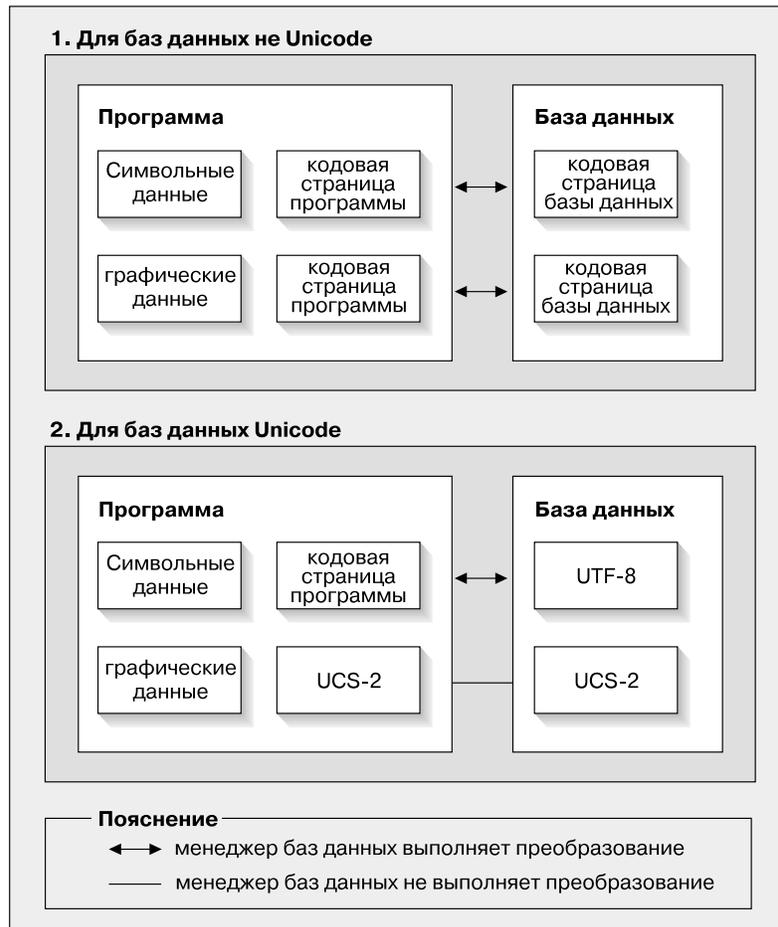


Рисунок 1. Преобразования кодовых страниц, выполняемые менеджером баз данных

Примечания:

1. Если при подключении к базам данных Unicode прикладная программа устанавливает `DB2CODEPAGE=1208`, локальная кодовая страница будет UTF-8, и преобразование кодовых страниц не требуется.
2. При подключении к базе данных Unicode прикладные программы CLI могут также принимать символьные данные как графические данные, а графические данные - как символьные данные.

Для прикладной программы можно указать кодовую страницу UTF-8, что означает, что она будет передавать и получать все графические данные в UCS-2, а символьные данные - в UTF-8. Эта кодовая страница прикладной программы поддерживается только для баз данных Unicode.

При использовании Unicode необходимо также принимать во внимание следующее:

1. Кодовая страница определяется во время создания базы данных, и по умолчанию ее значение определяется из версии (или кодовой страницы) операционной системы. Чтобы создать базу данных DB2 Unicode явным образом, можно использовать ключевые слова `CODESET` и `TERRITORY`. Например:

```
CREATE DATABASE unidb USING CODESET UTF-8 TERRITORY US
```

2. Кроме того, кодовая страница прикладной программы по умолчанию совпадает с локальной кодовой страницей, но может быть заменена на UTF-8 одним из двух способов:
 - Установкой UTF-8 (1208) для кодовой страницы прикладной программы командой:


```
db2set DB2CODEPAGE=1208
```
 - Для прикладных программ CLI/ODBC - путем вызова `SQLSetConnectAttr()` и заданием для `SQL_ATTR_ANSI_APP` значения `SQL_AA_FALSE`. Значение по умолчанию - `SQL_AA_TRUE`.
3. Данные в столбцах GRAPHIC будут занимать ровно по два байта на каждый символ Unicode, а данные в столбцах CHAR на каждый символ Unicode будут занимать от 1 до 3 байтов. Ограничения SQL на число символов для столбцов GRAPHIC в общем случае вдвое меньше, чем для столбцов CHAR, но по числу байтов они одинаковы. Максимальная длина в символах для столбца CHAR равна 254. Максимальная длина в символах для графического столбца равна 127. Дополнительную информацию смотрите описание MAX в главе "Functions" справочника *SQL Reference*.
4. Графический литерал отличают от символьного литерала по префиксу G. Например:


```
SELECT * FROM mytable WHERE mychar = 'данные utf-8' AND mygraphic = G'данные ucs-2'
```

Примечание: Для баз данных Unicode префикс G не требуется. Дополнительная информация и обновленная поддержка приводятся в разделе "Литералы в базах данных Unicode" на стр. 371.
5. Поддержка для прикладных программ CLI/ODBC и JDBC отличается от поддержки для встроенных программ. Конкретную информацию о поддержке CLI/ODBC смотрите в разделе "CLI Guide and Reference" на стр. 368.
6. Число байтов, необходимых для данных UCS-2, может быть различным на разных платформах. Во внутреннем формате DB2 используется прямой порядок байтов.

Обновление документации

В этом документе изменена следующая информация об использовании Unicode с DB2 версии 7.1:

- *SQL Reference*:
 - Глава 3. Language Elements
 - Глава 4. Functions
- *CLI Guide and Reference*:
 - Глава 3. Using Advanced Features
 - Приложение C. DB2 CLI and ODBC
- *Data Movement Utilities Guide and Reference*, Приложение C. Export/Import/Load Utility File Formats

Дополнительную информацию об использовании Unicode с DB2 смотрите в книге *Administration Guide*, Приложение J. Поддержка национальных языков (NLS): "Поддержка Unicode/UCS-2 и UTF-8 в DB2 UDB".

Глава 3. Language Elements

Расширение типов данных

В этом разделе в таблице 5 показан список приоритетов типов данных. Обратите внимание:

- Для базы данных Unicode считаются эквивалентными следующие типы данных:
 - CHAR и GRAPHIC
 - VARCHAR и VARGRAPHIC
 - LONG VARCHAR и LONG VARGRAPHIC
 - CLOB и DBCLOB
- В базе данных Unicode можно создать такие функции, у которых единственной разницей в записи будут только эквивалентные типы данных CHAR и GRAPHIC, например, foo(CHAR(8)) и foo(GRAPHIC(8)). Мы настоятельно не рекомендуем определять такие дублирующие функции, поскольку при обновлении до следующего выпуска одну из них придется отбросить.

Если такие дублирующие функции все же существуют, выбор из них при вызове выполняет двухпроходный алгоритм. На первом проходе для поиска совпадения используется тот же алгоритм, что и при разрешении функций в других базах данных (не Unicode). Если совпадение не обнаружено, будет выполнен второй проход с учетом следующих процедур расширения для строк CHAR и GRAPHIC: GRAPHIC->CHAR->VARGRAPHIC->VARCHAR->LONG VARGRAPHIC->LONG VARCHAR->DBCLOB->CLOB

Преобразование типов данных

К списку с заголовком "The following casts involving distinct types are supported" добавлена следующая запись:

- для базы данных Unicode преобразование из типа VARCHAR или VARGRAPHIC в пользовательский тип DT с исходным типом данных CHAR и GRAPHIC.

Далее приводятся изменения к таблице 6 *Supported Casts between Built-in Data Types*. (Приведены только измененные строки таблицы.)

Таблица 16. Поддерживаемые преобразования между встроенными типами данных

Конечный тип данных →	C H A R	V A R C H A R	L O N G V A R C H A R	C L O B	G R A P H I C	V A R G R A P H I C	L O N G V A R G R A P H I C	D B C L O B
Исходный тип данных ↓								
CHAR	Y	Y	Y	Y	Y ¹	Y ¹	-	-
VARCHAR	Y	Y	Y	Y	Y ¹	Y ¹	-	-
LONGVARCHAR	Y	Y	Y	Y	-	-	Y ¹	Y ¹
CLOB	Y	Y	Y	Y	-	-	-	Y ¹

Таблица 16. Поддерживаемые преобразования между встроенными типами данных (продолжение)

Конечный тип данных →	C H A R	V A R C H A R	L O N G V A R C H A R	C L O B	G R A P H I C	V A R G R A P H I C	L O N G V A R G R A P H I C	D B C L O B
Исходный тип данных ↓			C H A R					
GRAPHIC	Y ¹	Y ¹	-	-	Y	Y	Y	Y
VARGRAPHIC	Y ¹	Y ¹	-	-	Y	Y	Y	Y
LONGVARGRAPHIC	-	-	Y ¹	Y ¹	Y	Y	Y	Y
DBCLOB	-	-	-	Y ¹	Y	Y	Y	Y

¹ Преобразование типов поддерживается только для баз данных Unicode.

Назначения и сравнения

Назначения и сравнения, включающие в себя символьные и графические данные поддерживаются только тогда, когда одна из строк является литералом. Для разрешения функций и графические, и символьные литералы будут соответствовать и символьным, и графическим параметрам функций.

Далее приводятся изменения в Таблице 7 *Data Type Compatibility for Assignments and Comparisons*. (Приведены только измененные строки таблицы и новое примечание ⁶ к этой таблице.)

Операнды	Целое двоичное	Десятичное число	Плавающее	Символьная строка	Графическая строка	Длина	Время	Отметка времени	Двоичная строка	Пользовательский тип
Символьная строка	Нет	Нет	Нет	Да	Да ⁶	¹	¹	¹	Нет ³	²
Графическая строка	Нет	Нет	Нет	Да ⁶	Да	Нет	Нет	Нет	Нет	²

⁶ Поддерживается только для баз данных Unicode.

Назначения строк:

Storage Assignment

Последний абзац этого подраздела изменен так:

Когда символьная строка назначается столбцу фиксированной длины и длина строки меньше атрибута длины потребителя, эта строка справа дополняется необходимым числом пробелов (однобайтных, двухбайтных или UCS-2²). Заполняющий символ - всегда пробел даже для столбцов, определенных с атрибутом FOR BIT DATA.

Retrieval Assignment

Третий абзац этого подраздела изменен так:

Когда символьная строка назначается переменной фиксированной длины и длина строки меньше атрибута длины потребителя, строка дополняется справа необходимым числом пробелов (однобайтных, двухбайтных или UCS-2²). Заполняющий символ - всегда пробел даже для строк, определенных с атрибутом FOR BIT DATA.

² В UCS-2 определено несколько символов SPACE с разными свойствами. Для базы данных Unicode менеджер баз данных в качестве пробела UCS-2 всегда использует ASCII SPACE с кодом x'0020'. Для базы данных EUC для заполнения строк GRAPHIC используется IDEOGRAPHIC SPACE с кодом x'3000'.

Conversion Rules for String Assignments

В конец этого подраздела был добавлен следующий абзац:

Для баз данных Unicode символьные строки могут быть назначены графическому столбцу, а графические строки - символьному столбцу.

DBCS Considerations for Graphic String Assignments

Первый абзац этого подраздела изменен следующим образом:

Назначения графических строк обрабатываются аналогично назначениям символьных строк. Для баз данных не Unicode типы данных графических строк совместимы только с другими типами данных графических строк и никогда не совместимы с типами данных численных и символьных строк, а также с типами данных даты и времени. Для баз данных Unicode типы данных графических строк совместимы с типами данных символьных строк.

Сравнение строк:

Conversion Rules for Comparison

Этот подраздел изменен так:

При сравнении двух строк, если это необходимо, одна из них сначала преобразуется в схему кодировки и/или кодовую страницу другой строки. Подробности смотрите в разделе "Rules for String Conversions" главы 3 Language Elements справочника *SQL Reference*.

Правила для типов данных результатов

Символьные и графические строки в базе данных Unicode: Это новый подраздел, вставленный после подраздела "Graphic Strings".

В базе данных Unicode символьные и графические строки совместимы.

Если один из операндов...	А другой операнд...	Тип данных результата будет...
GRAPHIC(x)	CHAR(y) или GRAPHIC(y)	GRAPHIC(z), где z = max(x,y)
VARGRAPHIC(x)	CHAR(y) или VARCHAR(y)	VARGRAPHIC(z), где z = max(x,y)
VARCHAR(x)	GRAPHIC(y) или VARGRAPHIC	VARGRAPHIC(z), где z = max(x,y)

Если один из операндов...	А другой операнд...	Тип данных результата будет...
LONG VARGRAPHIC	CHAR(y) или VARCHAR(y) или LONG VARCHAR	LONG VARGRAPHIC
LONG VARCHAR	GRAPHIC(y) или VARGRAPHIC(y)	LONG VARGRAPHIC
DBCLOB(x)	CHAR(y) или VARCHAR(y) или CLOB(y)	DBCLOB(z), где $z = \max(x,y)$
DBCLOB(x)	LONG VARCHAR	DBCLOB(z), где $z = \max(x,16350)$
CLOB(x)	GRAPHIC(y) или VARGRAPHIC(y)	DBCLOB(z), где $z = \max(x,y)$
CLOB(x)	LONG VARGRAPHIC	DBCLOB(z), где $z = \max(x,16350)$

Rules for String Conversions

К следующему списку в этом разделе был добавлен третий пункт:

Для каждой пары кодовых страниц результат определяется последовательным применением следующих правил:

- Если кодовые страницы одинаковые, результатом является эта кодовая страница.
- Если одна из кодовых страниц - BIT DATA (кодовая страница 0), результатом является кодовая страница BIT DATA.
- В базе данных Unicode, если одна из кодовых страниц обозначает данные в схеме кодирования, отличной от другой страницы, результатом является UCS-2 по UTF-8 (то есть тип графических данных по типу символьных данных).¹
- В противном случае результирующая кодовая страница определяется в Таблице 8 раздела "Rules for String Conversions" Главы 3 Language Elements справочника *SQL Reference*. 'First' в этой таблице означает выбор кодовой страницы первого операнда, а 'second' - второго операнда.

¹ В базах данных не Unicode преобразование между разными схемами кодирования не поддерживается.

Expressions

Добавлено следующее:

В базе данных Unicode выражение, принимающее графическую или символьную строку, будет принимать любые типы строк, для которых поддерживается преобразование.

With the Concatenation Operator: В конец этого подраздела было добавлено следующее:

В базе данных Unicode конкатенация, включающая как операнды символьных строк, так и операнды графических строк, сперва будет преобразовывать символьные операнды в графические операнды. Обратите внимание на то, что в других базах данных (не Unicode) конкатенация не может включать и символьные, и графические операнды.

Predicates

К списку, начинающемуся с предложения "The following rules apply to all types of predicates", добавлено:

- В базе данных Unicode все предикаты, принимающие графическую или символьную строку, будет принимать любые типы строк, для которых поддерживается преобразование.

Глава 4. Functions

Scalar Functions

В конец этого раздела добавлено следующее предложение:

В базе данных Unicode все скалярные функции, принимающие графическую или символьную строку, будет принимать любые типы строк, для которых поддерживается преобразование.

CLI Guide and Reference

Глава 3. Using Advanced Features

Дальше приведен новый раздел для этой главы.

Написание прикладной программы Unicode CLI DB2

Существуют две основные области поддержки для прикладных программ Unicode CLI DB2:

1. Добавление набора функций, который может принимать строковые аргументы Unicode вместо строковых аргументов ANSI.
2. Добавление новых типов данных C и SQL для описания данных как данных ANSI или Unicode. В следующих разделах приводится дополнительная информация для обеих этих областей. Чтобы программа считалась прикладной программой Unicode, перед выполнением соединения он должна задать для атрибута соединения SQL_ATTR_ANSI_APP значение SQL_AA_FALSE. Этим достигается то, что CLI будет связываться как клиент Unicode, а все данные Unicode будут посылаться либо в UTF-8 для данных CHAR, либо в UCS-2 для данных GRAPHIC.

Функции Unicode: Далее приводится список функций API ODBC, которые поддерживают версии как Unicode (W), так ANSI (A) (в Unicode в имени функции будет присутствовать W):

SQLBrowseConnect	SQLForeignKeys	SQLPrimaryKeys
SQLColAttribute	SQLGetConnectAttr	SQLProcedureColumns
SQLColAttributes	SQLGetConnectOption	SQLProcedures
SQLColumnPrivileges	SQLGetCursorName	SQLSetConnectAttr
SQLColumns	SQLGetDescField	SQLSetConnectOption
SQLConnect	SQLGetDescRec	SQLSetCursorName
SQLDataSources	SQLGetDiagField	SQLSetDescField
SQLDescribeCol	SQLGetDiagRec	SQLSetStmtAttr
SQLDriverConnect	SQLGetInfo	SQLSpecialColumns
SQLDrivers	SQLGetStmtAttr	SQLStatistics
SQLError	SQLNativeSQL	SQLTablePrivileges
SQLExecDirect	SQLPrepare	SQLTables

Функции Unicode всегда возвращают или принимают аргументы длины строки в виде числа символов. Для функций, возвращающих информацию о длине для данных сервера, размер и точность вывода на экран описываются числом символов. Когда длина (размер передаваемых данных) может относиться и к строковым, и к другим данным, эта длина описывается в длинах октетов. Например, для SQLGetInfoW длина - это число байтов, а для SQLExecDirectW - число символов. CLI будет возвращать наборы результатов либо в Unicode, либо в ANSI, в зависимости от связывания прикладной программы. Если прикладная программа связывается с SQL_C_CHAR, драйвер будет преобразовывать данные SQL_WCHAR в SQL_CHAR. Менеджер

драйверов отображает SQL_C_WCHAR в SQL_C_CHAR для драйверов ANSI, но не производит отображения для драйверов Unicode.

Новые типы данных и действительные преобразования: Есть два новых определенных типа данных CLI или ODBC, SQL_C_WCHAR и SQL_WCHAR. SQL_C_WCHAR указывает, что буфер C содержит данные UCS-2. SQL_WCHAR указывает, что маркер конкретного столбца или параметра содержит данные Unicode. Для серверов Unicode DB2 графические столбцы будут описываться как SQL_WCHAR. Преобразование будет разрешено между SQL_C_WCHAR и SQL_CHAR, SQL_VARCHAR, SQL_LONGVARCHAR и SQL_CLOB, а также между графическими типами данных.

Таблица 17. Поддерживаемые преобразования данных

Тип данных SQL	SQL_C_CHAR	SQL_C_CHAR_UNICODE																	
BLOB	X	X									D					X			
CHAR	D	X	X	X	X	X	X	X	X	X	X	X						X	X
CLOB	D	X									X			X					
DATE	X	X						D		X									
DBCLOB		X									X		D				X		
DECIMAL	D	X	X	X	X	X	X				X	X						X	X
DOUBLE	X	X	X	X	X	X	D					X						X	X
FLOAT	X	X	X	X	X	X	D					X						X	X
GRAPHIC (He Unicode)	X	X											D						
GRAPHIC (Unicode)	X	X	X	X	X	X	X	X	X	X	X	X	D					X	
INTEGER	X	X	D	X	X	X	X						X					X	X
LONG VARCHAR	D	X										X							
LONG VARGRAPHIC (He Unicode)	X	X										X		D					
LONG VARGRAPHIC (Unicode)	X	X										X		D					

Таблица 17. Поддерживаемые преобразования данных (продолжение)

Тип данных SQL	SQL_C_CHAR	SQL_C_WCHAR	SQL_C_LONG	SQL_C_SHORT	SQL_C_TINYINT	SQL_C_FLOAT	SQL_C_DOUBLE	SQL_C_TYPE_DATE	SQL_C_TYPE_TIME	SQL_C_TYPE_TIMESTAMP	SQL_C_BINARY	SQL_C_BIT	SQL_C_DBCHAR	SQL_C_CHAR_LOB	SQL_C_CHAR_LLOCATOR	SQL_C_CHAR_LLOB	SQL_C_CHAR_LLOB_LLOCATOR	SQL_C_BIGINT	SQL_C_NUMERIC	
NUMERIC	D	X	X	X	X	X	X					X								X
REAL	X	X	X	X	X	D	X					X								X
SMALLINT	X	X	X	D	X	X	X					X						X	X	
BIGINT	X	X	X	X	X	X	X				X	X						D	X	
TIME	X	X							D	X										
TIMESTAMP	X	X						X	X	D										
VARCHAR	D	X	X	X	X	X	X	X	X	X	X	X							X	X
VARGRAPHIC (He Unicode)	X	X											D							
VARGRAPHIC (Unicode)	X	X	X	X	X	X	X	X	X	X	X	X	D					X		

Примечание:

- D** Преобразование поддерживается. Это - используемое по умолчанию преобразование для этого типа данных SQL.
- X** Все СУБД IBM поддерживают это преобразование.
- пусто** Никакие СУБД IBM не поддерживают это преобразование.
 - Данные не преобразуются в типы локатора больших объектов; наоборот, локаторы представляют значение данных; дополнительную информацию смотрите в разделе Использование больших объектов.
 - SQL_C_NUMERIC доступен только в 32-битных операционных системах Windows.

Устаревшее значение ключевого слова для исправления: До того, как стали поддерживаться прикладные программы Unicode, программы, написанные для работы с однобайтными символьными данными, можно было заставить работать с двухбайтными графическими данными при помощи нескольких последовательных

ключевых слов файлов `cli.ini`, таких как `GRAPHIC=1,2` или `3`, `Patch2=7` и т. д. Такой подход представляет графические данные как символьные и оказывает влияние на сообщаемую длину данных.

Для прикладных программ Unicode такие ключевые слова теперь не нужны, и, более того, они не должны использоваться, поскольку могут вызывать серьезные побочные эффекты. Если достоверно не известно, является ли конкретная прикладная программа программой Unicode, советуем попробовать обойтись без ключевых слов, влияющих на обработку графических данных.

Литералы в базах данных Unicode: В базах данных не Unicode нельзя сравнивать данные в столбцах `LONG VARGRAPHIC` и `LONG VARCHAR`. Столбцы `GRAPHIC/VARGRAPHIC` и `CHAR/VARCHAR` можно только сравнивать или назначать друг другу, явным образом используя функции преобразования типов, поскольку неявное преобразование кодовых страниц не поддерживается как таковое. Это относится к литералам `GRAPHIC/VARGRAPHIC` и `CHAR/VARCHAR`, когда литерал `GRAPHIC/VARGRAPHIC` отличается от литерала `CHAR/VARCHAR` при помощи префикса `G`.

В базах данных Unicode не требуется преобразование между литералами `GRAPHIC/VARGRAPHIC` и `CHAR/VARCHAR`. Кроме того, в них перед литералом `GRAPHIC/VARGRAPHIC` не требуется префикс `G`. Если, по крайней мере, один из аргументов является литералом, происходят неявные преобразования. Это позволяет применять литералы с префиксом `G` или без него в операторах, использующих либо `SQLPrepareW()`, либо `SQLExecDirect()`. При этом литералы для `LONG VARGRAPHIC` могут сохранять префикс `G`.

Дополнительную информацию смотрите в разделе "Casting Between Data Types" Главы 3 Language Elements справочника *SQL Reference*.

Новые ключевые слова для конфигурации CLI: С целью устранения дополнительных издержек при связи прикладных программ Unicode с базой данных были добавлены следующие три ключевых слова.

1. `DisableUnicode`

Описание ключевого слова:

Отключает базовую поддержку Unicode.

Синтаксис ключевого слова `db2cli.ini`:

`DisableUnicode = 0 | 1`

Значение по умолчанию:

0 (ложь)

Закладка Параметры CLI/ODBC DB2:

Это ключевое слово нельзя задать через записную книжку Параметры CLI/ODBC. Для использования этого ключевого слова надо непосредственно изменить файл `db2cli.ini`.

Замечания по использованию

Если сначала разрешить поддержку Unicode, а затем вызвать прикладную программу Unicode, CLI попытается связаться с базой данных и использованием кодовой страницы клиента, лучше всех обеспечивающей отсутствие дополнительных потерь данных, связанных с преобразованием кодовых страниц. Это может увеличить время соединения за счет обмена кодовых страниц, а также вызвать преобразование кодовых страниц на клиенте, которого не было до добавления этой поддержки.

Задание значения True для этого ключевого слова значения приведет к преобразованию всех данных Unicode в локальную кодовую страницу прикладной программы перед их отправкой на сервер. Это может привести к потере любых данных, которые нельзя представить в локальной кодовой странице.

2. ConnectCodepage

Описание ключевого слова:

Указывает конкретную кодовую страницу для использования при связи с источником данных для минимизации расходов соединения.

Синтаксис ключевого слова db2cli.ini:

ConnectCodepage = 0 | 1 <любая действительная кодовая страница db2>

Значение по умолчанию:

0

Закладка Параметры CLI/ODBC DB2:

Это ключевое слово нельзя задать через записную книжку Параметры CLI/ODBC. Для использования этого ключевого слова надо непосредственно изменить файл db2cli.ini.

Замечания по использованию

Прикладные программы не Unicode всегда связываются с базой данных, используя локальную страницу прикладной программы или установку среды кодовой страницы DB2. По умолчанию CLI будет обеспечивать связь прикладных программ Unicode с базами данных Unicode с использованием кодовых страниц UTF-8 и UCS-2, а с базами данных не Unicode - с использованием кодовых страниц этих баз данных. Этим предотвращается необязательная потеря данных из-за преобразования кодовых страниц.

Это ключевое слово позволяет пользователю указывать кодовую страницу базы данных при связывании с базой данных не Unicode, чтобы при этом соединении не производились ненужные дополнительные затраты.

Укажите значение 1, чтобы SQLDriverConnect() возвращало правильное значение в строке вывода соединения, чтобы использовать это значение при последующих вызовах SQLDriverConnect().

3. UnicodeServer

Описание ключевого слова:

Указывает, что источником данных является сервер Unicode.
Эквивалентно значению ConnectCodepage=1208.

Синтаксис ключевого слова db2cli.ini:

UnicodeServer = 0 | 1

Значение по умолчанию:

0

Закладка Параметры CLI/ODBC DB2:

Это ключевое слово нельзя задать через записную книжку Параметры CLI/ODBC. Для использования этого ключевого слова надо непосредственно изменить файл db2cli.ini.

Замечания по использованию

Это ключевое слово эквивалентно ConnectCodepage=1208 и добавлено только для удобства. Установите это ключевое слово для предотвращения избыточных расходов на связь с DB2 for OS/390 Версии 7 или более поздней. Нет

необходимости устанавливать это ключевое слово для баз данных DB2 for Windows, DB2 for Unix или DB2 for OS/2, поскольку при этом не требуется дополнительная обработка.

Приложение C. DB2 CLI and ODBC

Дальше приведен новый раздел для этого приложения.

Прикладные программы Unicode ODBC

Прикладные программы ODBC Unicode отправляют и получают символьные данные главным образом в формате UCS-2. Это происходит путем вызова Unicode-версий функций ODBC (с суффиксом 'W') и указания типов данных Unicode. Прикладная программа не указывает локальную кодовую страницу явно. Она сохраняет способность вызывать функции ANSI и передавать строки в кодировке локальной кодовой таблицы.

Например, прикладная программа может вызвать `SQLConnectW()` и передать ID пользователя и пароль DSN как аргументы Unicode. После этого она может вызвать `SQLExecDirectW()` и передать строку оператора SQL Unicode, а затем связать комбинацию буферов страниц ANSI с локальной кодировкой (`SQL_C_CHAR`) и буферов Unicode (`SQL_C_WCHAR`). Типом данных базы данных может быть локальная кодовая страница или UCS-2 и UTF-8.

Если прикладная программа CLI вызывает `SQLConnectW` или вызывает `SQLSetConnectAttr` со значением `SQL_AA_FALSE` для `SQL_ATTR_ANSI_APP`, эта прикладная программа считается программой Unicode. Это означает, что все данные CHAR посылаются базе данных и принимаются от нее в формате UTF-8. После этого прикладная программа может считывать данные CHAR в буфера `SQL_C_CHAR` в кодировке локальной кодовой страницы (с возможной потерей данных) или в буфера `SQL_C_WCHAR` в UCS-2 без какой-либо потери данных.

Если прикладная программа не выполняет любой из описанных выше вызовов, данные CHAR на сервере преобразуются в локальную страницу прикладной программы. Это означает, что в данных CHAR, считанных в `SQL_C_WCHAR`, возможны потери.

Если для переменной экземпляра `DB2CODEPAGE` установлена (с использованием `db2set`) кодовая страница 1208 (UTF-8), прикладная программа будет получать все данные CHAR в UTF-8, поскольку теперь это новая локальная страница. Прикладная программа должна также проверить, что все входные данные CHAR тоже находятся в UTF-8. ODBC подразумевает также, что все данные `SQL_C_WCHAR` находятся в собственном конечном формате. CLI будет выполнять все необходимые обращения байтов для `SQL_C_WCHAR`.

Сравнение прикладных программ ODBC Unicode и не Unicode: В этом выпуске DB2 Universal Database есть API `SQLConnectW()`. Драйвер Unicode должен экспортировать `SQLConnectW`, чтобы быть опознанным драйвером менеджеров как драйвер Unicode. Важно заметить, что многие из прикладных программ ODBC (такие как Microsoft Access и Visual Basic) вызывают `SQLConnectW()`. В предыдущих выпусках DB2 Universal Database CLI DB2 не поддерживали этот API, и поэтому он не распознавался менеджером драйверов ODBC как драйвер Unicode. Это приводило к тому, что менеджер драйверов ODBC преобразовывал все драйверы Unicode в локальную кодовую страницу прикладной программы. При добавлении поддержки функции `SQLConnectW()` эти прикладные программы не будут устанавливать связь как программы Unicode, а все необходимые преобразования данных возьмет на себя CLI DB2.

Теперь CLI DB2 принимает API Unicode API (с суффиксом "W") и стандартные API ANSI. ODBC определяет набор функций с суффиксом "A", но менеджер драйверов не пропускает к драйверу функции ANSI с суффиксом "A". Вместо этого он преобразует эти функции в вызовы функций ANSI без этого суффикса, а затем передает их драйверу.

Прикладная программа ODBC, вызывающая API SQLConnectW(), считается прикладной программой Unicode. Так как менеджер драйверов ODBC всегда будет вызывать API SQLConnectW() независимо от версии вызываемой прикладной программы, ODBC теперь использует атрибут связи SQL_ATTR_ANSI_APP для уведомления драйвера о том, что она должна рассматриваться как прикладная программа ANSI или UNICODE. Если для SQL_ATTR_ANSI_APP не установлено SQL_AA_FALSE, перед отправкой его на сервер CLI DB2 преобразует все данные Unicode в локальную кодировку страницы.

Data Movement Utilities Guide and Reference

Приложение C. Export/Import/Load Utility File Formats

В это приложение внесено следующее дополнение:

Утилиты экспорта, импорта и загрузки не поддерживаются, когда они используются с клиентом Unicode, связанным с базой данных не Unicode. Файлы клиента Unicode поддерживаются только тогда, когда клиент Unicode связан с базой данных Unicode.

Часть 8. Соединение с хост-системами

Глава 32. Дополнение по возможностям соединений	377
Настройка сервера прикладных программ в среде VM.	377
Параметры PATCH1 и PATCH2 и конфигурации CLI/ODBC/JDBC	377

Глава 32. Дополнение по возможностям соединений

Настройка сервера прикладных программ в среде VM

Добавьте следующее предложение после первого (и единственного) предложения в разделе "Задание сетевой информации" в подразделе "Определение сервера прикладных программ":

В качестве имени параметра DBNAME в SQLSTART EXEC задается RDB_NAME.

Параметры PATCH1 и PATCH2 и конфигурации CLI/ODBC/JDBC

Для конфигурирования драйвера CLI/ODBC/JDBC можно использовать Ассистент конфигурирования клиента или менеджер драйверов ODBC (если он установлен в системе) или можно вручную отредактировать файл `db2cli.ini`. Более подробную информацию смотрите в руководстве *Дополнение по установке и конфигурированию* или *CLI Guide and Reference*.

Поведение по умолчанию драйвера CLI/ODBC DB2 можно изменить, задав значения обоих ключевых слов PATCH1 и PATCH2 (в файле `db2cli.ini` или с помощью функции `API CLI SQLDriverConnect()` или `SQLBrowseConnect()`).

Задаваемое значение ключевого слова PATCH1 - сумма всех значений, которые нужно задать для него. Например, чтобы задать значения 1, 2 и 8, нужно задать для PATCH1 значение 11. Ниже представлено описание каждого значения и его влияние на драйвер:

- 1 - драйвер будет выполнять поиск "count(exp)", заменяя его на "count(distinct exp)". Это необходимо, поскольку некоторые версии DB2 поддерживают синтаксис "count(exp)" и этот синтаксис генерируется некоторыми прикладными программами ODBC. Требуется прикладным программам Microsoft, если сервер не поддерживает синтаксис "count(exp)".
- 2 - Работа некоторых прикладных программ ODBC прерывается, если возвращается SQL_NULL_DATA в функции `SQLGetTypeInfo()` для столбца `LITERAL_PREFIX` или `LITERAL_SUFFIX`. Это значение указывает драйверу, что вместо этого нужно возвращать пустую строку. Требуется для `Impromptu 2.0`.
- 4 - Указывает драйверу, что входные данные типа отметка времени должны интерпретироваться как данные типа дата, если часть значения отметки времени, определяющая значение времени, равна нулю. Требуется для `Microsoft Access`.
- 8 - Указывает драйверу, что входные данные типа отметка времени должны интерпретироваться как данные типа время, если часть значения отметки времени, определяющая дату, равна 1899-12-30. Требуется для `Microsoft Access`.
- 16 - Не используется.
- 32 - Указывает драйверу, чтобы он не возвращал информацию о столбцах `SQL_LONGVARCHAR`, `SQL_LONGVARIABLE` и `SQL_LONGVARGRAPHIC`. Для прикладной программы это выглядит так, как будто длинные поля не поддерживаются. Требуется для `Lotus 123`.
- 64 - Указывает драйверу, чтобы он завершал пустым символом выходные графические строки. Требуется для `Microsoft Access` в двухбайтной среде.
- 128 - Указывает драйверу, что он должен передавать на сервер запросы "SELECT Config, nValue FROM MSysConf". В настоящее время драйвер возвращает код ошибки со значением `SQLSTATE S0002` (таблица не найдена). Требуется, если пользователь создал в базе данных эту таблицу конфигурации и хочет, чтобы прикладная программа обращалась к ней.
- 256 - Указывает драйверу, что в вызове `SQLStatistics()` он должен первыми возвращать столбцы первичных ключей. В настоящее время драйвер возвращает индексы отсортированными по именам индексов (это стандартное поведение ODBC).
- 512 - Указывает драйверу, что он должен возвращать FALSE в `SQLGetFunctions()` для `SQL_API_SQLTABLEPRIVILEGES` и `SQL_API_SQLCOLUMNPRIVILEGES`.
- 1024 - Указывает драйверу, что он должен возвращать `SQL_SUCCESS` вместо `SQL_NO_DATA_FOUND` в `SQLExecute()` или `SQLExecDirect()`, если выполненный оператор `UPDATE` or `DELETE`

- не изменил ни одной строки. Требуется для прикладных программ Visual Basic.
- 2048 - Не используется.
- 4096 - Указывает драйверу, что он не должен выдавать команду COMMIT после закрытия указателя в режиме автоматического принятия.
- 8192 - Указывает драйверу, что он должен возвращать дополнительный набор результатов после вызова хранимой процедуры. Этот набор результатов содержит одну строку, состоящей из выходных значений этой хранимой процедуры. Может использоваться прикладными программами Powerbuild.
- 32768 - Указывает драйверу, что он должен дать программам Microsoft Query работать с синонимами DB2 MVS.
- 65536 - Указывает драйверу, что он должен сам вставлять "G" перед символьными литералами, которые на самом деле являются графическими литералами. Нужно всегда задавать это значение при работе в двухбайтной среде.
- 131072 - Указывает драйверу, что он должен описывать входящий в индекс уникальности столбец с типом отметка времени как столбец типа CHAR(26). Требуется для прикладных программ Microsoft.
- 262144 - Указывает драйверу, что нужно использовать таблицу псевдокаталога db2cli.procedures вместо таблиц SYSCAT.PROCEDURES и SYSCAT.PROCPARMS.
- 524288 - Указывает драйверу, что нужно использовать SYSTEM_TABLE_SCHEMA вместо TABLE_SCHEMA при выполнении запроса к системной таблице в системе DB2/400 V3.x. Это повышает производительность.
- 1048576 - Указывает драйверу, что строку нулевой длины для SQLPutData() нужно интерпретировать как SQL_NULL_DATA.

Ключевое слово PATCH2 отличается от ключевого слова PATCH1. В нем несколько значений задаются через запятые. Например, чтобы задать значения 1, 4 и 5, нужно задать для PATCH2 значение "1,4,5". Ниже представлено описание каждого значения и его влияние на драйвер:

- 1 - Указывает драйверу, что он должен перевести имя хранимой процедуры в операторе CALL в верхний регистр.
- 2 - Не используется.
- 3 - Указывает драйверу, что он должен переводить все аргументы в вызовах схемы в верхний регистр.
- 4 - Указывает драйверу, что он должен возвращать для вызовов схем (то есть SQLColumns(), SQLProcedureColumns() и т.д.) набор результатов в формате Версии 2.1.2 вместо формата Версии 5.
- 5 - Указывает драйверу, что не нужно оптимизировать обработку входных столбцов VARCHAR, если указатель на данные и указатель на длину идут в памяти подряд.
- 6 - Указывает драйверу, что он должен возвращать сообщение, что указатели с прокруткой не поддерживаются. Требуется для программ Visual Basic, если используются клиент DB2 Версии 5 и сервер DB2 UDB Версии 5.
- 7 - Указывает драйверу, что он должен отображать все типы данных столбцов GRAPHIC на типы данных столбцов CHAR. Требуется в двухбайтной среде.
- 8 - Указывает драйверу, что он должен игнорировать аргументы поиска каталогов в вызовах схем.
- 9 - Не выполнять принятие при раннем закрытии указателя.
- 10 - Не используется.
- 11 - Сообщать, что имя каталога поддерживается, (хранимые процедуры VB)
- 12 - Удалять двойные кавычки из аргументов вызовов схем, (Visual Interdev)
- 13 - Не добавлять ключевые слова из db2cli.ini к выходной строке соединения
- 14 - Игнорировать имя схемы в SQLProcedures() и SQLProcedureColumns()
- 15 - Всегда использовать точку как разделитель дробной части при символьном выводе
- 16 - Принудительно возвращать информацию описания при каждом открытии
- 17 - Не возвращать имена столбцов в описании
- 18 - Пытаться заменять литералы маркерами параметров
- 19 - В настоящее время DB2 MVS V4.1 не поддерживает синтаксис ODBC, в котором разрешены скобки в условии ON в условии внешнего объединения. Если в PATCH2 задано это значение, драйвер ODBC IBM DB2 будет удалять скобки, если условие внешнего объединения входит в управляющую последовательность ODBC. Это значение следует использовать только при работе с DB2 MVS 4.1.
- 20 - В настоящее время DB2 on MVS не поддерживает предикат BETWEEN с маркерами параметров в качестве обоих операндов (выражение ? BETWEEN ?). Если задано это значение, драйвер ODBC IBM переписет такой предикат в виде (выражение >= ? and выражение <= ?).
- 21 - Задать все только выходные (OUTPUT) параметры для хранимых процедур как SQL_NULL_DATA
- 22 - Если задано это значение, драйвер ODBC IBM будет сообщать, что внешнее (OUTER) объединение не поддерживается. Предназначено для прикладных программ, генерирующих SELECT DISTINCT col1 или ORDER BY col1 при использовании оператора внешнего объединения, где col1 имеет длину больше 254 символов, в результате DB2 UDB возвращает сообщение об ошибке (поскольку DB2 UDB не поддерживает в таких конструкциях столбцы длиной более 254 байт)

- 23 - Не оптимизировать ввод для параметров, связанных с cbColDef=0
- 24 - Обходной прием для работы со значениями времени - они представляются в символьном виде
- 25 - Обходной прием для работы с десятичными столбцами - из их символьного представления удаляются конечные нули
- 26 - Не возвращать прикладной программе sqlcode 464 - указывает, что возвращены наборы результатов
- 27 - Использовать для SQLTables значение ключевого слова TABLETYPE, даже если прикладная программа задает правильное значение
- 28 - Описывает столбцы real как столбцы double
- 29 - Исправление работы ADO для десятичных столбцов - удаляет начальные нули для значений x, где $1 > x > -1$
(Требуется только для некоторых версий MDAC)
- 30 - Запрещает оптимизацию кэширования хранимых процедур
- 31 - Сообщать данные статистики для алиасов в вызове SQLStatistics
- 32 - Переопределяет обработку кода причины 4 для sqlcode -727
- 33 - Возвращать преобразованное в символьную форму значение отметки времени в формате ISO
(а не в формате ODBC)
- 34 - Сообщать значение CHAR для столбцов CHAR FOR BIT DATA
- 35 - Сообщать о неверном TABLENAME, если запрошено
SQL_DESC_BASE_TABLE_NAME - оптимизация только для чтения ADO
- 36 - Зарезервировано
- 37 - Зарезервировано

Часть 9. Общие сведения

Глава 33. Общие сведения	383	Доступ к другим базам данных (не из семейства DB2)	403
DB2 Universal Database Business Intelligence Quick Tour	383	Запуск утилит DB2 for OS/390	404
DB2 Everywhere называется теперь DB2 Everyplace	383	Репликация	406
Сообщения об ошибках при попытке запуска Netscape	384	Регистрация агента	407
Необходима мышь	384	Кэширование на клиенте в Windows NT	408
Требования к конфигурации для Adobe Acrobat Reader в системах на основе UNIX	385	Пробные программные продукты на компакт-дисках с Enterprise Edition для UNIX	408
При попытке связывания из клиента времени выполнения DB2 возникает ошибка "Не найдены файлы связывания"	385	Пробные программные продукты на компакт-дисках с DB2 Connect Enterprise Edition для UNIX	409
Поиск SEARCH	385	Отбрасывание Менеджера связей данных	409
Окна памяти для HP-UX 11	385	Деинсталляция компонентов DLFM с использованием SMIT может удалить дополнительные наборы файлов	410
Справочник SQL поставляется в виде одного файла PDF	386	Ошибка SQL1035N при использовании CLP в Windows 2000	410
Действия пользователя при возникновении ошибки dlfm client_conf	387	Усовершенствование SQL Assist	410
В редких случаях, когда демон копирования не останавливается командой dlfm stop	387	Сообщение об ошибке Windows Millennium Edition (ME) и поисковый сервер NetQuestion	410
Деинсталляция Client Enabler DFS DB2	387		
Аутентификация клиента в Windows NT	387		
AutoLoader может зависать при ветвлении потоков	388		
Восстановление столбцов DATALINK	388		
Определите ID пользователя и пароль в IBM Communications Server для Windows NT (CS/NT)	388		
Определение узла	388		
Ограничения системы объединения	389		
Ограничение DataJoiner	389		
Менеджер каталогов данных на иврите для Windows NT	389		
Microsoft SNA Server и поддержка многоузлового изменения SNA (двухфазного принятия)	390		
Ошибка запуска SPM SNA DB2 после загрузки Windows	390		
Национальные настройки для сервера администратора DB2	391		
Клавиши быстрого вызова не работают	391		
Требования к учетным записям для DB2 в Windows NT и Windows 2000	391		
Потеря привилегии EXECUTE для пользователей Query Patroller, созданных в версии 6	392		
Ограничения Query Patroller	392		
Необходимость принятия всех пользовательских программ, которые будут использованы в Центре хранилищ данных (Data Warehouse Center - DWC)	392		
Новая опция команды экспорта Центра хранилищ данных	392		
API служб резервного копирования (XBSA)	393		
Агент OS/390	393		
Обзор установки	394		
Подробности установки	394		
Настройка дополнительных функций агента	397		
Составление расписания шагов хранилища при помощи программы триггера (XTClient)	398		
Преобразователи	399		

Глава 33. Общие сведения

DB2 Universal Database Business Intelligence Quick Tour

Quick Tour недоступен для DB2 for Linux или Linux/390.

Quick Tour оптимизирован для работы с мелкими системными шрифтами. Для правильного просмотра Quick Tour в OS/2 может потребоваться настройка размера шрифтов браузера Web. Информацию о том, как настроить размер шрифтов, смотрите в справке браузера Web. Для правильного просмотра Quick Tour при использовании однобайтных символов рекомендуется использовать шрифт Helv размером 8 пунктов. При использовании японского или корейского языка рекомендуется использовать шрифт Mincho размером 8 пунктов. Задавая параметры шрифтов, выберите в окне Предпочтения опцию "Использовать мои шрифты по умолчанию, переопределяя шрифты, заданные в документе".

В некоторых случаях Quick Tour может запускать второе окно браузера. Чтобы исправить эту ошибку, закройте Quick Tour и выполните шаги, описанные в разделе "Сообщения об ошибках при попытке запуска Netscape" на стр. 384.

При запуске Quick Tour вы можете получить сообщение об ошибке JavaScript, подобное следующему:

file:/C:/Program Files/SQLLIB/doc/html/db2qt/index4e.htm, строка 65:

Окно не определено.

Эта ошибка JavaScript не дает странице запуска Quick Tour index4e.htm автоматически закрыться после запуска Quick Tour. Страницу запуска Quick Tour можно закрыть, закрыв окно браузера, в котором выводится страница index4e.htm.

В разделе "Что нового" в теме Управление данными сказано, что в Версии 7.1 поддерживается архивирование журналов по требованию. Это неверно. Кроме того, сказано также:

Размер файлов журнала увеличен с 4 до 32 Гбайт.

Это предложение следует читать так:

Общее пространство активного журнала увеличено с 4 до 32 Гбайт.

В разделе о менеджере связей данных DB2 есть предложение:

Кроме того, теперь он поддерживает интерфейс XBSA фирмы Veritas для резервного копирования и восстановления при помощи NetBackup.

Это предложение следует читать так:

Кроме того, теперь он поддерживает интерфейс XBSA фирмы Veritas для архивирования и восстановления файлов. В число менеджеров памяти, поддерживающих интерфейс XBSA, входят Legato NetWorker и Veritas NetBackup.

DB2 Everywhere называется теперь DB2 Everyplace

Имя DB2 Everywhere было изменено на DB2 Everyplace.

Сообщения об ошибках при попытке запуска Netscape

Если при попытке запуска Netscape вы получили следующие сообщения об ошибках:

Не найден файл <путь файла> (или один из его компонентов).
Проверьте правильность пути и имени файла и убедитесь,
что доступны все необходимые библиотеки.

Не удалось открыть "D:\Program Files\SQLLIB\CC\..\doc\html\db2help\XXXXX.htm"

выполните следующие действия для исправления этой ошибки в системах Windows NT, 95 или 98 (что нужно сделать в Windows 2000, смотрите ниже):

1. В меню Пуск выберите Программы → Проводник. Откроется Проводник Windows.
2. В Проводнике Windows выберите Вид → Параметры. Откроется записная книжка Параметры.
3. Щелкните по закладке **Типы файлов**. Откроется страница Типы файлов.
4. В поле **Зарегистрированные типы** выберите Netscape Hypertext Document и нажмите кнопку **Изменить**. Откроется окно Изменение свойств типа.
5. В поле **Действия** выберите "open".
6. Нажмите кнопку **Изменить**. Откроется окно Изменение действия для типа.
7. Выключите переключатель **Использовать DDE**.
8. Убедитесь, что в самом конце строки в поле **Приложение, исполняющее действие** находится "%1" (включая кавычки и пробел перед первой кавычкой).

Если такие сообщения получены в системе Windows 2000, выполните следующие действия:

1. В меню **Пуск** выберите **Проводник Windows**. Откроется Проводник Windows.
2. В Проводнике Windows выберите **Средства** → **Параметры папок**. Откроется записная книжка Параметры папок.
3. Щелкните по закладке **Типы файлов**.
4. На странице Типы файлов в поле **Зарегистрированные типы** выберите HTML Netscape Hypertext Document и нажмите кнопку **Дополнительные**. Откроется окно Изменение свойств типа.
5. В поле **Действия** выберите "open".
6. Нажмите кнопку **Изменить**. Откроется окно Изменение действия для типа.
7. Выключите переключатель **Использовать DDE**.
8. Убедитесь, что в самом конце строки в поле **Приложение, исполняющее действие** находится "%1" (включая кавычки и пробел перед первой кавычкой).
9. Нажмите кнопку **ОК**.
10. Повторите шаги с 4 по 8 для типов файлов HTML Netscape Hypertext Document и SHTML Netscape Hypertext Document.

Необходима мышь

Для работы с этими инструментами на всех платформах, кроме Windows, требуется мышь.

Требования к конфигурации для Adobe Acrobat Reader в системах на основе UNIX

Для платформ на основе UNIX Acrobat Reader предлагается только для работы с английским языком, поэтому при попытке открыть файлы PDF с другими национальными языками могут возникнуть ошибки. В сообщениях об ошибках может говориться об ошибках доступа к шрифтам или об ошибках выделения содержимого файла PDF, но на самом деле эти ошибки вызваны тем, что английская версия Acrobat Reader может нормально работать в UNIX только в английской национальной среде.

Чтобы просмотреть такие файлы PDF, перед запуском английской версии Acrobat Reader переключитесь на английскую национальную среду, для чего выполните следующие действия:

- Отредактируйте файл сценария запуска Acrobat Reader, добавив в него после оператора `#!/bin/sh` следующую строку:

```
LANG=C;export LANG
```

Это обеспечит правильное поведение Acrobat Reader при его запуске другими прикладными программами (например, из Netscape Navigator или из меню справки прикладной программы).

- Введите в командной строке `LANG=C`, чтобы задать английскую среду для прикладной программы Acrobat Reader.

Чтобы получить дополнительную информацию, обратитесь в Adobe Systems (<http://www.Adobe.com>).

При попытке связывания из клиента времени выполнения DB2 возникает ошибка "Не найдены файлы связывания"

Поскольку у клиента времени выполнения DB2 нет полного набора файлов связывания, из него нельзя выполнить связывание средств GUI (его можно выполнить только из клиента администратора DB2).

Поиск SEARCH

Поиск SEARCH поддерживается только для широкополосной связи. Например, поиск SEARCH не будет работать через адаптер ATM. К поиску KNOWN это ограничение не относится.

Окна памяти для HP-UX 11

Окна памяти предназначены для пользователей больших 64-битных компьютеров HP, которые хотят воспользоваться преимуществами использования более чем 1,75 Гбайт совместно используемой памяти для 32-битных прикладных программ. Окна памяти не требуются, если вы работаете с 64-битной версией DB2. Окна памяти позволяют использовать для процесса или группы процессов отдельный 1 Гбайт совместно используемой памяти. Это позволяет экземпляру иметь свой собственный 1 Гбайт совместно используемой памяти плюс 0,75 Гбайта глобальной совместно используемой памяти. Если пользователи хотят использовать такие преимущества, они могут запустить несколько экземпляров, каждый в своем собственном окне памяти. Для использования окон памяти требуется:

- Среда DB2 EE
 - Исправления: Extension Software 12/98 и PHKL_17795.

- Для экземпляра должна быть задана переменная \$DB2INSTANCE.
- Для каждого экземпляра DB2, который нужно выполнять в окне памяти, должна быть запись в файле /etc/services.window. Например:

```
db2instance1 50
db2instance2 60
```

Примечание: Между именем и ID может быть только один пробел.

- Все команды DB2, которые нужно выполнить на сервере и для которых требуется несколько операторов, надо выполнять при помощи метода обратной связи TCP/IP. Это вызвано тем, что оболочка завершит работу, когда окно памяти закончит обработку первого оператора. Служба DB2 умеет справляться с этой ситуацией.
- Перед любой командой DB2, которая нужно выполнить для экземпляра, запущенного в окне памяти, нужно ввести db2win (имя команды, расположенной в каталоге sqllib/bin). Например:

```
db2win db2start
db2win db2stop
```

- Все команды DB2, выполняющиеся вне окна памяти (но когда окно памяти запущено), будут возвращать код 1042. Например:

```
db2win db2start <== 0K
db2 connect to db <==SQL1042
db2stop <==SQL1042
db2win db2stop <== 0K
```

- Среда DB2 EEE

- Исправления: Extension Software 12/98 и PHLK_17795.
- Для экземпляра должна быть задана переменная \$DB2INSTANCE.
- Переменная реестра DB2_ENABLE_MEM_WINDOWS должна иметь значение TRUE.
- Для каждого логического узла каждого экземпляра, который нужно выполнять в окне памяти, должна быть запись в файле /etc/services.window. Первое поле такой записи - имя экземпляра, после него идет номер порта. Например:

```
=== $HOME/sqllib/db2nodes.cfg для db2instance1 ===
5 host1 0
7 host1 1
9 host2 0

=== $HOME/sqllib/db2nodes.cfg для db2instance2 ===
1 host1 0
2 host2 0
3 host2 1

=== /etc/services.window на host1 ===
db2instance10 50
db2instance11 55
db2instance20 60

=== /etc/services.window на host2 ===
db2instance10 30
db2instance20 32
db2instance21 34
```

- Не нужно предварять команды DB2 командой db2win (она используется только в среде EE).

Справочник SQL поставляется в виде одного файла PDF

В приложении "Использование библиотеки DB2" в каждой книге указано, что Справочник SQL доступен в формате PDF в виде двух отдельных томов. Это неверно.

Хотя в печатном виде книга состоит из двух томов и два соответствующих номера форм правильны, существует только один файл PDF, содержащий оба тома. Имя этого файла PDF - db2s0x70.

Действия пользователя при возникновении ошибки `dlfm client_conf`

Если на клиенте DLFM по какой-либо причине возникает ошибка `dlfm client_conf`, она может быть вызвана устаревшими записями в каталогах DB2. Чтобы исправить ошибку, выполните следующие команды:

```
db2 uncatalog db <имя_базы_данных>
db2 uncatalog node <алиас узла>
db2 terminate
```

Затем вновь выполните `dlfm client_conf`.

В редких случаях, когда демон копирования не останавливается командой `dlfm stop`

В очень редких случаях `dlfm_copurd` (демон копирования) не останавливается, когда пользователь выполняет команду `dlfm stop` или при аварийном завершении работы. В этом случае перед перезапуском `dlfm` выполните команду `dlfm shutdown`.

Деинсталляция Client Enabler DFS DB2

Перед деинсталляцией Client Enabler DFS DB2 пользователь `root` должен убедиться, что файлы DFS не используются и что в файловом пространстве DFS нет открытых пользователями оболочек. От имени пользователя `root` введите команду:

```
stop.dfs dfs_cl
```

Убедитесь, что `/...` более не смонтирован:

```
mount | grep -i dfs
```

Если это не так, а Client Enabler DFS DB2 деинсталлирован, нужно перезагрузить компьютер.

Аутентификация клиента в Windows NT

Для поддержки существующего механизма аутентификации клиента в среде Windows NT введена новая переменная реестра DB2 DB2DOMAINLIST (список доменов DB2). Эта переменная используется на сервере DB2 for Windows NT для определения одного или нескольких доменов Windows NT. Запросы соединения или подключения будут приниматься только от пользователей, входящих в определенные в этом списке домены.

Эту переменную среды нужно использовать только в чистой среде доменов Windows NT с серверами и клиентами DB2 Версии 7.1 (или старше).

Информацию о задании этой переменной реестра смотрите в разделе "Переменные реестра DB2 и переменные среды" руководства *Administration Guide: Performance*.

AutoLoader может зависать при ветвлении потоков

AIX 4.3.3 содержит исправление ошибки libc, которая может вызывать зависание AutoLoader при ветвлении потоков. AutoLoader - это многопоточная программа. Один из потоков запускает другой процесс. При запуске дочернего процесса в нем создается образ памяти родительского процесса.

Блокировки, используемые libc.a для управления многопоточным выделением памяти из кучи из одного процесса, удерживаются потоком, не запускающим процессы. Поскольку поток, не запускающий процессы, не будет существовать в дочернем процессе, такая блокировка не будет освобождена в дочернем процессе, что может вызвать зависание родительского процесса.

Восстановление столбцов DATALINK

При выполнении восстановления резервной копии, сделанной в отключенном режиме после восстановления базы данных, с повтором транзакций или без повтора, не выполняется обработка быстрого согласования. В таких случаях все таблицы со столбцами DATALINK, зависящими от связей файлов, переводятся в состояние отложенного согласования связей данных (DRP).

Определите ID пользователя и пароль в IBM Communications Server для Windows NT (CS/NT)

Если для соединения удаленных клиентов DB2 с сервером DB2 используется протокол связи APPC и в качестве продукта SNA используется CS/NT, убедитесь, что в файле конфигурации CS/NT правильно заданы следующие ключевые слова. Обычно этот файл находится в каталоге `x:\ibmcs\private`.

Определение узла

TG_SECURITY_BEHAVIOR

Этот параметр позволяет пользователю определить, как узел обрабатывает информацию защиты, содержащуюся в команде ATTACH, если менеджер транзакций не сконфигурирован для защиты

IGNORE_IF_NOT_DEFINED

Этот параметр позволяет пользователю определить, содержит ли команда ATTACH параметры защиты, и игнорировать их, если менеджер транзакций не сконфигурирован для защиты.

Если используется параметр IGNORE_IF_NOT_DEFINED, не нужно определять в CS/NT ID пользователя и пароль.

VERIFY_EVEN_IF_NOT_DEFINED

Этот параметр позволяет пользователю определить, содержит ли команда ATTACH параметры защиты, и проверить их, даже если TP не сконфигурирован для защиты. Это поведение по умолчанию.

Если используется параметр VERIFY_EVEN_IF_NOT_DEFINED, в CS/NT нужно определить ID пользователя и пароль.

Чтобы определить в CS/NT ID пользователя и пароль, выполните следующие действия:

1. Пуск → Программы → IBM Communications Server → Конфигурация узла SNA. Откроется окно Введение в конфигурацию Communications Server.

2. Выберите файл конфигурации, который нужно изменить. Нажмите кнопку **Далее**. Откроется окно Выбор сценария конфигурации.
3. Выберите SPI-C, APPC или эмуляцию 5250. Нажмите кнопку **Завершить**. Откроется окно узел SNA Communications Server.
4. Щелкните по символу [+] рядом с SPI-C и APPC.
5. Щелкните по символу [+] рядом с Защитой LU6.2.
6. Щелкните правой кнопкой мыши по Пароли пользователей и выберите **Создать**. Откроется окно Определение пароля для ID пользователя.
7. Заполните поля ID пользователя и пароль. Нажмите кнопку **ОК**. Нажмите кнопку **Завершить**, чтобы принять изменения.

Ограничения системы объединения

Для систем объединения применяются следующие ограничения:

- Типы данных Oracle NCHAR, NVARCHAR2, NCLOB и BFILE не поддерживаются в запросах, содержащих псевдонимы.
- Из Центра управления не поддерживаются команды Create Server Option, Alter Server Option и Drop Server Option. Для ввода этих команд необходимо использовать процессор командной строки (CLP).
- Для запросов, содержащих псевдонимы, DB2 UDB не всегда следует значению параметра конфигурации базы данных DFT_SQLMATHWARN. Вместо этого DB2 UDB возвращает арифметические ошибки непосредственно от удаленного источника данных независимо от значения параметра DFT_SQLMATHWARN.
- Оператор CREATE SERVER OPTION не позволяет задать значение 'I' для опции сервера COLSEQ для источников данных с регистронезависимыми последовательностями слияния.
- Оператор ALTER NICKNAME возвращает SQL0901N, если задана неверная опция.
- Для источников данных Oracle, Microsoft SQL Server и Sybase числовые типы данных не могут отображаться на тип данных DB2 BIGINT. По умолчанию тип данных Oracle number(p,s), где $10 \leq p \leq 18$ и $s = 0$, отображается на тип данных DB2 DECIMAL.

Ограничение DataJoiner

В распределенных запросах в среде объединения можно использовать только операции чтения.

Менеджер каталогов данных на иврите для Windows NT

Доступна версия компонента менеджера каталогов данных на иврите; она находится на компакт-диске Менеджер хранилищ данных DB2 для NT.

Перевод на иврит находится в файле zip IL_ICM.ZIP, расположенном в каталоге DB2\IL на компакт-диске Менеджер хранилищ данных DB2 для NT.

Чтобы установить версию менеджера каталогов данных на иврите, установите сначала английскую версию менеджера хранилищ данных DB2 для Windows NT и все необходимые для него программы на поддерживающей иврит версии Windows NT.

Установив менеджер хранилищ данных DB2 для Windows NT, распакуйте (unzip) файл IL_ICM.ZIP из каталога DB2\IL в тот же каталог, где установлен менеджер хранилищ данных DB2 для Windows NT. Задайте для программы распаковки unzip правильные опции, чтобы она создала структуру каталогов, сохраненную в файле архива.

После распаковки этого файла нужно изменить значение глобальной переменной среды LC_ALL с En_US на Iw_IL. Чтобы изменить это значение:

1. Откройте Панель управления Windows NT и дважды щелкните по значку **Система**.
2. В окне **Свойства системы** щелкните по закладке **Среда** и найдите переменную **LC_ALL** в разделе **Системные переменные**.
3. Щелкните по этой переменной, чтобы ее значение появилось в поле ввода **Значение**. Измените это значение с **En_US** на **Iw_IL**.
4. Нажмите кнопку **Задать**.
5. Закройте окно **Свойства системы** и **Панель управления**.

Теперь установлена версия менеджера каталогов данных на иврите.

Microsoft SNA Server и поддержка многоузлового изменения SNA (двухфазного принятия)

Прикладные программы хоста и AS/400 не могут обращаться к серверам DB2 UDB, используя двухфазное принятие SNA, если в качестве продукта SNA используется Microsoft SNA Server. Во всех публикациях по DB2 UDB, в которых указывается, что такие обращения поддерживаются, содержат неверную информацию. Требуется IBM Communications Server for Windows NT Версии 5.02 или более поздней.

Примечание: Прикладные программы, обращающиеся к серверам баз данных хоста или AS/400 с помощью DB2 UDB for Windows *могут* использовать двухфазное принятие SNA, если применяется Microsoft SNA Server Версии 4 Service Pack 3 или более поздний.

Ошибка запуска SPM SNA DB2 после загрузки Windows

Если используется Microsoft SNA Server Версии 4 SP3 или более поздней, убедитесь, что после перезагрузки системы был успешно запущен SPM SNA DB2. Проверьте, нет ли в файле `\sqllib\<имя экземпляра>\db2diag.log` записей, подобных следующим:

```
2000-04-20-13.18.19.958000 Экземпляр:DB2 Узел:000
PID:291(db2syscs.exe) TID:316 Appid:нет
common_communication sqlccspsmconnmgr_APPC_init Проверка:19
SPM0453C Менеджер точек синхронизации не запущен, так как не запущен сервер Microsoft SNA.
2000-04-20-13.18.23.033000 Экземпляр:DB2 Узел:000
PID:291(db2syscs.exe) TID:302 Appid:нет
common_communication sqlccsna_start_listen Проверка:14
DIA3001E Неудачная попытка запуска поддержка протокола "SNA SPM".
2000-04-20-13.18.23.603000 Экземпляр:DB2 Узел:000
PID:291(db2syscs.exe) TID:316 Appid:нет
common_communication sqlccspsmconnmgr_listener Проверка:6
DIA3103E Ошибка поддержки протокола APPC. Фраза APPC - "APPC(DISPLAY 1 BYTE)". Первичный rc - "F004"
```

Если в файле `db2diag.log` есть такие записи и их отметка времени соответствует времени последней перезагрузки, необходимо:

1. Выполнить команду `db2stop`.
2. Запустить службу `SnaServer` (если она еще не запущена).
3. Выполнить команду `db2start`.

Вновь проверьте файл `db2diag.log`, чтобы убедиться, что в него больше не добавляются такие записи.

Национальные настройки для сервера администратора DB2

Убедитесь, что национальная версия экземпляра сервера администратора DB2 совместима с национальной версией экземпляра DB2. В противном случае экземпляр DB2 не сможет связываться с сервером администратора DB2.

Если в профиле пользователя сервера администратора DB2 не задана переменная среды LANG, сервер администратора DB2 будет запущен с системной национальной версией по умолчанию. Если системная национальная версия по умолчанию не определена, сервер администратора будет запущен с кодовой страницей 819. Если экземпляр DB2 использует одну из национальных версий DBCS, а сервер администратора DB2 запущен с кодовой страницей 819, этот экземпляр не сможет связываться с сервером администратора DB2. Национальная версия экземпляра сервера администратора DB2 и национальная версия экземпляра DB2 должны быть совместимы. Например, в системе Linux с упрощенным китайским в профиле пользователя сервера администратора DB2 должно быть задано "LANG=zh_CN".

Клавиши быстрого вызова не работают

В некоторых языках для Центра управления в системах на основе UNIX и в OS/2 не работают некоторые клавиши быстрого вызова. Используйте для выбора опций мышь.

Требования к учетным записям для DB2 в Windows NT и Windows 2000

При установке DB2 для Windows NT или Windows 2000 программа установки создает несколько служб Windows и назначает каждой из них свою учетную запись. Для правильной работы DB2 программа установки предоставляет учетной записи, ассоциированной со службой DB2, следующие права пользователя:

- Действовать как часть операционной системы
- Создавать объекты маркеров
- Увеличивать квоты
- Регистрироваться как служба
- Заменять элемент уровня процесса.

Если вы хотите использовать для служб DB2 другую учетную запись, необходимо предоставить ей эти права пользователя.

В дополнение к этим правам пользователя у учетной записи должен быть доступ для чтения к каталогу, в котором установлен программный продукт DB2.

У учетной записи для службы Сервера администратора DB2 (службы DB2DAS00) должны быть также полномочия на запуск и остановку других служб DB2 (то есть эта учетная запись должна принадлежать к группе уполномоченных пользователей), а также полномочия DB2 SYSADM для всех экземпляров DB2, которыми управляет этот сервер.

Потеря привилегии EXECUTE для пользователей Query Patroller, созданных в версии 6

Из-за того, что в Query Patroller Версии 7 добавлено несколько новых хранимых процедур (IWM.DQPGROUP, IWM.DQPVALUR, IWM.DQPCALCT и IWM.DQPJOB), у существующих пользователей, созданных в Query Patroller Версии 6, не сохраняется привилегия EXECUTE в отношении этих пакетов. Программа для автоматического исправления этой ошибки добавлена в FixPak 1.

При попытке использования DQP Query Admin для изменения информации пользователя DQP не пытайтесь удалить существующих пользователей из списка пользователей.

Ограничения Query Patroller

Из-за ограничений платформы для JVM (виртуальной Java-машины) Query Enabler не поддерживается в HP-UX и NUMA-Q. Кроме того, в NUMA-Q не поддерживается Query Patroller Tracker. Если требуются все инструменты клиента Query Patroller, мы рекомендуем воспользоваться другой платформой (например, Windows NT), чтобы запускать эти инструменты для сервера HP-UX или NUMA-Q.

Необходимость принятия всех пользовательских программ, которые будут использованы в Центре хранилищ данных (Data Warehouse Center - DWC)

Если вы хотите использовать хранимую процедуру, построенную Построителем хранимых процедур DB2, в качестве пользовательской программы в Центре хранилищ данных (DWC), в эту хранимую процедуру перед оператором `con.close()`; необходимо вставить оператор:

```
con.commit();
```

Если этот оператор не вставить, для изменений, внесенных хранимой процедурой при запуске ее из DWC, будет выполнен откат.

Чтобы изменения вступили в силу в базе данных, для всех пользовательских программ в DWC необходимо *явным образом* выполнять принятие для всех включенных в них функций DB2, то есть в пользовательские программы необходимо добавлять оператор COMMIT.

Новая опция команды экспорта Центра хранилищ данных

У команды экспорта из командной строки для файлов языка тегов появилась новая опция /B. Эта опция недоступна через интерфейс Центра хранилищ данных.

Новый синтаксис для команды **iwhtexp2**:

```
iwhtexp2 имя_файла.INP имя_базы_данных id_пользователя пароль [PREFIX=схема_таблицы] [/S] [/R] [/I]
```

где

- имя_файла.INP - полное имя файла INP
- имя_базы_данных - имя управляющей базы данных Центра хранилищ данных
- id_пользователя - ID пользователя, использованный для регистрации в этой базе данных
- пароль - пароль, использованный для регистрации в этой базе данных
- необязательные параметры:
 - PREFIX=схема_таблицы - схема таблицы для таблиц управляющей базы данных (значение по умолчанию - IWH)

- /S: - экспортировать расписания с выбранными шагами
- /R: - не экспортировать источники хранилища с выбранными шагами
- /B: - не экспортировать участвующие шаги с выбранными шагами

Примечание: Если указано /R или /B, источники хранилища или участвующие шаги должны уже существовать при импорте полученного файла языка тегов, иначе будет возвращена ошибка.

API служб резервного копирования (XBSA)

API служб резервного копирования (XBSA) был определен The Open Group в Великобритании как открытый интерфейс прикладного программирования между прикладными программами или утилитами, которым для целей резервного копирования или архивирования требуется управление хранением данных. Стандарт определен в книге "Open Group Technical Standard System Management: Backup Services API (XBSA)", номер документа C425 (ISBN: 1-85912-056-3).

Для поддержки этих API созданы две новые переменные реестра DB2, в настоящее время поддерживаемые в AIX, HP, Solaris и Windows NT:

DB2_VENDOR_INI

Задаёт файл, содержащий все установки среды, специфичные для конкретного производителя. Значение читается при запуске менеджера баз данных.

DB2_XBSA_LIBRARY

Указывает на предоставляемую производителем библиотеку XBSA. В AIX эта установка должна включать в себя совместно используемый объект, если он не носит имя shr.o. В HP, Solaris и Windows NT имени совместно используемого объекта не требуется. Например, чтобы использовать NetWorker Business Suite Module для DB2 фирмы Legato, надо задать значение для этой переменной реестра так:

```
db2set DB2_XBSA_LIBRARY="/usr/lib/libxdb2.a(bsashr10.o)"
```

Интерфейс XBSA можно вызвать из команды BACKUP DATABASE или RESTORE DATABASE. Например:

```
db2 backup db sample use XBSA
db2 restore db sample use XBSA
```

Агент OS/390

Содержание документа. Установка в OS/390 и возможности

В этом документе вы найдете указания о том, как устанавливать агент OS/390, и информацию о его возможностях. Краткий обзор процесса установки смотрите в разделе "Обзор установки", а подробные указания - в разделе "Подробности установки". Информацию о возможностях агента смотрите в разделах "Установка дополнительных возможностей агента", "Преобразователи" и "Доступ к другим базам данных (не DB2)".

Обзор

Центр хранилищ данных DB2 содержит агент OS/390. Этот агент можно использовать для связи между DB2 Universal Database for OS/390 и другими базами данных, включая базы данных DB2 на других платформах и базы данных, отличные от DB2. Агент может обмениваться данными с поддерживаемыми источниками

данных, использующими соединение ODBC. Агент работает под OS/390 UNIX Systems Services. Для него требуется OS/390 V2R6 или новее, и он совместим с DB2 for OS/390 Версий 5, 6 и 7.

Агент OS/390 поддерживает следующие задачи:

- Копирование данных из исходной базы данных DB2 в базу данных назначения DB2
- Выборка примера содержимого из таблицы или файла
- Выполнение пользовательских программ
- Доступ к другим базам данных (не DB2) через IBM DataJoiner
- Доступ к данным VSAM или IMS через Classic Connect фирмы Cross Access
- Запуск утилит DB2 Universal Database for OS/390
- Запуск задания Apply для IBM Data Propagator

Обзор установки

Эти шаги представляют собой общее описание процесса установки. Подробности шагов приводятся в разделе "Подробности установки".

1. Установка агента OS/390 с ленты DB2 Universal Database for OS/390
2. Изменение переменных среды в файле профиля.
3. Конфигурирование соединений:
 - Между ядром и демоном агентов.
 - Между агентом и базами данных, к которым он будет обращаться.
4. Связывание CLI локально и с любыми удаленными базами данных.
5. Задание файла инициализации ODBC.
6. Задание авторизации так, чтобы пользователь:
 - Мог запускать демон агентов.
 - Имел полномочия выполнения для плана DSNAOCLI.
 - При необходимости имел полномочия на чтение и запись в каталогах журнала и трассировки ODBC
7. Запуск демона агентов.

Подробности установки

Установка агента OS/390

Агент OS/390 включен в состав ленты с DB2 Universal Database for OS/390 Версии 7. Подробности установки агента OS/390 смотрите в программном каталоге, прилагаемом к этой ленте.

Перед установкой агента OS/390 в подсистеме DB2 необходимо установить APAR PQ36585 или PQ36586.

Изменение переменных среды в файле профиля

Эти переменные задают для агента различные библиотеки DB2, выходные каталоги и т.п.

В следующем примере приводится пример содержимого файла `.profile`. Файл `.profile` определяет переменные среды и должен находиться в домашнем каталоге пользователя, запускающего демон агентов:

```

export VWS_LOGGING=/usr/lpp/DWC/logs
export VWP_LOG=/usr/lpp/DWC/vwp.log
export VWS_TEMPLATES=/usr/lpp/DWC/
export DSNAOINI=/usr/lpp/DWC/dsnaoini
export LIBPATH=/usr/lpp/DWC/:$LIBPATH
export PATH=/usr/lpp/DWC/:$PATH
export STEPLIB=DSN710.SDSNEXIT:DSN710.SDSNLOAD

```

Конфигурирование соединений

Чтобы сконфигурировать соединения ядра с демоном агентов, добавьте в файл /etc/services или TCPIP.ETC.SERVICES следующие строки:

```

wkernel 11000/tcp
wvd 11001/tcp
wvlogger 11002/tcp

```

Чтобы сконфигурировать соединения между агентом OS/390 и базами данных, добавьте все удаленные базы данных в базу данных связи OS/390 (communications database - CDB). Несколько примеров добавления в CDB:

```

INSERT INTO SYSIBM.LOCATIONS
      (LOCATION, LINKNAME, PORT)
VALUES
      ('NTDB', 'VWNT704', '60002');

INSERT INTO SYSIBM.IPNAMES
      (LINKNAME, SECURITY_OUT, USERNAMES, IPADDR)
VALUES
      ('VWNT704', 'P', '0', 'VWNT704.STL.IBM.COM');

INSERT INTO SYSIBM.USERNAMES
      (TYPE, AUTHID, LINKNAME, NEWAUTHID, PASSWORD)
VALUES
      ('0', 'MVSUID', 'VWNT704', 'NTUID', 'NTPW');

```

Дополнительную информацию о конфигурировании соединений и изменении базы данных связи смотрите в главе "Connecting Distributed Database Systems" руководства *DB2 UDB for OS/390 Installation Guide*, GC26-9008-00.

Связывание CLI

Поскольку агент OS/390 использует CLI для связи с DB2, необходимо связать план CLI со всеми удаленными базами данных, к которым агент будет обращаться. Несколько примеров операторов связывания пакетов для локальной базы данных DB2 for OS/390:

```

BIND PACKAGE (DWC6CLI) MEMBER(DSNCLICS) ISO(CS)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLINC) ISO(NC)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIRR) ISO(RR)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIRS) ISO(RS)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIUR) ISO(UR)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIMS)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIC1)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIC2)
BIND PACKAGE (DWC6CLI) MEMBER(DSNCLIF4)

```

Несколько примеров операторов связывания пакетов для базы данных DB2, работающей под Windows NT:

```

BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLICS) ISO(CS)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLINC) ISO(NC)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIRR) ISO(RR)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIRS) ISO(RS)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIUR) ISO(UR)

```

```
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIC1)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIC2)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIQR)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIF4)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIV1)
BIND PACKAGE (NTDB.DWC6CLI) MEMBER(DSNCLIV2)
```

Пример оператора для связывания пакетов CLI в единый план:

```
BIND PLAN(DWC6CLI) PKLIST(*.DWC6CLI.* )
```

Задание файла инициализации ODBC

Пример файла инициализации ODBC, `inisamp`, включен в каталог `usr/lpp/DWC/`. Можно отредактировать этот файл для работы с вашей системой или создать свой собственный файл. Чтобы убедиться, что этот файл работает правильно, проверьте, правильно ли он сконфигурирован:

- Переменная среды `DSNAOINI` должна указывать на этот файл инициализации.
- Имя этого файла должно следовать схеме `dsnaoini.имя_положения`.
- Файл должен включать в себя параметры `CONNECTTYPE=2` и `MVSATTACHTYPE=CAF`.

Дополнительную информацию о связывании CLI и файле `DSNAOINI` смотрите в справочном руководстве *DB2 UDB for OS/390 ODBC Guide and Reference*, SC26-9005.

Задание авторизации

Агент OS/390 является процессом демона. Демон агентов можно запускать с обычной защитой UNIX или с защитой UNIX OS/390.

Поскольку агенту требуется авторизация демона, определите следующие исполняемые программы агента для RACF Program Control:

- `libt1s4d.dll`
- `iwhcomnt.dll`
- `vwd`

Чтобы определить исполнимые программы для управления программами RACF, перейдите в каталог, в котором хранятся исполняемые файлы Центра хранилищ данных, и запустите следующие команды:

```
extattr +p libt1s4d.dll
extattr +p iwhcomnt.dll
extattr +p vwd
```

Чтобы использовать команду `extattr` с параметром `+p`, у вас должен быть доступ по крайней мере для чтения к классу `BPX.FILEATTR.PROGCTL FACILITY`. В следующем примере показана команда RACF, используемая для предоставления такого разрешения ID пользователя `SMORG`:

```
RDEFINE FACILITY BPX.FILEATTR.PROGCTL UACC(NONE)
PERMIT BPX.FILEATTR.PROGCTL CLASS(FACILITY) ID(SMORG) ACCESS(READ)
SETROPTS RACLIST(FACILITY) REFRESH
```

Дополнительную информацию об авторизации смотрите в книге *OS/390 UNIX System Services Planning*, SC28-1890.

Запуск демона агентов

После завершения конфигурирования системы запустите демон агентов:

1. Свяжитесь по telnet с UNIX Systems Services на OS/390, используя имя хоста OS/390 и порт USS.
2. Запустите демон агентов:
 - Чтобы запустить демон в приоритетном режиме, введите в командную строку vwd.
 - Чтобы запустить демон в фоновом режиме, введите:
vwd>/usr/lpp/DWC/logs/vwd.log 2>&1 &

Чтобы проверить, что демон агентов OS/390 запущен, введите в командной строке оболочки UNIX:

```
ps -e | grep vwd
```

Другой вариант - ввести с консоли OS/390 D 0MVS,a=all и найти строку vwd

Настройка дополнительных функций агента

В программный пакет Менеджер хранилищ DB2 включены следующие пользовательские программы:

- vwrftp: запускает командный файл FTP
- vwrmvsv: запускает поток заданий JCL
- vwrgrsu: копирует файл по протоколу FTP
- XTClient: программа триггера клиента
- etidlmvs: утилита от ETI (Evolutionary Technologies International); удаляет файл в MVS.
- etiremvsv: утилита от ETI; запускает FTP на хосте MVS.
- etiehmvs: утилита от ETI; запускает JCL на хосте MVS.

Кроме того, можно определять в Центре хранилищ данных пользовательские программы и хранимые процедуры. Агент OS/390 поддерживает любые исполнимые программы, работающие под UNIX Systems Services.

Пользовательская программа назначается одному или нескольким шагам. При запуске пользовательской программы происходят следующие действия:

- Запускается агент.
- Агент запускает пользовательскую программу.
- Пользовательская программа возвращает агенту код возврата и файл обратной связи.
- Агент возвращает результаты ядру.

Чтобы запускать программы ETI на OS/390, сначала необходимо установить FixPack 2 для DB2 Universal Database Версии 7.1.

Переменная среды VWP_LOG задает каталог, куда могут записывать вывод пользовательские программы.

Если вы используете пользовательскую программу для выполнения задания, использующего FTP, сначала необходимо создать JCL и данные, которые вы хотите передать. Имя задания JCL должно иметь вид USERIDx, где x - одна буква или цифра (пример: MYUSERA). Класс вывода для файлов MSGCLASS и SYSOUT, заданных в JCL, должен указывать класс вывода, принадлежащий JES.

Ограничение: максимальная LRECL для выполняемого задания - 254 символов. JES просматривает только первые 72 символа JCL.

Изменение шаблона Центра хранилищ данных для поддержки FTP

Центр хранилищ данных устанавливает шаблон JCL для передачи файлов с использованием FTP. Если вы планируете, что агент OS/390 будет использовать команды FTP GET или PUT для передачи файлов с хоста OS/390 на другой удаленный хост, необходимо изменить учетную информацию в шаблоне JCL для вашей системы OS/390:

1. Зарегистрируйтесь с ID, у которого есть полномочия копирования и изменения файлов в каталоге /usr/lpp/DWC.
2. Найдите файл ftp.jcl и сделайте его копию под новым именем имя_системы.ftp.jcl, где *имя_системы* - имя системы OS/390.
3. Создайте копию этого файла для каждой системы OS/390, на которой вы собираетесь запускать программы преобразования vwrpmvs или ETI extract. Например, если хотите запустить любую из этих программ на STLMVS1, создайте копию этого файла с именем STLMVS1.ftp.jcl.
4. При помощи текстового редактора исправьте JCL в соответствии с местными требованиями. Измените учетную информацию, чтобы она соответствовала стандартам учетной информации вашей системы MVS. Не изменяйте параметры в скобках, такие как [USERID] и [FTPFILE]. (Скобки представлены шестнадцатеричными символами x'AD' и x'BD', соответственно. Если для вашего типа терминала TSO в опции SPF 0 не установлено 3278A, вы можете увидеть вместо скобок специальные символы. В этом нет ничего страшного, если не изменять x'AD' и x'BD', а также какие-либо данные между этими символами.)
5. Измените переменную среды VWS_TEMPLATES, чтобы она указывала на каталог, в который скопирован файл шаблона.

В Центр хранилищ данных включен следующий пример шаблона JCL:

```
//[USERID]A JOB , 'PUT/GET ',  
// CLASS=A,  
// USER=&SYSUID,  
// NOTIFY=&SYSUID,  
// TIME=(,30),  
// MSGCLASS=H  
//STEP1 EXEC PGM=FTP,PARM='( EXIT'  
//INPUT DD DSN=[FTPFILE],DISP=SHR  
//OUTPUT DD SYSOUT=*  
//SYSPRINT DD SYSOUT=*
```

Выборка содержимого таблицы или файла

Использование агента OS/390 позволяет производить выборку содержимого таблиц DB2 и плоских файлов, таких как файлы UNIX Systems Services и собственные плоские файлы OS/390. Classic Connect с использованием агента OS/390 позволяет также производить выборку содержимого файлов IMS или VSAM. В случае плоского файла для определения его формата агент использует параметры в свойствах определения файла.

Составление расписания шагов хранилища при помощи программы триггера (XTClient)

Программа триггера используется для создания расписания шагов хранилища данных с платформы OS/390. Вы или планировщик заданий OS/390 может

инициализировать задание, которое переключает шаг в Центре хранилищ данных. Если этот шаг выполнен успешно, шаг триггера в JCL возвращает код возврата 0.

Чтобы использовать программу триггера, в службах системы Unix OS/390 должен быть установлен Java Development Kit (JDK) 1.1.8 или более новый. Чтобы запустить этот триггер, сначала запустите XTServer на компьютере, на котором работает сервер хранилища данных. Этот процесс описан в главе 5 руководства *Data Warehouse Center Administration Guide* в теме "Starting a step from outside the Data Warehouse Center". После запуска XTServer запустите XTClient в OS/390. Далее приводится пример JCL для запуска триггера.

```
//DBA1A JOB 1,'XTCLIENT',CLASS=A,MSGCLASS=H,
//          MSGLEVEL=(1,1),REGION=4M,NOTIFY=&SYSUID
//*****
//* submit iwhetrig
//*****
//BRADS EXEC PGM=BPXBATCH,
//          PARM=('sh cd /usr/lpp/DWC/; java XTClient 9.317.171.133 1100x
//          9 drummond pw bvmvs2nt 1 1 100')
//STDOOUT DD PATH='/tmp/xtclient.stdout',
//          PATHOPTS=(OWRONLY,OCREAT),
//          PATHMODE=SIRWXU
//STDERR DD PATH='/tmp/xtclient.stderr',
//          PATHOPTS=(OWRONLY,OCREAT),
//          PATHMODE=SIRWXU
//
```

Примечание: В приведенном примере кода JCL показано, как продолжать параметры на новой строке. Для этого вводите параметры до колонки 71, а в колонку 72 поместите 'X' и продолжайте с колонки 16 следующей строки.

Первая часть этого параметра (cd /usr/lpp/DWC/;) представляет собой оператор перехода в каталог, в котором установлен агент OS/390. Вторая часть параметра запускает XTClient и передает 8 следующих параметров:

- Имя хоста или IP-адрес сервера Центра хранилищ данных
- Порт сервера Центра хранилищ данных (обычно 11009)
- ID пользователя Центра хранилищ данных
- Пароль Центра хранилищ данных
- Имя выполняемого шага
- Команда сервера Центра хранилищ данных, где:
 - 1 = заполнение шага
 - 2 = перевод шага в режим тестирования
 - 3 = перевод шага в режим производства
 - 4 = перевод шага обратно в режим тестирования
 - 5 = перевод шага обратно в режим разработки
- Опция необходимости ожидания завершения шага; 1 = да, 0 = нет
- Максимальное число строк (чтобы прочесть все строки, укажите 0 или пробел)

Преобразователи

Введение

12 перечисленных преобразователей представляют собой хранимые процедуры Java, обеспечивающие некоторые из основных преобразований данных. Чтобы запустить эти преобразователи, сначала необходимо установить хранимые процедуры Java в

подсистеме DB2. Дополнительная информация об этих преобразователях приводится в руководстве *IBM DB2 Universal Database Data Warehouse Center Administration Guide Version 7*, SC26-9993-00.

IWH.CLEAN
IWH.PERIODTABLE
IWH.KEYTABLE
IWH.CHISQUARE
IWH.CORRELATION
IWH.STATISTICS
IWH.INVERTDATA
IWH.PIVOTDATA
IWH.REGRESSION
IWH.ANOVA
IWH.SUBTOTAL
IWH.MOVINGAVERAGE

Установка хранимых процедур Java

Это краткая версия полного набора указаний об установке хранимых процедур Java, который приводится в справочнике *Application Programming Guide and Reference for Java™*, SC26-9018

1. Установите в подсистеме DB2 PTF UQ46170 и UQ46114.
2. Установите в системе OS/390 Visual Age for Java 2.0 или более новую его версию.
3. Установите JDBC в DB2 и свяжите пакеты JDBC в подсистеме DB2.
4. Установите хранимые процедуры RRS и DB2 WLM для подсистемы DB2.
5. Установите хранимые процедуры Java для DB2. При этом надо создать процедуру запуска WLM Java для адресного пространства хранимых процедур Java.
6. В WLM необходимо связать процедуру запуска WLM Java с именем среды WLM. Чтобы связать имя среды с процедурой JCL, используйте панель среды прикладных программ WLM с названием "Create an Application Environment".
7. Укажите имя среды прикладных программ WLM в опции WLM_ENVIRONMENT оператора CREATE или ALTER PROCEDURE, чтобы связать хранимую процедуру или пользовательскую функцию со средой прикладных программ.
8. Убедитесь в том, что у владельца запущенных задач DB2 есть доступ к библиотекам в процедуре запуска WLM Java.

Шаги установки преобразователей хранилища

Это краткая версия полного набора указаний, который приводится в руководстве *IBM DB2 Universal Database Data Warehouse Center Administration Guide Version 7*, SC26-9993-00

1. Либо установите Fixpack 3 в DB2 Universal Database for NT версии 7, либо измените параметры управляющей базы данных хранилища на TRANSREGISTERED=1 и TRANSFENCED=1.

Чтобы изменить параметры управляющей базы данных хранилища, введите в командное строки DB2 Universal Database следующие операторы SQL:

```
CONNECT TO ваша управляющая база данных vw  
UPDATE IWH.INFORESOURCE SET TRANSREGISTERED = '1' WHERE SUBDBTYPE = 'DB2 MVS'  
UPDATE IWH.INFORESOURCE SET TRANSFENCED = '1' WHERE SUBDBTYPE = 'DB2 MVS'
```

2. Определите преобразователи для DB2
 - Если у вас DB2 for OS/390 Версии 7, используйте операторы SQL из /usr/lpp/DWC/createXfSQLV7.
 - Если у вас DB2 for OS/390 Версии 6, используйте операторы SQL из /usr/lpp/DWC/createXfSQL.

- Если у вас DB2 for OS/390 Версии 5, используйте приведенные в виде комментариев операторы SQL в /usr/lpp/DWC/createXfSQL. Удалите значки комментариев у всех операторов CREATE PROCEDURE. Затем удалите значки комментариев у операторов INSERT INTO SYSIBM.SYSPROCEDURES и используйте их для определения преобразователей для DB2 for OS/390 Версии 5.

При установке хранимых процедур Java WLM используется для связывания процедуры запуска WLM Java с именем среды WLM. Это имя среды указано в опции WLM ENVIRONMENT оператора CREATE PROCEDURE. DSNWLMJ - имя среды WLM, включенное в определения преобразователей, описанных выше. Можно либо добавить имя связи WLM для DSNWLMJ, либо для каждого из определений преобразователей заменить опцию WLM ENVIRONMENT на имя, которое вы уже связали с процедурой запуска.

3. Установите связи от UNIX Systems Services к модулям загрузки преобразователей в IWH710.SIWHLOAD.

- Свяжитесь по Telnet с UNIX Systems Services в системе хоста OS/390.
- Перейдите в каталог, в котором установлен агент OS/390. Каталог установки по умолчанию - /usr/lpp/DWC.
- Если вы используете DB2 V7, перейдите к шагу 4. Если вы используете DB2 V5 или V6, отредактируйте набор данных trlinks в каталоге установки. Превратите следующую строку в комментарий, поставив знак решетки (#) в столбце 1:

```
ln -e IWHXFV7 xf.jll;
```

Удалите в следующей строке из столбца 1 значок комментария (решетку - #).

```
#ln -e IWHXF xf.jll;
```

Сохраните внесенные изменения.

- Введите trlinks и нажмите клавишу Enter. В этом каталоге будет создана связь xf.jll, которая будет указывать агенту, загружать модуль IWHXF или IWHXFV7.
4. Посредством APF авторизуйте IWH710.SIWHPDSE, а затем добавьте его в перечисление STEPLIB в процедуре запуска хранимых процедур Java DB2.
5. Добавьте каталог, в котором находится связь xf.jll (по умолчанию: /usr/lpp/DWC), в переменные среды CLASSPATH и LIBPATH в наборе данных среды WLM.
- Если вы не знаете, где расположен набор данных среды WLM, посмотрите в процедуре запуска хранимых процедур Java DB2. Ваш набором данных среды WLM - тот, на который указывает оператор определения данных (DD) JAVAENV.
6. Запустите хранимые процедуры, а затем создайте и запустите шаги хранилища.

Ограничения для хранимых процедур Java

Объекты Java в сигнатуре хранимой процедуры поддерживаются только в DB2 for OS/390 Версии 7. По этой причине преобразователи в DB2 for OS/390 версий 5 или 6 не поддерживают в своих параметрах пустые значения. В этих версиях при передаче пустого параметра он приравнивается к нулевому значению. Преобразователи версий 5 и 6 рассматривают нулевые параметры, как пустые строки.

DB2 поддерживает в хранимых процедурах оператор SQL COMMIT только в DB2 for OS/390 Версии 7. Хранимая процедура INVERTDATA отбрасывает и повторно создает таблицу в хранимой процедуре, поэтому ей требуется оператор принятия. Поэтому IWH.INVERTDATA не поддерживается в DB2 for OS/390 Версии 5 или Версии 6.

DB2 for OS/390 не поддерживает пользовательские функции Java, поэтому IWH.FORMATDATE не поддерживается на платформе 390.

Пример процедуры запуска для хранимых процедур Java

(описан в книге "DB2 for OS/390 Application Programming Guide and Reference for Java"):

```
//DSNWLMJ PROC DB2SSN=DSN,NUMTCB=5,APPLENV=DSNWLMJ <-- значение WLM ENVIRONMENT в CREATE PROC
//*****
//* ЭТА ПРОЦЕДУРА ИСПОЛЬЗУЕТСЯ ДЛЯ ЗАПУСКА УСТАНОВЛЕННОГО WLM *
//* АДРЕСНОГО ПРОСТРАНСТВА SPAS ДЛЯ СРЕДЫ ПРИКЛАДНЫХ ПРОГРАММ DSNWLMJ *
//* V WLM,APPLENV=DSNWLMJ,RESUME *
//*****
//DSNWLMJ EXEC PGM=DSNX9WLM,TIME=1440,REGION=0M,
// PARM='&DB2SSN, &NUMTCB, &APPLENV'
//STEPLIB DD DSN=DSN.TESTLIB,DISP=SHR
// DD DSN=IWH710.SIWHPDSE,DISP=SHR <-- Содержит в себе преобразователи
// DD DSN=DSN.HPJSP.PDSE.JDBC,DISP=SHR <-- DLL HPJ из установки HPJ
// DD DSN=SYS1.PP.PDSELINK,DISP=SHR <-- библиотек времени выполнения HPJ
// DD DSN=DSN710.SDSNEXIT,DISP=SHR
// DD DSN=DSN710.SDSNLOAD,DISP=SHR
// DD DSN=SYS1.SCEERUN,DISP=SHR
// DD DSN=DSN.PDSE,DISP=SHR <-- Информация установки HPJ
//JAVAENV DD DSN=DSN.WLMENVJ.JSPENV,DISP=SHR <-- Переменные среды, смотрите далее
//CEEDUMP DD SYSOUT=A
//DSSPRINT DD SYSOUT=A
//JSPDEBUG DD SYSOUT=A
//SYSABEND DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
```

Пример набора данных переменной среды

(описан в книге "DB2 for OS/390 Application Programming Guide and Reference for Java"):

```
ENVAR("TZ=PST07",
"DB2SQLJPROPERTIES=/usr/lpp/db2/jdbc/db2710/classes/db2sqljjdbc.properties",
"LIBPATH=/usr/lpp/DWC",
"VWSPATH=/usr/lpp/DWC",
"CLASSPATH=/usr/lpp/db2/jdbc/db2710/classes:/usr/lpp/DWC:/usr/lpp/hpj/lib"),
MSGFILE(JSPDEBUG)
```

Поддержка национальных языков для преобразователей

Большая часть сообщений, сгенерированных агентом OS/390, посылается для интерпретации на платформу NT, и поэтому в большинстве случаев язык сообщения зависит от того, как установлена UDB для NT. Преобразователи являются исключением. Агент OS/390 предоставляет для преобразователей следующие файлы сообщений:

Имя файла:	Язык:
Xf.properties_Fi_FI	Финский в Финляндии
Xf.properties_No_NO	Норвежский в Норвегии
Xf.properties_Ru_RU	Русский в России
Xf.properties_Zh_CN	Китайский в Китае (КНР)
Xf.properties_Zh_TW	Китайский на Тайване
Xf.properties_Da_DK	Датский в Дании
Xf.properties_De_DE	Немецкий в Германии
Xf.properties_En_US	Английский в США
Xf.properties_Es_ES	Испанский в Испании
Xf.properties_Fr_FR	Французский во Франции
Xf.properties_It_IT	Итальянский в Италии
Xf.properties_Ja_JP	Японский в Японии

Xf.properties_Ko_KR Корейский в Корее
Xf.properties_Pt_BR Португальский в Бразилии
Xf.properties_Sv_SE Шведский в Швеции

Если сообщения преобразователей должны быть на языке, отличном от английского, выберите один из этих файлов и скопируйте его содержимое в Xf.properties.

Доступ к другим базам данных (не из семейства DB2)

Для доступа к другим базам данных (не DB2 Universal Database) агент OS/390 использует DataJoiner. DataJoiner позволяет агенту использовать нормальный входной поток DRDA, как если бы он был базой данных UDB. Если требование ODBC направлено источнику не из семейства DB2, DataJoiner вызывает дополнительный слой программного кода для доступа к "чужим" базам данных.

DataJoiner может связываться с Oracle, Sybase, Informix, Microsoft SQL Server, Teradata и любыми другими базами данных, у которых есть драйвер ODBC для операционных сред Windows NT, AIX или Solaris фирмы Sun. Агент OS/390 может обращаться к DataJoiner как к источнику, но не как к потребителю. DataJoiner не поддерживает двухфазное принятие. Несмотря на то, что DataJoiner поддерживает TCP/IP как реквестер прикладных программ в версиях 2.1 и 2.1.1, у него нет сервера прикладных программ. Поскольку агенту OS/390 для использования TCP/IP может потребоваться сервер прикладных программ, вместо доступа к DataJoiner из OS/390 используйте соединение SNA.

Доступ к IMS и VSAM в OS/390

Classic Connect приобретается и устанавливается отдельно от агента хранилища. Агент OS/390 может обращаться к IMS и VSAM при помощи драйвера ODBC Classic Connect. Classic Connect позволяет устанавливать DB2-подобное определение наборов данных IMS и VSAM, а затем обращаться к ним с использованием ODBC.

Агент OS/390 загружает правильный драйвер ODBC на основе того, направлено требование к Classic Connect или к DB2. Если вы обращаетесь к источнику DB2, агент загружает драйвер ODBC DB2. Если вы обращаетесь к источнику VSAM или IMS, агент загружает драйвер ODBC Classic Connect. Затем обрабатывается требование агента.

Установка драйвера ODBC Classic Connect и доступ к хранилищам

Classic Connect приобретается и устанавливается отдельно от агента OS/390. Classic Connect может представлять файл или части файла в виде одной или нескольких реляционных таблиц. Для доступа Classic Connect к данным IMS и VSAM эти данные необходимо преобразовать. Можно отображать данные вручную или использовать преобразователь нереляционных данных Microsoft Windows Classic Connect.

1. Установите сервер данных Classic Connect в OS/390
2. Необязательно: установите программный продукт Classic Connect Data Mapper в NT.
3. Задайте определения логической таблицы Classic Connect так, чтобы Classic Connect могла обращаться к данным реляционным способом. Для создания определений для структур IMS и VSAM используйте преобразователь данных или создайте эти определения вручную
4. После установки Classic Connect можно сконфигурировать доступ к хранилищу данных:
 - a. Создайте файл .ini Classic Connect.

Пример файла конфигурации прикладной программы Classic Connect `sxa.ini` находится в каталоге `/usr/lpp/DWC/`; воспроизводим его здесь:

```
* язык сообщений
NL = US English
* основной файл ресурсов
NL CAT = usr/lpp/DWC/v4r1m00/msg/engcat
FETCH BUFFER SIZE = 32000
DEFLOC = CXASAMP
USERID = uid
USERPASSWORD = pwd
DATASOURCE = DJX4DWC tcp/9.112.46.200/1035
MESSAGE POOL SIZE = 1000000
```

- b. Измените строку `DATASOURCE` в файле `.ini`. Эта строка содержит имя источника данных и адрес протокола. Имя источника данных должно соответствовать имени Query Processor, определенному на сервере данных Classic Connect, которое задано в записи `QUERY PROCESSOR SERVICE INFO ENTRY` в файле конфигурации этого сервера данных. Адрес протокола можно найти в этом же файле в записи `TCP/IP SERVICE INFO`. `USERID` и `USERPASSWORD` в этом файле используются при определении источника данных хранилища.
- c. Экспортируйте переменную среды `CXA_CONFIG` в файлы программ Classic Connect; обычно они расположены в том же каталоге, что и файл `.ini`.
- d. Исправьте переменную среды `LIBPATH`, включив в нее путь к файлам программ Classic Connect; обычно они расположены в том же каталоге, что и файл `.ini`.
- e. Необязательно: проверьте установку при помощи программы тестирования `sxasamp`, введя `sxasamp` из каталога, в котором находится файл `.ini`. `location/uid/pwd` - имя источника данных/`id` пользователя/пароль пользователя, определенные в файле `.ini`.
- f. Определите источник данных для хранилища точно так же, как вы определяете любой другой источник данных `DB2`.

Файл `dsnaoini` изменять не надо, поскольку в `DB2 for OS/390` нет менеджера драйверов. Менеджер драйверов для Classic Connect встроен в агент `OS/390`.

Запуск утилит `DB2 for OS/390`

Чтобы использовать агент `OS/390` для запуска утилит, необходимо установить для него `APAR PQ44904`.

`DSNUTILS` - хранимая процедура `DB2 for OS/390`, выполняемая в средах `WLM` и `RRS`. Ее можно использовать для запуска любых установленных утилит `DB2` при помощи интерфейса пользовательских хранимых процедур. У утилит `DB2 for OS/390` `LOAD`, `REORG` и `RUNSTATS` есть страницы свойств, которые можно использовать для изменения работы этих утилит. Чтобы изменить свойства какой-нибудь из этих утилит, щелкните по ней правой кнопкой мыши в окне Разработчик моделей процессов и выберите пункт **Свойства**.

Менеджер хранилищ предоставляет также интерфейс для `DSNUTILS`, позволяющий включать утилиты `DB2` в шаги Менеджера хранилищ.

Для установки хранимой процедуры `DSNUTILS`:

1. При установке `DB2` запустите задание `DSNTIJSJG`, чтобы установить и связать хранимую процедуру `DSNUTILS`. Убедитесь, что определение `DSNUTILS` использует `PARAMETER STYLE GENERAL`.
2. Разрешите управляемые `WLM` хранимые процедуры.

3. Задайте среды RRS и WLM.
4. Запустите примеры пакетных программ DSNUTILS, поставляемые с DB2.
(Рекомендуется, но не обязательно.)
5. Свяжите план DSNUTILS с планом DSNCLI, чтобы CLI мог вызывать хранимую процедуру:
BIND PLAN(DSNAOCLI) PKLIST(*.DSNAOCLI.*, *.DSNUTILS.*)
6. Сконфигурируйте шаг с использованием Менеджера хранилищ и выполните этот шаг. Надо использовать тип заполнения APPEND. *Иначе Менеджер хранилищ перед выполнением утилиты удалит все содержимое таблицы.*

Копирование данных между таблицами DB2 for OS/390 с использованием утилиты LOAD

Предположим, что вы хотите скопировать таблицу, выгрузив ее в плоский файл, а затем загружая этот плоский файл в другую таблицу. Обычно, чтобы это сделать, необходимо выгрузить данные, отредактировать операторы управления загрузкой, которые образуются при выгрузке, а затем загрузить данные. При использовании хранилища можно указать, что вы хотите перегрузить данные в другую таблицу без остановки между шагами и редактирования управляющих операторов вручную. Делается это так: воспользуйтесь интерфейсом Reorg/Generic для создания шага, который выгружает файл с использованием утилиты UNLOAD или утилиты REORG TABLESPACE. Обе эти утилиты создают два выходных набора данных, один - с данными таблицы, а другой - с управляющим оператором утилиты, который может использовать утилита LOAD. В управляющем операторе, сгенерированном утилитой, имя таблицы INTO TABLE - это имя выгруженной таблицы. Пример параметров DSNUTILS, которые можно использовать для шага Reorg Unload:

Таблица 18. Свойства для шага Reorg Unload

UTILITY_ID	REORGULX
RESTART	NO
UTSTMT	REORG TABLESPACE DBVW.USAINENT UNLOAD EXTERNAL
UTILITY_NAME	REORG TABLESPACE
RECDSN	DBVW.DSNURELD.RECDSN
RECDEVT	SYSDA
RECSpace	50
PNCHDSN	DBVW.DSNURELD.PNCHDSN
PNCHDEVT	SYSDA
PNCHSPACE	3

Для создания шага загрузки используется интерфейс Reorg/Generic DSNUTILS. Обычно параметр оператора утилиты DSNUTILS задает оператор управления утилитой. Интерфейс утилит хранилища допускает также задание имени файла в поле оператора утилиты. Указать файл, содержащий действительный управляющий оператор, можно при помощи ключевого слова :FILE:, а имя таблицы, которую вы хотите загрузить - при помощи ключевого слова :TABLE:. Чтобы использовать утилиту LOAD для работы с выходными данными предыдущего примера, задайте для свойств LOAD следующие значения параметров:

Примечание: В поле UTSTMT введите либо оператор загрузки, либо имя файла, который был выводом утилиты REORG с опцией UNLOAD EXTERNAL.

Таблица 19. Свойства шага LOAD

UTILITY_ID	LOADREORG
RESTART	NO
UTSTMT	:FILE:DBVW.DSNURELD.PNCHDSN:TABLE:[DBVW].INVENTORY
UTILITY_NAME	LOAD
RECDSN	DBVW.DSNURELD.RECDSN
RECDEVT	SYSDA

Это будет работать с любыми исходными таблицами или таблицами назначения DB2 for OS/390 в той же самой или в другой подсистеме DB2. Плоский файл управляющего оператора может быть либо файлом HFS, либо собственным файлом MVS.

Более подробную информацию о DSNUTILS и об утилитах DB2, доступных для платформы OS/390, смотрите в справочном руководстве DB2 for OS/390 Utility Guide and Reference.

Репликация

Агент OS/390 можно использовать для автоматизации шага применения репликации Data Propagator. Для репликации требуется исходная база данных, управляющая база данных и база данных назначения. Это могут быть как разные, так и одни и те же базы данных. Задание Capture читает журнал DB2 для определения того, какие строки исходной базы данных были добавлены, изменены или удалены, и записывает изменения в таблицу изменений данных. Затем это задание записывает изменения в таблицу изменения данных. Потом выполняется задание Apply для применения изменений к базе данных назначения. Программный пакет Менеджер хранилищ DB2 может автоматизировать выполнение задания Apply путем создания шага репликации. Менеджер хранилищ позволяет определить тип выполняемого задания Apply и время его запуска. Для этого надо включить библиотеку SASNLINK в переменную среды steplib.

Добавление поддержки репликации в шаблон Центра хранилищ данных

Центр хранилищ данных содержит шаблон JCL для поддержки репликации. Если вы собираетесь использовать агент OS/390 для запуска программы Apply, необходимо изменить учетную информацию и информацию о наборах данных в этом шаблоне для системы OS/390. Чтобы изменить шаблон:

1. Зарегистрируйтесь с ID, у которого есть полномочия копирования и изменения файлов в каталоге /usr/lpp/DWC.
2. Найдите файл apply.jcl и сделайте копию этого файла под именем `имя_системы.apply.jcl`, где `имя_системы` - имя системы MVS. Например, если имя вашей системы - STLMVS1, создайте копию этого файла под именем `STLMVS1.apply.jcl`.
3. При помощи текстового редактора исправьте JCL в соответствии с местными требованиями. Измените учетную информацию так, чтобы она соответствовала стандартам учетной информации, и измените наборы данных STEPLIB DD и MSGS DD для вашей системы MVS.

4. При необходимости измените имя программы в строке EXEC. Подробности изменения имен программ смотрите в справочном руководстве *DB2 Replication Guide and Reference*. Не изменяйте параметры в скобках, такие как [USERID] и [APPLY_PARMS]. (Скобки представлены шестнадцатеричными символами x'AD' и x'BD', соответственно. Если для вашего типа терминала TSO в опции SPF 0 не установлено 3278A, вы можете увидеть вместо скобок специальные символы. В этом нет ничего страшного, если не изменять x'AD' и x'BD', а также какие-либо данные между этими символами.)
5. Измените переменную среды VWS_TEMPLATES, чтобы она указывала на каталог, в который скопирован файл шаблона.

В следующем примере показан шаблон JCL, поставляемый с Центром хранилищ данных:

Шаблон JCL Apply:

```
//[USERID]A JOB ,MSGCLASS=H,MSGLEVEL=(1,1),
// REGION=2M,TIME=1440,NOTIFY=&SYSUID
//* НЕ ИЗМЕНЯЙТЕ ПЕРВУЮ СТРОКУ ЭТОГО ШАБЛОНА.
//* ОСТАЛЬНЫЕ СТРОКИ JCL НАДО ИЗМЕНИТЬ В СООТВЕТСТВИИ С ВАШЕЙ СРЕДОЙ
//*****
//* RUN APPLY/MVS ON OS/390 DB2 6.1 *
//*****
//ASNARUN EXEC PGM=ASNAPV66,REGION=10M,
// [APPLY_PARMS]
//STEPLIB DD DISP=SHR,DSN=DPROPR.V6R1M0.SASNLINK
// DD DISP=SHR,DSN=DSN610.SDSNLOAD
//MSGS DD DSN=DPROPR.V2R1M0A.MSGS,DISP=SHR
//ASNASPL DD DSN=&&ASNASPL,DISP=(NEW,DELETE,DELETE),
// UNIT=SYSDA,SPACE=(CYL,(10,1)),
// DCB=(RECFM=VB,BLKSIZE=6404)
//SYSTEM DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//
```

Регистрация агента

Многие из компонентов Менеджера хранилищ DB2, такие как сервер, регистратор, агенты, а также некоторые программы Центра хранилищ данных производят записи в журналы в каталоге регистрации, заданном переменной среды VWS_LOGGING. Эти файлы журнала представляют собой плоские текстовые файлы.

Запись агента в журнал можно запустить из Центра хранилищ данных. На левой панели щелкните правой кнопкой мыши по пункту **Хранилище** и выберите **Свойства**. На вкладке Уровень трассировки измените необходимым образом установки для уровня трассировки. Трассировка агента поддерживает уровни от 0 до 4:

- Уровень 1 - трассировка входов и выходов
- Уровень 2 - уровень 1 плюс отладочная трассировка
- Уровень 3 - уровень 2 плюс трассировка данных
- Уровень 4 - трассировка внутреннего буфера

При установке уровня трассировки выше 1 производительность уменьшится. Трассировку следует включать только для отладки. Информация трассировки сохраняется в файле AGNTxxx.LOG. Информация среды сохраняется в файле AGNTxxx.SET.

Кэширование на клиенте в Windows NT

Если пользователь пытается с использованием действительного маркера и диска совместного пользования обратиться к файлу READ PERM DB, расположенному на компьютере с сервером Windows NT, на котором установлена DB2 Datalinks, этот файл открывается правильно. Однако после этого последующие требования открытия с использованием того же элемента реально не доходят до сервера и обслуживаются из кэша на клиенте. Даже после истечения срока действия маркера содержимое файла остается видимым для пользователя, поскольку запись все еще находится в журнале. Эта ошибка не возникает, если файл расположен на рабочей станции Windows NT.

Ошибка может быть исправлена путем задания нулевого значения для переменной реестра `\\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Lanmanserver\Parameters\EnableOpLocks` на сервере Windows NT. При этом любое обращение к файлу, расположенному на сервере, с рабочей станции клиента через диск совместного пользования всегда будет направляться на этот сервер, а не обслуживаться из кэша клиента. Поэтому маркер проверяется повторно для каждого запроса.

Недостаток этого решения - снижение общей производительности любого доступа к файлам с сервера через диски совместного пользования. Даже при такой установке, если к файлу обращаться через отображение диска совместного пользования непосредственно на сервере, а не с какого-либо компьютера клиента, требование все равно будет обслуживаться из кэша. Поэтому истечение срока годности маркера действовать не будет.

Примечание: Во всех случаях, если доступ к файлу производится локально, а не через диск совместного пользования, задание срока годности маркера и последующее его истечение будет происходить в обычном порядке.

Пробные программные продукты на компакт-дисках с Enterprise Edition для UNIX

На компакт-дисках с DB2 Universal Database (UDB) Enterprise Edition (EE) для платформ на основе UNIX Версии 6 и Версии 7 находится 90-дневная пробная версия DB2 Connect Enterprise Edition (CEE). Поскольку функции DB2 Connect встроены в программный продукт DB2 UDB EE, чтобы пользоваться этими функциями, в системах, в которых установлена DB2 UDB EE, программный продукт DB2 CEE устанавливать не требуется. Если вы установили 90-дневную пробную версию DB2 CEE и хотите обновить ее до лицензированной версии, необходимо купить программный продукт DB2 CEE и установить лицензионный ключ DB2 CEE. Переустанавливать программный продукт не требуется. Указания по установке лицензионного ключа приводятся в книге Quick Beginnings (Быстрый старт) для DB2 EE или DB2 CEE для платформы UNIX.

Если вы установили пробную версию программного продукта CEE при установке EE и не хотите устанавливать CEE постоянно, пробную 90-дневную версию CEE можно удалить, как описано ниже. Если вы удалили пробную версию Connect EE, функции DB2 Connect останутся вам доступны через DB2 EE.

Чтобы удалить DB2 Connect Версии 7, деинсталируйте (в зависимости от платформы) следующие наборы файлов:

- В AIX деинсталируйте набор файлов db2_07_01.clic.
- В NUMA-Q и Solaris Operating Environments деинсталируйте пакет db2clic71.

- В Linux деинсталируйте RPM db2cltc71-7.1.0-x.
- В HP-UX деинсталируйте набор файлов DB2V7CONN.clic.

Чтобы удалить DB2 Connect Версии 6, деинсталируйте (в зависимости от платформы) следующие наборы файлов:

- В AIX деинсталируйте набор файлов db2_06_01.clic.
- В NUMA-Q и Solaris Operating Environments деинсталируйте пакет db2cplic61.
- В Linux деинсталируйте RPM db2cplic61-6.1.0-x.
- В HP-UX деинсталируйте набор файлов DB2V6CONN.clic.

Пробные программные продукты на компакт-дисках с DB2 Connect Enterprise Edition для UNIX

На компакт-дисках с DB2 Connect Enterprise Edition (EE) для платформ на основе UNIX Версии 6 и Версии 7 находится 90-дневная пробная версия DB2 Universal Database (UDB) Enterprise Edition (EE). 90-дневная пробная версия DB2 UDB EE поставляется для ее оценки и не требуется для работы DB2 Connect.

Если вы установили 90-дневную пробную версию DB2 UDB EE и хотите обновить ее до лицензированной версии, необходимо купить программный продукт DB2 UDB EE и установить лицензионный ключ DB2 UDB EE. Переустанавливать программный продукт не требуется. Указания по установке лицензионного ключа приводятся в книге Quick Beginnings (Быстрый старт) для DB2 EE или DB2 CEE для платформы UNIX. Если вы установили пробную версию программного продукта UDB EE при установке Connect EE и не хотите постоянной установки UDB EE, пробную 90-дневную версию EE можно удалить в соответствии со следующими указаниями. Удаление пробной версии DB2 UDB EE не оказывает влияния на работу DB2 Connect EE.

Чтобы удалить DB2 UDB EE Версии 7, деинсталируйте (в зависимости от платформы) следующие наборы файлов:

- В AIX деинсталируйте набор файлов db2_07_01.elic.
- В NUMA-Q и Solaris Operating Environments деинсталируйте пакет db2elic71.
- В Linux деинсталируйте RPM db2elic71-7.1.0-x.
- В HP-UX деинсталируйте набор файлов DB2V7ENTP.elic.

Чтобы удалить DB2 UDB EE Версии 6, деинсталируйте (в зависимости от платформы) следующие наборы файлов:

- В AIX деинсталируйте набор файлов db2_06_01.elic.
- В NUMA-Q и Solaris Operating Environments деинсталируйте пакет db2elic61.
- В Linux деинсталируйте RPM db2elic61-6.1.0-x.
- В HP-UX деинсталируйте набор файлов DB2V6ENTP.elic.

Отбрасывание Менеджера связей данных

Теперь для заданной базы данных можно отбросить Менеджер связей данных DB2. Обработка некоторых относящихся к связям данных требований SQL, а также таких утилит, как резервное копирование/восстановление, включает в себя обмен данными со всеми DLM, сконфигурированными для базы данных. Раньше в DB2 не было возможности отбрасывать сконфигурированный DLM даже тогда, когда он не работал. Это приводило к дополнительной затрате ресурсов при обработке SQL и утилит. После добавления DLM механизм базы данных связывается с ним при

обработке требований, что может приводить к неудачному выполнению некоторых требований SQL (например, отбросить таблицу/табличное пространство/базу данных).

Деинсталляция компонентов DLFM с использованием SMIT может удалить дополнительные наборы файлов

Перед удалением DB2 (Версии 5, 6 или 7) на компьютере с AIX, на котором установлен Менеджер связей данных, выполните следующие шаги:

1. В качестве пользователя root сделайте копию /etc/vfs, используя команду:

```
cp -p /etc/vfs /etc/vfs.bak
```

2. Деинсталлируйте DB2.

3. В качестве пользователя root замените /etc/vfs на резервную копию, сделанную на шаге 1:

```
cp -p /etc/vfs.bak /etc/vfs
```

Ошибка SQL1035N при использовании CLP в Windows 2000

Если DB2 установлена в каталоге, к которому доступ для записи есть только у некоторых пользователей (например, у администраторов), обычный пользователь при попытке использования процессора командной строки DB2 может получить ошибку SQL1035N.

Во избежание этой ошибки DB2 необходимо устанавливать в каталог, к которому доступ для записи есть у всех пользователей.

Усовершенствование SQL Assist

Инструмент SQL Assist теперь позволяет пользователю для объединения таблиц указывать оператор объединения, отличный от "=". Диалог Тип объединения, запускающийся после нажатия на кнопку Тип объединения на странице Объединения инструмента SQL Assist, дополнен включением в него выпадающего списка операторов объединения.

Доступные операции: "=", "<>", "<", ">", "<=" и ">=". SQL Assist представляет собой инструмент, помогающий пользователю при создании простых операторов SQL. Этот инструмент доступен из Командного центра (закладка Интерактивные), Центра управления (диалоговые окна Создать производную таблицу и Создать переключатель), Построителя хранимых процедур (мастер "Вставка хранимой процедуры SQL") и из Центра хранилищ данных (шаг Процесс SQL).

Сообщение об ошибке Windows Millennium Edition (ME) и поисковый сервер NetQuestion

Поисковая служба HTML, NetQuestion, не будет работать в том виде, в каком она установлена в Windows ME. При перезагрузке сервера вы получите следующее сообщение об ошибке:

Проверьте правильность имени поисковой службы

Если вы хорошо знакомы с NetQuestion и с процессом загрузки Windows ME, эту ошибку можно исправить, отредактировав файл **autoexec.bat**; в нем надо заменить строку, выглядящую примерно так:

```
IF EXIST C:\Program Files\IBM\IMNMQ\IMNENV.BAT CALL C:\Program Files\IBM\IMNMQ\IMNENV.BAT
```

| на содержимое файла **IMNENV.BAT** (расположенного в системе по пути, указанному
| в предыдущей строке).

Часть 10. Дополнительная информация

Глава 34. Дополнительная информация

Электронная поддержка DB2 Universal Database и DB2 Connect

Самый свежий и полный источник информации по DB2, в том числе по тем проблемам, которые стали известны после выпуска этого документа - это Web-сайт электронной поддержки DB2 Universal Database & DB2 Connect, расположенный по адресу: <http://www.ibm.com/software/data/db2/udb/winos2unix/support>.

DB2 Magazine

Чтобы получать самую свежую информацию о продуктах семейства DB2, подпишитесь на журнал "DB2 magazine" (подписка бесплатная). Электронная версия этого журнала доступна по адресу <http://www.db2mag.com>; на этом же сайте можно найти инструкции о том, как запросить подписку.

Часть 11. Приложения

Приложение. Замечания

IBM может предлагать описанные продукты, услуги и возможности не во всех 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:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

Следующий абзац не применяется в Великобритании или в любой другой стране, 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 ОГРАНИЧИВАЯСЬ 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.

Любые ссылки в данной информации на Web-сайты, не принадлежащие IBM, приводятся только для удобства и никоим образом не означают поддержки IBM этих Web-сайтов. Материалы этих Web-сайтов не являются частью данного продукта IBM и вы можете использовать их только на собственную 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
Office of the Lab Director
1150 Eglinton Ave. East
North York, Ontario
M3C 1H7
CANADA

Такая информация может быть предоставлена на определенных условиях (в некоторых случаях к таким условиям может относиться оплата).

Лицензированная программа, описанная в данной публикации, 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 измерены 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 может содержать 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.

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. _введите год или годы_. Все права защищены.

Товарные знаки

Следующие термины (они могут быть помечены звездочкой - *) являются товарными знаками корпорации International Business Machines в Соединенных Штатах и/или в других странах:

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager	OS/390
CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/DS
DB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational Database Architecture	SystemView
DRDA	VisualAge
eNetwork	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
	WIN-OS/2

Следующие термины являются товарными знаками или зарегистрированными товарными знаками других компаний:

Microsoft, Windows и Windows NT - товарные знаки или зарегистрированные товарные знаки Microsoft Corporation.

Java, все товарные знаки и логотипы на основе Java и Solaris - товарные знаки Sun Microsystems, Inc. в Соединенных Штатах и/или в других странах.

Tivoli и NetView - товарные знаки Tivoli Systems Inc. в Соединенных Штатах и/или других странах.

UNIX - зарегистрированный товарный знак в Соединенных Штатах и в других странах, его использование лицензируется исключительно фирмой X/Open Company Limited.

Названия других компаний, продуктов и услуг (они могут быть отмечены двойной звездочкой - **) могут быть товарными знаками или марками сервиса других фирм.

Индекс

A

- Address Windowing Extensions (AWE) 84
- Adobe Acrobat Reader
 - вызов в неанглийских версиях 385
- AIX 4.3.3
 - AutoLoader завис после ветвления потоков 388
- API служб резервного копирования
 - поддержка DB2 393
- app_ctl_heap_sz 89
- Archive Log 107
- AS/400
 - программы Capture и Apply 132
- AutoLoader
 - зависание при ветвлении потоков 388

C

- CLI
 - написание программы Unicode 368
 - copy daemon 387

D

- Data Links File System
 - несовместима с Norton Utilities 42
- DataJoiner
 - ограничения 389
- DATALINK
 - ошибки при восстановлении 388
 - репликация 129
- DB2 Connect
 - предварительные требования в Sun Cluster 2.2 40
- DB2 Data Links Manager
 - взаимодействие с восстановлением 69
- DB2 Query Patroller, схема 203
- DB2 Server for VM and VSE
 - управление базами данных из Центра управления 147
- DB2_INDEX_2BYTEVARLEN 53
- db2ArchiveLog 214
- db2expln
 - выполнение 96
- db2start
 - в Windows 95, 98 и ME 137
- DFS Client Enabler
 - деинсталляция 387
- dlfm client_conf
 - reasons for failure 387
- dqpcrt, команда 203, 204
- dqpdrop, команда 203, 206
- dqplist, команда 206
- dqpsetup, команда 203, 204
- dynexpln
 - выполнение 96

G

- GRANT
 - CREATE ON SEQUENCE 310

H

- HP и Sun Solaris
 - поддержка резервного копирования и восстановления 66
- HP-UX
 - нет поддержки Query Enabler 392
- HP-UX 11
 - окна памяти 385

I

- ID пользователей
 - предельное число символов 41
- Internet Explorer
 - ошибки Центра управления 12

J

- JDBC
 - указатели с возможностью прокрутки, доступ 253
- JDK 1.1
 - путь установки 151

L

- Linux
 - задание среды для Центра управления 18
 - параметры национальных версий для японского и упрощенного китайского 11

M

- maxappls 89
- mon_heap_sz 89

N

- NetQuestion
 - ошибка в Windows ME 410
- Netscape
 - использование для доступа к информации электронной справки 8
 - сообщения об ошибках 153, 384
- NEXTVAL 52
- Norton Utilities
 - несовместимы с Data Links File System 42
- NUMA-Q
 - нет поддержки Query Enabler и Tracker 392

O

- ODBC
 - направления выборки 253

- ODBC (*продолжение*)
 - указатели с возможностью прокрутки, доступ 253
- OS/2
 - уровень JDK для сообщений 233
 - установка Центра управления 148

P

- PREVVAL 52

Q

- Query Enabler
 - не поддерживается в HP-UX и NUMA-Q 392
- Query Patroller
 - потеря привилегий пользователей после перенастройки 392
- Query Patroller Tracker
 - не поддерживается в NUMA-Q 392
- Quick Tour 383

R

- Red Flag Linux
 - версия для упрощенного китайского 11

S

- SNA SPM
 - проверка запуска после перезагрузки системы 390
- softmax 90
- SQL, составной 252
- Sun Cluster 2.2
 - предварительные требования DB2 Connect 40
- Sun Solaris и HP
 - поддержка резервного копирования и восстановления 66
- Sybase
 - несовместимость с DB2 on Windows 12

U

- Unicode
 - база данных и программы 361
 - в программах ODBC 373
 - изменения в документации 363
 - функции в 368

V

- VI
 - настройка для работы с DB2 74

W

- Windows
 - несовместимость между Sybase и DB2 12
- Windows 2000
 - перенастройка 17
 - поддержка протокола IPXX/SPX 17
 - рекомендуемые браузеры Web 8
- Windows ME
 - ошибка при запуске NetQuestion 410

A

- авторизация
 - предоставление привилегии на создание последовательности 310
 - привилегия на создание последовательности для всех пользователей 310
- анализатор репликации 131
- архивирование журналов по требованию 65
- архивирование и восстановление программы обработчика пользователя 73
- аутентификация доменом при помощи DB2DOMAINLIST 387

Б

- блокировка, оптимистичная 253
- браузеры Web
 - рекомендации для Windows 2000 8

В

- возврат значения столбца идентификации функция IDENTITY_VAL_LOCAL 336
- возможности доступности 7
- восстановление
 - взаимодействие с менеджером связей данных DB2 69
 - инкрементное 59
 - ошибки столбцов DATALINK 388
 - параллельное 62
- восстановление после аварии при помощи инструмента db2inidb 58
- вызов последовательности 330
- выражение
 - операция конкатенации 367
 - строка 367
- выражение NEXTVAL 330
- выражение PREVVAL 330
- выражения
 - NEXTVAL 52
 - PREVVAL 52
 - последовательности 330
- высокая доступность 58

Д

- двойная регистрация 65
- деинсталляция DB2 DFS Client Enabler 387

- дешифрование информации
 - функция DECRYPT 298
- динамические составные операторы 78
- динамический составной оператор 341
- доступ удаленного клиента
 - конфигурирование Communication Server для NT 388
- доступность 391

Ж

- журнал
 - создание зеркальной копии 65
- журналы
 - архивирование по требованию 65
- журналы - архивирование по требованию 65

З

- запуск DB2
 - использование db2start в Windows 95, 98 и ME 137
- значения национальных версий в программах, использующих CLI 46

И

- изоляция уровня операторов 77
- имена схем
 - предельное число символов 41
- именованные конвейеры при резервном копировании 62
- индексы
 - ключи большого размера 78
 - обращение к нескольким 84
- инкрементное резервное копирование и восстановление 59
- инструмент db2inidb 58
- интерфейс CLI
 - значения кодовых страниц 46

К

- клиенты
 - загрузка 17
 - трехуровневая среда 253
- клиенты Unicode
 - ограничения перемещения данных 374
- клоны баз данных
 - создание 58
- ключи индексов
 - большие 78
- ключи разделения
 - изменение 39
- кодовые страницы
 - в программах, использующих CLI 46
 - для различных национальных версий 43
 - языки стран Балтии 46
- команда db2iupdt, запуск после обновления DB2 18
- команда iwh2exp2
 - новая опция 392
- команда, синтаксис 204
- команды
 - dqpcrt 203, 204

- команды (*продолжение*)
 - dqpdrop 203, 206
 - dqplist 206
 - dqpsetup 203, 204
- команды установки вручную 204
- конкатенация
 - операция 367
- контейнеры
 - добавление к табличному пространству SMS 54
- контейнеры DMS
 - параллельное создание и расширение 49

М

- мастер по созданию баз данных
 - задание размера экстенда 155
- Менеджер каталогов данных на иврите 389
- утилита 197
- методы Java
 - в операторах CREATE PROCEDURE или CREATE FUNCTION 249
- многораздельные базы данных
 - ошибки при добавлении узлов 9
- монитор производительности 49

Н

- назначение строк столбцам, правила для 365
- назначения
 - строки, основные правила для 365
- национальная версия
 - для японского и упрощенного китайского в Linux 11
 - совместимость между сервером администратора и экземпляром 46, 391
- национальные версии
 - национальные языковые версии 43
 - упрощенный китайский для Red Flag Linux 11
- неработоспособные триггеры
 - подробное описание 321

О

- обращение к нескольким индексам 84
- окна памяти
 - в HP-UX 11 385
- операнд
 - строка 367
- оператор ALTER SEQUENCE
 - подробное описание 306
- оператор CREATE PROCEDURE динамический составной
 - оператор 341
 - оператор FOR 346
 - оператор RETURN 348
 - оператор SIGNAL 350
 - оператор процедуры SQL 344
- оператор CREATE SEQUENCE
 - подробное описание 301
- оператор FOR 346

оператор GRANT (привилегии для последовательности)
 подробное описание 310
 оператор RETURN 348
 оператор SET CONSTRAINTS
 заменен на SET INTEGRITY 43
 оператор SET ENCRYPTION PASSWORD
 подробное описание 311
 Оператор SET ENCRYPTION PASSWORD 57
 оператор SET INTEGRITY
 заменяет SET CONSTRAINTS 43
 оператор SET переменная
 подробное описание 326
 оператор SIGNAL 350
 операторы SQL
 ALTER SEQUENCE 306
 CREATE SEQUENCE 301
 GRANT (привилегии для последовательности) 310
 SET ENCRYPTION PASSWORD 311
 SET переменная 326
 операторы SQL триггеров
 оператор SET переменная 326
 операционная система Solaris
 64-битная
 конфигурация 141
 ограничения 141
 отсутствие поддержки версий до UltraSparc 8
 поиск электронной информации 9
 уровень исправлений для V2.6 8
 отделение зеркальной копии
 оперативное 58
 отделенная зеркальная копия
 как резервная база данных 58
 ошибка "доступ к файлу запрещен" 148
 ошибка "неверный ярлык" 148

П

параллельное восстановление 62
 параметр конфигурации
 MIN_DEC_DIV_3 88
 параметры конфигурации
 app_ctl_hear_sz 89
 maxappls 89
 maxlocks 87
 mon_heap_sz 89
 softmax 90
 переменная реестра DB2_AWE 85
 переменная реестра DB2CODEPAGE
 в программах, использующих CLI 46
 переменные SQL 341
 переменные реестра
 DB2_AWE 85
 DB2DOMAINLIST 387
 для поддержки API служб резервного копирования 393
 новые и измененные 92
 перемещение данных
 ограничивается клиентами
 Unicode 374
 перенастройка
 в Windows 2000 17
 запуск db2iupdt для обновления экземпляров 18
 ошибки в ходе 11

перенастройка (*продолжение*)
 проблемы с производными таблицами 17
 поддержка NLV
 Unicode 361
 поддержка национальных языков 46
 поддержка протокола IPX/SPX в Windows 2000 17
 поддержка языков с двумя направлениями письма
 с Центром управления в Windows NT 147
 подпрограммы
 процедуры 333
 поиск HTML
 ошибка в Windows ME 410
 поиск search 385
 поиск электронной информации в операционной системе Solaris 9
 порог кучи сортировки 87
 последовательности 53
 вызовы 330
 выражение NEXTVAL 330
 выражение PREVVVAL 330
 изменение 55
 отбрасывание 55
 привилегии 56
 создание 51
 последовательности, описание 243
 правила именования 42
 предельное число символов 41
 правила преобразования
 для операций, комбинирующих строки 367
 для сравнения строк 367
 предварительные требования для установки
 для UDB EE и CEE в Linux for OS/390 19
 преобразование символов
 правила для операций, комбинирующих строки 367
 правила при сравнении строк 367
 преобразования ADT 259
 привилегии
 USAGE 56
 привилегия USAGE 56
 прикладные программы Unicode ODBC 373
 программы Capture и Apply
 в AS/400 132
 запуск из прикладных программ 133
 сообщения об ошибках 133
 программы предварительной выборки при параллельном создании и расширении контейнеров DMS 49
 производные таблицы
 недоступны после перенастройки 17
 производные таблицы каталога SEQUENCES 352
 процедура SQL
 динамический составной оператор 341
 оператор DECLARE 341
 оператор FOR 346
 оператор RETURN 348
 оператор SIGNAL 350

процедура SQL (*продолжение*)
 переменные 341
 процессы, остановка переда установкой 18
 пулы буферов
 AWE 84

Р

размер кучи сортировки 86
 размер экстенда
 задание в мастере по созданию баз данных 155
 распределенные базы данных
 конфигурирование изменения 39
 резервное копирование
 автономное
 из отделенной зеркальной копии 63
 в именованные конвейеры 62
 инкрементное 59
 рекомендация по заданию dl_expint 88
 репликация
 DATALINK 129
 диагностика ошибок 131
 для серверов других типов (не IBM) 130
 ограничения на имена таблиц и столбцов 129
 планирование 131
 предварительные условия для повсеместного изменения 130
 сценарии 130
 реплицируемые таблицы сводок 39, 82

С

связывание утилит баз данных при помощи клиента времени выполнения 252
 связывание файлов программ 203
 связывание файлов связывания программ вручную 203
 сервер
 установка вручную 202
 сервер SNA
 доступ к UDB из прикладных программ хоста или AS/400 390
 символьная строка
 назначение, обзор 365
 синтаксис команды 204
 системы объединения, ограничения 389
 скалярная функция
 DAYOFWEEK_ISO 255
 скалярная функция WEEK_ISO 255
 скалярные функции
 DAYOFWEEK_ISO 255
 WEEK_ISO 255
 соединение для проверки многоузлового изменения
 несовместимость версий экземпляров 148
 создание зеркальной копии журналов 65
 создание схемы DB2 Query Patroller вручную 203
 сообщения
 программ Capture и Apply 133

сообщения об ошибках
 SQL10012N 151
 во время первоначальной загрузки в
 Windows ME 410
 доступ к файлу запрещен 148
 неверный ярлык 148
 при выполнении триггеров 322
 при добавлении узлов к
 многораздельным базам данных 9
 при перенастройке 11
 проверьте имя поисковой службы 410
 файлы связывания не найдены 385
 составной SQL, использование 252
 составной оператор (динамический)
 переменные 341
 составные операторы
 динамические 78
 список блокировок
 параметр конфигурации maxlocks 87
 средства GUI
 ошибки связывания 385
 статистика подэлементов 80
 статистика системного каталога
 сбор статистики распределения 79
 статическое профилирование
 JDBC/ODBC/CLI, ограничение 256
 статическое профилирование, в
 программах JDBC/ODBC/CLI 256
 столбец
 назначения строк, основные правила
 для 365
 столбец идентификации
 изменение 54
 столбцы идентификации (IDENTITY) 53
 строка
 выражение 367
 операнд 367
 структуры таблиц 133
 схема DB2 Query Patroller 203

Т

табличные пространства
 перевод в состояние ONLINE 55
 табличные пространства DMS
 параллельное создание и изменение
 размеров 56
 табличные пространства SMS
 добавление контейнеров 54
 типы данных
 расширение в базе данных
 Unicode 364
 триггеры
 неработоспособные 321
 сообщения об ошибках 322

У

указатели
 атрибуты, по умолчанию 253
 возможность изменения 253
 возможность прокрутки 253
 одновременность 253
 одновременность значений 253
 с возможностью прокрутки
 OS/390 253
 на стороне сервера 253

указатели (продолжение)
 с возможностью прокрутки
 (продолжение)
 статический 253
 управляемый 253
 устранение неисправностей
 программ 254
 тип 253
 управляемый
 изменение 253
 чувствительность 253
 упрощенный китайский
 параметры национальной версии в
 Linux 11
 уровни изоляции
 в операторе DELETE 323
 в операторе INSERT 324
 в операторе SELECT 290
 в операторе SELECT INTO 325
 в операторе UPDATE 329
 уровень операторов 77
 уровни исправлений
 в Solaris Версии 2.6 8
 условие SWITCH ONLINE 55
 установка
 вручную 202
 зависание при использовании съемного
 диска 11
 команды 204
 остановка процессов DB2 во время 18
 учетная запись, необходимая в
 Windows 391
 установка вручную
 сервер 202
 утилита DB2MCS, перезагрузка
 компьютера для задания PATH 40

Ф

файл конфигурации утилиты ограничения
 ресурсов 86
 файлы связывания программ 203
 функции
 DECRYPT 57
 ENCRYPT 56
 GETHINT 57
 в базе данных Unicode 368
 процедуры 333
 скалярная
 DECRYPT 298
 ENCRYPT 296
 GETHINT 299
 IDENTITY_VAL_LOCAL 336
 функции CLI
 SQLBindFileToParam 257
 SQLNextResult 257
 функции Unicode CLI
 обработка строк 251
 функция CLI SQLBindFileToParam,
 поправка 257
 функция DECRYPT
 значения и аргументы 298
 подробное описание 298
 функция ENCRYPT
 значения и аргументы 296
 подробное описание 296
 функция GETHINT
 значения и аргументы 299

функция GETHINT (продолжение)
 подробное описание 299
 функция IDENTITY_VAL_LOCAL
 значения и аргументы 336
 подробное описание 336
 функция SQLNextResult 257
 функция приостановленного ввода-вывода
 поддержка непрерывной
 доступности 58

Х

хранимые процедуры
 в Центре хранилищ данных 392
 хранимые процедуры CLI
 автоматическое связывание 255
 ограничения 255

Ц

центр оповещения 49
 Центр управления
 записная книжка Параметры
 инструментов 151
 запуск переведенных версий 45
 использование для управления базами
 данных VM и VSE 147
 исправление для OS/390 150
 на платформе OS/390 148
 ограничения на одновременный запуск
 апплетов 151
 ошибки при использовании Internet
 Explorer 12
 поддержка двух направлений письма
 в Windows NT 147
 проблемы с Центром сценариев в
 Windows 95 152
 проблемы со всплывающими
 справками в OS/2 150
 устранение неисправностей 150
 устранение неисправностей на
 платформах UNIX 150
 Центр хранилищ данных
 использование хранимых
 процедур 392
 синтаксис команды iwthexp2 392

Ш

шифрование
 данных 56
 шифрование данных 56
 шифрование информации
 функция ENCRYPT 296
 функция GETHINT 299
 шлюзы
 трехуровневая среда 253

Я

языки стран Балтии
 поддержка кодовых страниц 46
 японский
 параметры национальной версии в
 Linux 11



Напечатано в Дании