

IBM DB2 Connect



Руководство пользователя

Версия 7

IBM DB2 Connect



Руководство пользователя

Версия 7

Перед тем как использовать данный документ и продукт, описанный в нем, прочтите общие сведения под заголовком Приложение Н, “Замечания” на стр. 223.

Этот документ содержит информацию, которая является собственностью IBM. Она предоставляется в соответствии с лицензионным соглашением и защищена законами об авторском праве. Информация в данной публикации не включает никаких гарантий на продукт и никакое из утверждений в данном руководстве не следует понимать подобным образом.

Чтобы заказать публикации, обратитесь к вашему представителю IBM или в местное отделение IBM, или позвоните по телефону 1-800-879-2755 в Соединенных Штатах или 1-800-IBM-4YOU в Канаде.

Отсылая информацию IBM, вы тем самым даете IBM неисключительное право использовать или распространять эту информацию любым способом, как фирма сочтет нужным, без каких-либо обязательств перед вами.

Содержание

Об этой книге	vii	Серверы прикладных программ и DB2 Connect	34
Для кого предназначена эта книга	vii	DB2 Connect и конфигурации сервера прикладных программ	35
<hr/>			
Часть 1. Введение в DB2 Connect	1	Использование DB2 Connect с мониторами транзакций	36
Глава 1. Обзор DB2 Connect	3	Примеры мониторов транзакций	38
Понятие базы данных	4	Tuxedo и DB2 Connect	39
Установка DB2 Connect	5	Модель распределенной обработки транзакций (DTP) X/Open	39
DB2 Connect и SQL	6	Как использовать DB2 Connect с менеджером транзакций стандарта XA	39
Утилиты администратора	6		
Глава 2. Основные понятия архитектуры распределенных реляционных баз данных (DRDA)	9	Глава 4. Программирование в среде DB2 Connect	41
DRDA и DB2 Connect	9	Программирование в распределенной среде	41
Удаленная единица работы	11	Использование языка определения данных (DDL)	42
Распределенное требование	12	Использование языка манипулирования данными (DML)	43
Изменение информации на нескольких узлах (двухфазное принятие)	13	Использование языка управления данными (DCL)	44
Сценарии многоузлового изменения для хоста и AS/400, требующие SPM	15	Соединение и разъединение	44
DRDA и доступ к данным	18	Прекомпиляция	45
Использование Центра управления для разрешения многоузловых изменений	18	Определение порядка сортировки	48
Электронные публикации, связанные с DRDA	19	Поддержание реляционной целостности	48
Глава 3. Сценарии использования DB2 Connect	21	Блокировка	48
Прямой доступ к базам данных	22	Различия в SQLCODE и SQLSTATE	49
DB2 Connect Enterprise Edition в качестве сервера связи	24	Использование системных каталогов	49
DB2 Connect и программы Web	26	Переполнения при численных преобразованиях в процессе присваивания	49
Преимущества и ограничения традиционного программирования с использованием CGI	26	Уровни изоляции	50
DB2 Connect на сервере Web	27	Хранимые процедуры	51
DB2 Connect в качестве сервера прикладных программ Java	28	Составной SQL NOT ATOMIC	53
Net.Data	29	Многоузловое изменение с DB2 Connect	54
IBM WebSphere	30	Операторы SQL сервера хоста или AS/400, поддерживаемые DB2 Connect	55
Использование DB2 Connect с серверами прикладных программ	32	Операторы SQL сервера хоста или AS/400, не поддерживаемые DB2 Connect	55
Сервер прикладных программ	33	Реализация учета оплаты в DB2 Universal Database for OS/390	56
		Отправка учетной информации на сервер DB2 for OS/390	58
		Задание учетной строки	58
		Литература	59

Глава 5. Запуск собственных прикладных программ	61
Связывание утилит баз данных	61
Выполнение программ CLI/ODBC	62
Подробности доступа CLI/ODBC для конкретных платформ	63
Подробная информация о конфигурации	67
Выполнение программ Java	68
Настройка среды	69
Прикладные программы Java	71
Апплеты Java	72

Часть 2. Справочник и устранение неисправностей . . . **73**

Глава 6. Изменение каталогов базы данных	75
Сбор информации	75
Каталог узла	76
Каталог DCS	77
Системный каталог баз данных	85
Задание нескольких записей для одной базы данных	86
Изменение каталогов	86

Глава 7. Связывание прикладных программ и утилит	89
Команда BIND	94
Повторное связывание	94

Глава 8. Монитор системы баз данных	97
Слежение за соединениями с удаленными клиентами	97
Переключатели монитора для DB2 Connect	98
Вывод состояния переключателей монитора	98
Использование команд GET SNAPSHOT	98
Вывод состояния программы DCS	101
Команда LIST DCS APPLICATIONS	101
LIST DCS APPLICATIONS SHOW DETAIL	102
Команда LIST DCS APPLICATIONS EXTENDED	105
Использование центра управления DB2 для вывода расширенной информации о программах DCS	106
Использование монитора производительности Windows	107

Глава 9. Утилиты администратора	109
--	------------

Процессор командной строки	109
Использование утилит импорта и экспорта	110
Передача данных с рабочей станции на сервер баз данных S/390 или AS/400	111
Передача данных с сервера DRDA на рабочую станцию	111
Смешанные однобайтные и двухбайтные данные	111
Замена для утилиты SQLQMF	112

Глава 10. Защита	113
Аутентификация	113
Типы защиты	115
Типы защиты для соединений APPC	115
Типы защиты для соединений TCP/IP	117
Обсуждение типов защиты	117
Изменение пароля MVS	118
Конфигурирование рабочей станции DB2 Connect для работы с программой управления сроками действия паролей	119
Конфигурирование хоста для поддержки управления сроками действия паролей	120
Дополнительные советы, касающиеся защиты	120
Расширенные коды защиты	120
Защита TCP/IP уже проверена	121
Защита программ ODBC и Java на настольных системах	121
Поддержка изменения пароля	121

Глава 11. Отображение SQLCODE	123
Отключение отображения SQLCODE	123
Оптимизация отображения SQLCODE	123

Глава 12. Производительность	129
Производительность: основные понятия и средства	129
Потоки данных	129
Критические участки	131
Тестирование производительности	131
Средства определения производительности	131
Оптимизация доступа ODBC	133
Разработка программ	134
Составные операторы SQL и хранимые процедуры	134
Группировка требований	135
Логика предикатов	135
Блокирование данных	135
Статические и динамические операторы SQL	137
Другие особенности SQL	137

Настройка DB2 Connect	138
RQRIOLBK	138
DIR_CACHE	138
Другие параметры DB2 Connect	139
Поддержка пула соединений	140
Как работает пул соединений	140
Концентратор соединений DB2 Connect	141
Настройка баз данных	146
Настройка сети	149
Конкуренция за системные ресурсы	153
Поиск причин низкой производительности	153
Дополнительные советы и замечания по настройке производительности SNA	154
Общая информация о производительности для DB2 Connect	154
Выбор и настройка сетевого подключения	154
Другие источники информации о производительности DB2 Connect	156
Поддержка Multi Path Channel для SNA с использованием ESCON	156
Как настраивать соединения DB2 Connect через NCP	156
Сведения об усовершенствованиях OSA-2	160
Другие источники информации	162
Другие публикации	162
Источники в WWW	162
Дополнительные советы и подсказки пользователям SNA	162
Глава 13. Диагностика ошибок	163
Другие источники информации	163
Руководство Troubleshooting Guide	163
Источники в WWW	163
Документация по смысловым кодам APPC, CPI-C и SNA	163
Сбор информации	164
Ошибка при начальном соединении	164
Ошибки, возникающие после начального соединения	165
Средства диагностики	167
Утилита трассировки (ddcstrc)	167
Синтаксис команды трассировки	168
Параметры трассировки	169
Вывод трассировки	170
Анализ выходного файла трассировки	171
Часто встречающиеся ошибки при работе с DB2 Connect	176
SQL0965 или SQL0969	177
SQL1338 при выполнении оператора CONNECT	177

SQL1403N при выполнении оператора CONNECT	178
SQL5043N	179
SQL30020	180
SQL30060	180
SQL30061	180
SQL30073 с кодом возврата 119C при выполнении оператора CONNECT	181
SQL30081N с кодом возврата 1	182
SQL30081N с кодом возврата 2	183
SQL30081N с кодом возврата 9	183
SQL30081N с кодом возврата 10	184
SQL30081N с кодом возврата 20	184
SQL30081N с кодом возврата 27	185
SQL30081N с кодом возврата 79	185
SQL30081N с кодом ошибки протокола связи 10032	186

Часть 3. Приложения **187**

Приложение А. Функции, введенные в предыдущих выпусках	189
DB2 Connect, Версия 6 Выпуск 1	189
DB2 Connect, Версия 5 Выпуск 2	189
DB2 Connect, Версия 5.0	190
DDCS, Версия 2 Выпуск 4	192
DDCS, Версия 2 Выпуск 3	193

Приложение В. Листок настройки каталогов	195
---	------------

Приложение С. Вопросы поддержки национальных языков	197
Преобразование символьных данных	197

Приложение D. Использование служб каталога DCE	201
Создание объекта базы данных	202
Создание объекта локатора базы данных	204
Создание объекта маршрутизации информации	206
Задание параметров конфигурации	207
Каталогизация базы данных	208
Защита с использованием служб каталога DCE	208

Приложение Е. Связывание утилит для клиентов ранних версий	213
---	------------

Приложение F. Настройка производительности прикладных программ CLI/ODBC с помощью параметра CLISCHEMA	215
Среда назначения	215
CLI/ODBC	215
Параметр инициализации DB2 CLISCHEMA	216
Замечания по использованию	217
Утилиты db2cli и bldschem	217
Рекомендуемый подход	219
Дополнительные советы и указания	219
Оптимизатор каталога db2ocat	220
Дополнительные источники информации	220

Приложение G. Дополнительные источники информации	221
Другие публикации на темы, связанные с данной	221
Приложение H. Замечания	223
Товарные знаки	225
Индекс	227
Как связаться с IBM	229
Информация о продукте	229

Об этой книге

Эта книга содержит общую информацию об использовании следующих продуктов IBM DB2 Connect:

- DB2 Connect Personal Edition для OS/2 и 32-битных операционных систем Windows
- DB2 Connect Enterprise Edition (EE) для AIX, HP-UX, Linux, PTX, Solaris, OS/2 и 32-битных операционных систем Windows
- DB2 Connect Unlimited Edition for OS/390

Эта книга делится на три части:

- Часть 1, Введение в DB2 Connect, где содержится концептуальный обзор DB2 Connect, Distributed Relational Database Architecture (DRDA) и возможные сценарии использования.
- Часть 2, Справочник и устранение неисправностей, где излагается информация об изменении каталогов баз данных, связывании программ, утилитах управления, DB2 System Monitor, защите, диагностике ошибок и производительности.
- Часть 3. Приложения, содержащие различную информацию, подсказки и советы.

В этой книге объясняются также понятия, применимые ко всем продуктам DB2 Connect. Информацию по конкретным платформам смотрите в следующих книгах:

- *DB2 Connect Personal Edition. Быстрый старт* – конфигурирование однопользовательской DB2 Connect в OS/2 и 32-битных средах Windows.
- *DB2 Connect Personal Edition for Linux Quick Beginnings* – конфигурирование однопользовательской DB2 Connect в Linux.
- *DB2 Connect Enterprise Edition for OS/2 and Windows Quick Beginnings* – конфигурирование многопользовательского шлюза DB2 Connect в OS/2 или 32-битных средах Windows.
- *DB2 Connect Enterprise Edition for UNIX Quick Beginnings* – конфигурирование многопользовательского шлюза DB2 Connect в AIX, HP-UX, Linux, PTX или Solaris.

Для кого предназначена эта книга

Эта книга предназначена для программистов и администраторов, отвечающих за конфигурирование и поддержание связей DB2 Connect. Такие связи можно устанавливать между клиентами DB2 и любой из следующих СУБД сервера прикладных программ DRDA:

- DB2 Universal Database for OS/390 Версии 5 или более новой
- DB2 for MVS Версии 3 или более новой
- DB2 for VSE & VM
- DB2 Universal Database for AS/400
- Любой другой реляционной СУБД, в которой реализована функция сервера прикладных программ DRDA.

Примечания:

1. DB2 Universal Database (DB2 UDB) не требует DB2 Connect, чтобы программы хоста или AS/400 могли обращаться к данным DB2 UDB.
2. DB2 Universal Database for OS/390 Версии 5.1 или более новой необходима для использования функций DRDA уровня 3, в том числе TCP/IP-соединений баз данных и хранимых процедур с многострочными наборами результатов.
3. DB2 Universal Database for OS/390 Версии 6.1 или более новой необходима для использования функций DRDA уровня 4, поддерживаемых в DB2 Connect. В число таких функций входят поддержка длинных целых, больших объектов, ID строк и особых типов данных.

Часть 1. Введение в DB2 Connect

Глава 1. Обзор DB2 Connect

DB2 Connect обеспечивает быструю и надежную связь с базами данных на мэйнфреймах IBM для прикладных программ, работающих в различных операционных системах (в том числе на основе UNIX).

DB2 Connect предлагает несколько вариантов соединения. DB2 Connect Personal Edition обеспечивает прямое соединение с базами данных хоста или AS/400, а DB2 Connect Enterprise Edition – не прямое соединение, при котором клиенты могут обращаться к базам данных хоста или AS/400 через сервер DB2 Connect. DB2 Connect Unlimited Edition – уникальное комплексное решение, упрощающее выбор продукта и лицензирование.

DB2 Connect Enterprise Edition

DB2 Connect Enterprise Edition представляет собой сервер обеспечения связи, который концентрирует соединения нескольких настольных клиентов и сетевых прикладных программ с серверами баз данных DB2, работающими в системах хоста или AS/400, и управляет ими. Базы данных DB2 for AS/400, DB2 for OS/390 и DB2 for VSE & VM корпорации IBM по-прежнему используются в крупнейших организациях для обработки важнейших данных. Обработка данных ведется в этих базах данных хоста и AS/400, но возникает насущная необходимость интеграции этих данных в прикладные программы рабочих станций Windows, UNIX и OS/2.

DB2 Connect Enterprise Edition позволяет локальным и удаленным прикладным программам клиентов создавать, изменять и обслуживать реляционные базы данных, а также управлять ими, используя язык структурированных запросов (SQL), API (интерфейсы прикладного программирования) DB2, ODBC (Open Database Connectivity – интерфейс открытого взаимодействия с базами данных), JDBC (Java-интерфейс взаимодействия с базами данных), SQLJ (встроенный язык SQL для Java) или DB2 CLI (Call Level Interface – интерфейс уровня вызовов). Кроме того, DB2 Connect поддерживает интерфейсы данных Microsoft Windows, такие как ActiveX Data Objects (ADO), Remote Data Objects (RDO) и OLE DB.

DB2 Connect Enterprise Edition в настоящее время доступен для AIX, HP-UX, Linux, OS/2, PTX, Solaris и 32-битных систем Windows. Эти серверы поддерживают прикладные программы, выполняемые на рабочих станциях OS/2, UNIX (AIX, HP-UX, Linux, PTX, Solaris, Silicon Graphics IRIX) и 32-битных систем Windows.

DB2 Connect Personal Edition

DB2 Connect Personal Edition обеспечивает доступ с одной рабочей станции к базам данных DB2, расположенным на таких серверах, как MVS/ESA, OS/390, OS/400, VM и VSE, а также на серверах DB2 Universal Database в OS/2, UNIX и 32-битных системах Windows. DB2 Connect Personal Edition предоставляет столь же широкий набор API, как и DB2 Connect Enterprise Edition, а также встроенные средства поддержки SNA на всех платформах Windows.

В настоящее время существуют версии продукта для OS/2, Linux и 32-битных операционных систем Windows.

DB2 Connect Unlimited Edition

DB2 Connect Unlimited Edition – уникальный пакет, обеспечивающий гибкое разворачивание DB2 Connect и упрощающий выбор продукта и лицензирование. Этот продукт содержит DB2 Connect Personal Edition и DB2 Connect Enterprise Edition с лицензионными условиями, допускающими неограниченное разворачивание любого продукта DB2 Connect. Лицензионная плата определяется размером системы System/390, с которой будут работать пользователи DB2 Connect.

Это пакетное предложение доступно только для систем OS/390, и лицензия действует только для источников данных DB2 for OS/390.

Понятие базы данных

Термин *база данных* используется в этой книге для описания систем управления реляционными базами данных (RDBMS). В других системах, с которыми связывается DB2 Connect, термин "база данных" может использоваться в несколько иных значениях. В DB2 Connect термин "база данных" может также означать:

MVS (Версия 4 или более ранняя)

Подсистема DB2 for MVS/ESA, определяемая своим именем LOCATION NAME.

Чтобы узнать LOCATION NAME, зарегистрируйтесь в TSO и выполните при помощи доступных средств запросов следующий запрос SQL:

```
select current server from sysibm.sysdummy1
```

LOCATION NAME указывается также в наборе данных Boot Strap Data Set (BSDS) и в сообщении DSNL004I (LOCATION=имя_положения), появляющемся при запуске DDF.

OS/390 (Версия 5 и более новые)

Подсистема DB2 Universal Database for OS/390, определяемая своим именем LOCATION NAME.

Чтобы узнать LOCATION NAME, зарегистрируйтесь в TSO и выполните при помощи доступных средств запросов следующий запрос SQL:

```
select current server from sysibm.sysdummy1
```

LOCATION NAME указывается также в наборе данных Boot Strap Data Set (BSDS) и в сообщении DSNL004I (LOCATION=имя_положения), появляющемся при запуске DDF.

VSE DB2 for VSE, выполняемая в разделе и определяемая своим DBNAME

VM DB2 for VM, выполняемая на виртуальной машине CMS и определяемая своим DBNAME

OS/400

DB2 Universal Database for AS/400, входящая в операционную систему OS/400. На компьютере AS/400 может существовать только одна база данных. Если база данных будет использоваться прикладными программами, выполняющимися вне системы AS/400, для базы данных должно быть задано имя в каталоге реляционных баз данных. Это имя называют именем реляционной базы данных (Relation Database Name – RDB Name).

Чтобы вывести имя RDB, выполните в вашей системе AS/400 команду **WRKRDBDIRE**. Имя RDB вашей локальной системы будет помечено *LOCAL в столбце Remote Location. Чтобы изменить его, используйте команду CHGRDBDIRE.

Установка DB2 Connect

Прежде чем использовать DB2 Connect, выполните следующие действия:

- Шаг 1. Установите DB2 Connect и сконфигурируйте сервер хоста или AS/400 и связь рабочей станции, как это описано в книге *DB2 Connect Quick Beginnings* или *Дополнение по установке и настройке*.
- Шаг 2. Обновите каталоги баз данных, как описано в разделе Глава 6, “Изменение каталогов базы данных” на стр. 75.

Примечание: В OS/2 и в 32-битных системах Windows для этого рекомендуется использовать Ассистент конфигурирования клиента (CCA).

На всех остальных платформах для обновления каталогов баз данных нужно использовать процессор командной строки (CLP) DB2.

Оба метода описаны в руководстве *Дополнение по установке и настройке*.

Шаг 3. Выполните связывание утилит DB2 Connect с каждой системой управления базами данных хоста или AS/400, как описано в разделе Глава 7, “Связывание прикладных программ и утилит” на стр. 89.

Это также можно сделать, используя CCA или диалоговое окно Настройка источников данных, если они имеются в системе.

DB2 Connect и SQL

DB2 Connect передает операторы SQL от прикладных программ на серверы баз данных хоста или AS/400. DB2 Connect может передавать почти все правильные операторы SQL. Исключения описаны в разделе “Операторы SQL сервера хоста или AS/400, не поддерживаемые DB2 Connect” на стр. 55.

Существует два типа обработки операторов встроенного SQL: статический и динамический. При использовании статических операторов SQL уменьшается время, необходимое для выполнения оператора SQL, так как используется предварительная обработка. Динамические операторы SQL обрабатываются при передаче их на сервер баз данных хоста или AS/400. Динамический SQL обеспечивает большую гибкость, однако производительность обычно снижается. Программист прикладных программ выбирает, какой именно тип операторов SQL использовать – статический или динамический. Оба эти типа поддерживаются DB2 Connect.

Различные серверы баз данных хоста или AS/400 используют разные реализации SQL. Дополнительную информацию об общих операторах SQL, которые поддерживаются всеми системами IBM, смотрите в справочнике *SQL Reference*.

DB2 Connect полностью поддерживают общие операторы SQL IBM, а также реализации SQL в DB2 Universal Database for OS/390, DB2 for MVS/ESA, DB2 for VSE & VM (прежнее название – SQL/DS) и DB2 Universal Database for AS/400. Настоятельно рекомендуется для независимости базы данных использовать IBM SQL. Дополнительную информацию смотрите в разделе Глава 4, “Программирование в среде DB2 Connect” на стр. 41.

Утилиты администратора

Системному администратору DB2 Connect в работе могут помочь следующие утилиты:

- Процессор командной строки позволяет вводить операторы SQL для базы данных сервера баз данных хоста или AS/400. Он передает эти операторы SQL указанной базе данных.

- Командный центр DB2 поддерживает графический интерфейс для процессора командной строки.
- Утилиты импорта и экспорта позволяют загружать, импортировать и экспортировать данные в файл на рабочей станции и из такого файла, в базу данных или из базы данных сервера баз данных хоста или AS/400. Эти файлы можно затем использовать для импорта данных в базы данных, электронные таблицы и другие прикладные программы на рабочей станции. Дополнительную информацию об утилитах импорта и экспорта смотрите в книге *Data Movement Utilities Guide and Reference*.
- Пользователи DB2 Connect Enterprise Edition на Windows NT и Windows 2000 могут использовать программу просмотра событий и монитор производительности. При помощи программы просмотра событий можно смотреть исключительные ситуации, записанные DB2 Connect. Монитор производительности позволяет локально или дистанционно следить за производительностью серверов DB2 Connect и настраивать ее.
- Центр управления DB2 позволяет следить за всеми аспектами работы серверов DB2 Connect и управлять ими. Он также позволяет администраторам работать с объектами баз данных DB2 for OS/390, такими как таблицы, производные таблицы, пулы буферов и потоки. Дополнительную информацию об управлении системами DB2 for OS/390 через Центр управления DB2 смотрите в книге *Application Development Guide*.

Дополнительную информацию об этих утилитах смотрите в разделе Глава 9, “Утилиты администратора” на стр. 109.

Кроме этого, утилита системного монитора баз данных позволяет системному администратору следить за соединениями системы. Она также помогает системному администратору находить источник ошибки. Системный администратор может задать соотношения между прикладными программами клиента и соответствующими заданиями, выполняемыми на сервере баз данных хоста или AS/400. Дополнительную информацию смотрите в разделе Глава 8, “Монитор системы баз данных” на стр. 97.

Глава 2. Основные понятия архитектуры распределенных реляционных баз данных (DRDA)

DRDA (Distributed Relational Database Architecture – архитектура распределенных реляционных баз данных) – это набор протоколов, которые позволяют нескольким системам баз данных как IBM, так и других разработчиков, а также прикладным программам работать совместно. Любой набор продуктов управления реляционными базами данных, использующих DRDA, можно связать с помощью сетевых соединений, чтобы образовалась распределенная система управления реляционными базами данных. DRDA координирует взаимодействие между системами, определяя, чем и как должны обмениваться эти системы.

При описании DB2 Connect мы часто будем использовать термин "единица работы". *Единица работы (unit of work, UOW)* – это одна логическая транзакция. Она представляет собой последовательность операторов SQL, в которой либо все операции выполняются правильно, либо вся последовательность считается выполненной неудачно.

Другое ключевое понятие – распределенная единица работы (многоузловое изменение). В *распределенной единице работы* участвуют несколько серверов баз данных. Конкретнее, мы определяем *многоузловое изменение* как транзакцию со следующими характеристиками:

- За единицу работы на нескольких серверах баз данных производятся изменения.
- Сама программа управляет распределением работы и инициирует принятие.
- В единице работы может быть несколько требований.
- Каждое требование относится к одному серверу управления базами данных.
- Принятие координируется между несколькими серверами баз данных.

Дополнительную информацию о многоузловом принятии смотрите в разделе "Изменение информации на нескольких узлах (двухфазное принятие)" на стр. 13.

DRDA и DB2 Connect

DB2 Connect применяет архитектуру DRDA, чтобы удешевить и упростить доступ к данным, хранящимся на DB2 Universal Database for AS/400, DB2 Universal Database for OS/390, DB2 for MVS/ESA, DB2 for VSE & VM и других совместимых с DRDA серверах баз данных. Используя в полной мере

архитектуру DRDA, DB2 Connect предлагает производительное и недорогое решение, которое удовлетворяет требованиям, предъявляемым потребителями к системному управлению.

В терминологии DRDA *реквестер прикладных программ* – это программный код, обслуживающий тот конец соединения, где находятся прикладные программы (программа, запрашивающая данные); *сервер прикладных программ* – это код, обслуживающий тот конец, где расположена база данных. В среде DB2 Connect рабочая станция DB2 Connect может работать только как реквестер прикладных программ, обслуживающий прикладные программы.

На рис. 1 показан поток данных между рабочей станцией DB2 Connect и сервером DRDA в случае, когда есть только локальные клиенты. Кроме того, для связи между рабочей станцией DB2 Connect и любым удаленным клиентом используется собственный протокол DRDA.

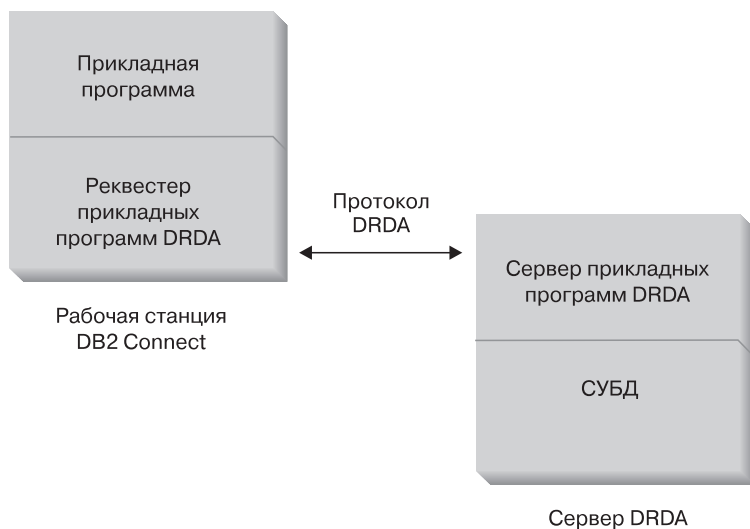


Рисунок 1. Поток данных между рабочей станцией DB2 Connect и сервером DRDA

Для поддержки соединений между системами управления базами данных на сервере DRDA и клиентами баз данных DRDA использует следующие архитектуры:

- CDRA (Character Data Representation Architecture – Архитектура представления символьных данных)
- DDM (Distributed Data Management Architecture – архитектура распределенного управления данными)
- FD:OCA (Formatted Data Object Content Architecture – архитектура содержимого объекта форматированных данных)
- SNA (Systems Network Architecture – системная сетевая архитектура)

- MSA (SNA Management Services Architecture – архитектура служб управления SNA)
- Протокол TCP/IP

Эти структуры используются как строительные блоки. Потоки данных в сети определяются в архитектуре DRDA, которая устанавливает протокол передачи данных для доступа к распределенной реляционной базе данных.

Требование направляется к нужному пункту назначения с помощью каталогов, содержащих различного вида информацию связи, а также имя нужной базы данных на сервере DRDA.

Удаленная единица работы

Удаленная единица работы позволяет пользователю или прикладной программе читать или изменять данные в одном месте в каждой единице работы. За единицу работы поддерживается доступ к одной базе данных. Хотя прикладная программа может изменять несколько удаленных баз данных, внутри одной единицы работы она может работать только с одной базой данных.

Характеристики удаленной единицы работы:

- В одной единице работы можно использовать несколько требований (операторов SQL).
- В одной единице работы можно использовать несколько указателей.
- Каждая единица работы может изменять только одну базу данных.
- Прикладная программа выполняет или принятие, или откат единицы работы. В некоторых ошибочных ситуациях откат единицы работы может выполнять сервер баз данных или DB2 Connect.

Например, на рис. 2 на стр. 12 показан клиент баз данных, на котором выполняется программа обслуживания платежей; она обращается к базе данных с таблицами накопительных и чековых счетов, а также к плану оплаты банковских услуг. Эта программа должна:

- Принять значение переводимой суммы от пользовательского интерфейса.
- Вычесть эту сумму из суммы на накопительном счете и определить новый остаток.
- Прочитать план оплаты, чтобы определить стоимость перевода денег для накопительного счета с данным остатком.
- Вычесть стоимость перевода из суммы на накопительном счете.
- Добавить переводимую сумму к чековому счету.
- Выполнить принятие транзакции (единицы работы).

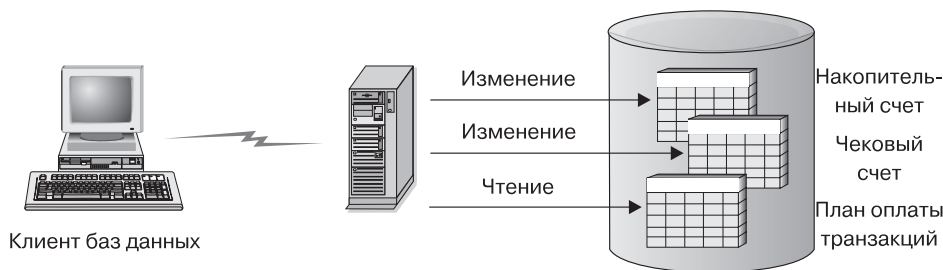


Рисунок 2. Использование в транзакции одной базы данных

Для конфигурирования такой программы надо:

1. Создать таблицы для накопительного счета, чекового счета и плана оплаты в одной базе данных, как описано в книге *Administration Guide*.
2. Для физически удаленной базы данных – задать для сервера баз данных использование подходящего протокола связи, как описано в книгах *Quick Beginnings*.
3. Для физически удаленной базы данных – внести в каталог узел и базу данных, чтобы определить эту базу для сервера, как описано в книгах *Quick Beginnings*.
4. Прекомпилировать прикладную программу, задав соединение типа 1; для этого надо указать CONNECT(1) в команде PREP, как описано в книге *Application Development Guide*.

Распределенное требование

Распределенное требование – это функция распределенных баз данных, которая позволяет программам и пользователям выполнять операторы SQL, обращающиеся (в одном операторе) к нескольким СУБД или к нескольким базам данных. Пример – объединение таблиц из двух различных подсистем DB2 for OS/390.

DB2 Connect Версии 7 поддерживает распределенные требования для баз данных и СУБД. Можно, например, выполнить операцию UNION между таблицей DB2 и представлением Oracle. Поддерживаются СУБД семейства DB2 (DB2 UDB for Windows, UNIX and OS/2, DB2 for OS/390 и DB2 for AS/400), а также Oracle.

Распределенное требование обеспечивает *прозрачность положений* для объектов баз данных. Если информация (в таблицах и производных таблицах) перемещается, можно исправить ссылки (так называемые *псевдонимы*), не внося никаких изменений в сами программы, запрашивающие информацию. Распределенные требования обеспечивают также *компенсацию* для тех СУБД, которые не полностью поддерживают диалект SQL DB2, а также некоторые

возможности оптимизации. Те операции, которые нельзя выполнить в таких СУБД (например, рекурсивные операторы SQL), выполняются в DB2 Connect.

Распределенное требование выполняется *полуавтономным* способом. Например, запросы DB2, содержащие ссылки на объекты Oracle, могут выполняться одновременно с программами Oracle, обращающимися к тому же серверу. Распределенное требование не монополизует и не ограничивает доступ (за исключением ограничений целостности и блокировки) к Oracle или к другим объектам СУБД.

Реализация функции распределенных требований включает в себя экземпляр DB2 Connect Версии 7, базу данных, которая используется как база данных объединения, и один или несколько удаленных источников данных. *База данных объединения* содержит записи каталога, определяющие источники данных и их характеристики. Под *источником данных* понимается СУБД и сами данные. Программы обращаются к базе данных объединения, как и к любой другой базе данных DB2. База данных объединения DB2 Connect не должна управлять пользовательскими данными. Ее единственное назначение – хранить информацию об источниках данных.

Когда система объединения сконфигурирована, к информации источников данных можно обращаться так, как если бы она находилась в одной большой базе данных. Пользователи и программы шлют запросы на одну базу данных объединения, которая по необходимости получает данные от систем семейства DB2 и Oracle. Пользователи и программы задают в запросах псевдонимы; при помощи этих псевдонимов производится обращение к таблицам и производным таблицам на источниках данных. С точки зрения конечного пользователя псевдонимы подобны алиасам.

На производительность распределенных требований влияют многие факторы. Самый важный фактор – обеспечить точность и актуальность информации об источниках данных и объектах, хранимой в глобальном каталоге базы данных объединения. Эта информация используется оптимизатором DB2 и может повлиять на решения об операциях для оценки источников данных. Дополнительную информацию о производительности систем объединения смотрите в книге *Administration Guide: Performance*.

Изменение информации на нескольких узлах (двухфазное принятие)

Изменение информации на нескольких узлах, которое называют также распределенной единицей работы (DUOW) и двухфазным принятием изменений, – это возможность, позволяющая прикладным программам изменять данные на нескольких удаленных серверах баз данных с гарантией сохранения целостности данных. Пример многоузлового изменения – банковская операция, при которой деньги переводятся с одного счета на другой, находящийся на другом сервере баз данных.

При такой операции важно, чтобы изменения, происходящие при снятии денег, были приняты только в том случае, когда приняты изменения, происходящие при занесении денег на счет. Особенности многоузлового изменения сказываются, когда данные, соответствующие этим счетам, хранятся на разных серверах баз данных.

Продукты DB2 полностью поддерживают многоузловые изменения. Эта поддержка действует как для прикладных программ, использующих обычный SQL, так и для прикладных программ, использующих продукты монитора транзакций, которые соответствуют спецификации интерфейса X/Open XA. Примеры таких продуктов – IBM TxSeries (CICS и Encina), IBM Message and Queuing Series, IBM Component Broker Series, IBM San Francisco Project, а также Microsoft Transaction Server (MTS), BEA Tuxedo и некоторые другие. Требования к настройке зависят от того, используется ли многоузловое изменение на основе самого SQL или многоузловое изменение на основе монитора транзакций.

Программы многоузлового изменения – и использующие собственно SQL, и использующие монитор транзакций, – надо прекомпилировать с опциями CONNECT 2 SYNCPOINT TWOPHASE. В обоих случаях оператор SQL CONNECT указывает, какие базы данных надо использовать для последующих операторов SQL. Если монитор транзакций не сообщает DB2, что будет координировать транзакции (для этого он посылает DB2 вызов xa_open для установления соединения с базой данных), то для координации транзакций будет использоваться программное обеспечение DB2.

При использовании многоузлового изменения на основе TP Monitor прикладная программа должна затребовать принятие или откат, используя API TP Monitor, например, CICS SYNCPOINT, Encina Abort(), MTS SetAbort().

Когда используется многоузловое изменение на основе самого SQL, нужно использовать обычные операторы SQL COMMIT и ROLLBACK.

Монитор транзакций при многоузловом изменении может координировать транзакцию, которая обращается к менеджерам ресурсов как DB2, так и других СУБД, например, Oracle, Informix или SQLServer. Многоузловое изменение на основе SQL используется только с серверами DB2.

Для работы транзакции с многоузловым изменением каждая из баз данных, участвующих в распределенной транзакции, должна поддерживать распределенные единицы работы. В настоящее время обеспечивают поддержку распределенных единиц работы и тем самым могут участвовать в распределенных транзакциях следующие серверы:

- DB2 UDB for UNIX, OS/2 и Windows Версии 5 и более новые
- DB2 for MVS/ESA V3.1 и 4.1
- DB2 for OS/390 V5.1
- DB2 Universal Database for OS/390 V6.1 и более новые

- DB2/400 V3.1 и более новые (только с использованием SNA)
- DB2 Server for VM and VSE V5.1 и более новые (только с использованием SNA)
- Database Server 4

Распределенная транзакция может производить изменения на любом наборе поддерживаемых серверов баз данных. Например, прикладная программа может изменять несколько таблиц в DB2 Universal Database на Windows NT или Windows 2000, базу данных DB2 for OS/390 и базу данных DB2/400 в одной транзакции.

Сценарии многоузлового изменения для хоста и AS/400, требующие SPM

Серверам баз данных хоста и AS/400 для участия в распределенной транзакции, начатой PC, UNIX или прикладными программами Web, необходим DB2 Connect. Кроме того, многие сценарии многоузлового изменения, в котором участвуют серверы баз данных хоста и AS/400, требуют, чтобы был сконфигурирован менеджер точек синхронизации (SPM). При создании экземпляра DB2 автоматически конфигурируется DB2 SPM со значениями по умолчанию.

Необходимость использования SPM определяется выбором протокола (SNA или TCP/IP) и использованием монитора транзакций. В следующей таблице приведена сводка сценариев, которые требуют SPM. В этой таблице показано также, что для любого вида доступа к хосту или AS/400 с компьютеров Intel или UNIX требуется DB2 Connect. Кроме того, для многоузловых изменений требуется DB2 Connect SPM, если доступ производится через SNA или используется TP Monitor.

<i>Таблица 1 (Стр. 1 из 3). Сценарии многоузлового изменения хоста и AS/400, требующие SPM</i>				
Используется монитор транзакций	Протокол	Необходим SPM?	Требуемый продукт (выберите один)	Поддерживаемые базы данных хоста и AS/400
Да	TCP/IP	Да	<ul style="list-style-type: none"> • DB2 Connect Enterprise Edition • DB2 Universal Database Enterprise Edition • DB2 Universal Database Enterprise – Extended Edition 	<ul style="list-style-type: none"> • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 или более новая

Таблица 1 (Стр. 2 из 3). Сценарии многоузлового изменения хоста и AS/400, требующие SPM

Используется монитор транзакций	Протокол	Необходим SPM?	Требуемый продукт (выберите один)	Поддерживаемые базы данных хоста и AS/400
Да	SNA	Да	<ul style="list-style-type: none"> • DB2 Connect Enterprise Edition* • DB2 Universal Database Enterprise Edition* • DB2 Universal Database Enterprise – Extended* <p>Примечание: *Только платформы AIX, OS/2, Windows NT и Windows 2000.</p>	<ul style="list-style-type: none"> • DB2 for MVS/ESA V3.1 и 4.1 • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 или более новая • DB2/400 V3.1 или более новая • DB2 Server for VM or VSE V5.1 или более новый
Нет	TCP/IP	Нет	<ul style="list-style-type: none"> • DB2 Connect Personal Edition • DB2 Connect Enterprise Edition • DB2 Universal Database Enterprise Edition • DB2 Universal Database Enterprise – Extended Edition 	<ul style="list-style-type: none"> • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 или более новая

Таблица 1 (Стр. 3 из 3). Сценарии многоузлового изменения хоста и AS/400, требующие SPM				
Используется монитор транзакций	Протокол	Необходим SPM?	Требуемый продукт (выберите один)	Поддерживаемые базы данных хоста и AS/400
Нет	SNA	Да	<ul style="list-style-type: none"> • DB2 Connect Enterprise Edition* • DB2 Universal Database Enterprise Edition* • DB2 Universal Database Enterprise – Extended Edition* <p>Примечание: *Только платформы AIX, OS/2, Windows NT и Windows 2000.</p>	<ul style="list-style-type: none"> • DB2 for MVS/ESA V3.1 и 4.1 • DB2 for OS/390 V5.1 • DB2 Universal Database for OS/390 V6.1 или более новая • DB2/400 V3.1 или более новая • DB2 Server for VM and VSE V5.1 или более новый

Примечание: Распределенная транзакция может производить изменения на любом наборе поддерживаемых серверов баз данных. Например, прикладная программа может изменять несколько таблиц в DB2 UDB на Windows NT, базу данных DB2 for OS/390 и базу данных DB2/400 в одной транзакции.

Дополнительную информацию о двухфазном принятии, а также указания по конфигурированию некоторых популярных мониторов транзакций смотрите в руководстве *Administration Guide*.

Можно также обратиться к Технической библиотеке продуктов и служб DB2 в WWW:

1. Вызовите страницу Web:
<http://www.ibm.com/software/data/db2/library/>.
2. Выберите ссылку **DB2 Universal Database**.
3. Ищите "Technotes", задав ключевые слова "DDCS", "SPM", "MTS", "CICS" и "ENCINA".

DRDA и доступ к данным

Хотя DRDA определяет протоколы связи с базами данных, в ней не определяются программные интерфейсы, или API, которые должны использовать прикладные программисты. Вообще DRDA может использоваться прикладной программой для передачи любого требования, которое может выполнить сервер DRDA назначения. Все имеющиеся на сегодняшний день серверы DRDA могут выполнять требования SQL, направленные прикладной программой через DB2 Connect.

IBM предлагает прикладным программистам средства генерации требований SQL для Windows, OS/2 и некоторых платформ UNIX. Эти средства входят в Клиент разработки программ DB2. Клиент разработки программ DB2 поддерживают различные типы API: встроенный SQL, JDBC и Интерфейс уровня вызовов DB2 (DB2 CLI). Эти API могут использовать программисты, пишущие программы на многих языках программирования. Более подробную информацию об этих API смотрите в руководстве *Application Building Guide*.

Разработчики программ могут также использовать API других авторов. Например, для разработки программ Windows можно использовать Microsoft ODBC и ADO. DB2 Connect содержит драйвер ODBC и провайдер OLE DB, поддерживающие программы, которые используют API ODBC и ADO. IBM не предоставляет средств для разработки прикладных программ ODBC; эти средства предоставляются корпорацией Microsoft.

Использование Центра управления для разрешения многоузловых изменений

Для многоузлового изменения можно использовать Центр управления. Эта простая процедура описана ниже. Дополнительную информацию о процессе конфигурирования многоузлового изменения, в том числе о ручном конфигурировании системы, смотрите в электронной книге *Дополнение по возможностям соединений*.

Запуск мастера по многоузловому изменению

В Центре управления щелкните по знаку [+], чтобы вывести дерево. Правой кнопкой мыши щелкните по экземпляру, который хотите конфигурировать. Появится всплывающее меню. Выберите пункт **Многоузловое изменение** → **Конфигурировать**.

Шаги мастера

Интерфейс мастера подобен записной книжке. На каждой странице этой записной книжки вас будут спрашивать определенную информацию о конфигурации. Ниже эти страницы показаны в том порядке, в котором они появляются.

Шаг 1. Задайте монитор процессора транзакций.

В этом поле выводятся умолчания для монитора транзакций, который вы включили. Если вы не хотите использовать монитор транзакций, выберите **Не использовать монитор ТР**.

Шаг 2. Задайте протоколы связи, которые вы будете использовать.

Шаг 3. Задайте базу данных Менеджера транзакций.

По умолчанию на этой панели выводится первая база данных, с которой вы соединились (1ST_CONN). Можно оставить это значение или выбрать другую внесенную в каталог базу данных.

Шаг 4. Укажите типы серверов баз данных, участвующих в изменении, а также должен ли использоваться только протокол TCP/IP.

Шаг 5. Задайте параметры Менеджера точек синхронизации.

Эта страница появляется, только если на предыдущей странице вы указали, что собираетесь использовать Менеджер точек синхронизации DB2 в сценариях многоузлового изменения.

Проверка возможности многоузлового изменения

Шаг 1. Щелкните по экземпляру правой кнопкой мыши и выберите из всплывающего меню **Многоузловое изменение** → **Проверить**. Откроется окно Проверка многоузлового изменения.

Шаг 2. Выберите базу данных, которую хотите проверить, в списке **Доступные базы данных**. Кнопки со стрелками в центре окна позволяют перемещать выбранные базы данных в подокно **Выбранные базы данных** или из него. Можно также изменять выбранные ID пользователя и пароль, редактируя их прямо в окне **Выбранные базы данных**.

Шаг 3. Закончив выбор, нажмите кнопку **ОК** в нижней части окна. Откроется окно Результаты проверки многоузлового изменения.

Шаг 4. В окне Результаты проверки многоузлового изменения показано, для каких баз данных тест прошел успешно, а для каких – нет. В окне выводятся коды SQL и сообщения об ошибках для баз данных, где произошли ошибки.

Электронные публикации, связанные с DRDA

Полезную информацию, связанную с DRDA, можно найти в следующих электронных книгах.

Для AS/400:

<http://www.as400.ibm.com/db2/v4r4book.htm>

Для OS/390:

<http://www.ibm.com/software/data/db2/os390/library.html>

Для DataJoiner:

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

Электронные публикации по базам данных и управлению данными:

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

Глава 3. Сценарии использования DB2 Connect

DB2 Connect позволяет реализовать множество различных решений для доступа к базам данных хоста или AS/400. В этом разделе описаны основные сценарии, которые можно применять в зависимости от конкретных задач и используемых систем.

DB2 Connect Personal Edition используется для соединения одной рабочей станции, на которой установлена 32-битная операционная система Windows, Linux или OS/2, с базой данных S/390 или AS/400. DB2 Connect Personal Edition лучше всего подходит для случаев, когда поддержку TCP/IP обеспечивает сам сервер баз данных и используются традиционные двухуровневые прикладные программы клиент-сервер.

Например, DB2 Connect Personal Edition хорошо подходит для работы с традиционными двухуровневыми прикладными программами VisualBasic и Microsoft Access. Если для работы прикладных программ требуется промежуточный сервер прикладных программ, нужно использовать DB2 Connect Enterprise Edition. Информацию о размещении программ при использовании DB2 Connect Personal Edition смотрите в разделе “Прямой доступ к базам данных” на стр. 22.

DB2 Connect Enterprise Edition обычно устанавливается на промежуточном сервере для соединения клиентов DB2 с базой данных хоста или AS/400. Его также можно использовать на компьютерах, на которых несколько локальных пользователей должны напрямую обращаться к серверам хоста или AS/400.

Например, DB2 Connect Enterprise Edition можно установить на крупном компьютере, на котором работает много локальных пользователей. Его также можно установить на сервере Web, мониторе транзакций или на других серверах трехуровневых прикладных программ, на которых работают несколько процессов и потоков локальных прикладных программ SQL. В этих случаях DB2 Connect Enterprise Edition можно установить на том же компьютере или на отдельном компьютере, чтобы уменьшить загрузку процессора.

DB2 Connect Enterprise Edition лучше всего подходит для случаев, когда:

- Серверы баз данных хоста и AS/400 сами не поддерживают TCP/IP, а прямое соединение с ними рабочих станций через SNA нежелательно. Смотрите раздел “DB2 Connect Enterprise Edition в качестве сервера связи” на стр. 24.
- Серверы Web выполняют прикладные программы Web. Смотрите раздел “DB2 Connect и программы Web” на стр. 26.

- Серверы Web выполняют прикладные программы Web, использующие апплеты Java.
- В качестве промежуточного уровня используется сервер прикладных программ. Смотрите раздел “Использование DB2 Connect с серверами прикладных программ” на стр. 32.
- Используются мониторы транзакций, такие как CICS, Encina, Microsoft Transaction Server (MTS), Tuxedo, Component Broker и MQSeries. Смотрите раздел “Использование DB2 Connect с мониторами транзакций” на стр. 36.

DB2 Connect Unlimited Edition – уникальный пакет, обеспечивающий гибкое разворачивание DB2 Connect и упрощающий выбор продукта и лицензирование. Этот продукт содержит DB2 Connect Personal Edition и DB2 Connect Enterprise Edition с лицензионными условиями, допускающими неограниченное разворачивание любого продукта DB2 Connect. Лицензионная плата определяется размером системы System/390, с которой будут работать пользователи DB2 Connect. Это пакетное предложение доступно только для систем OS/390, и лицензия действует только для источников данных DB2 for OS/390.

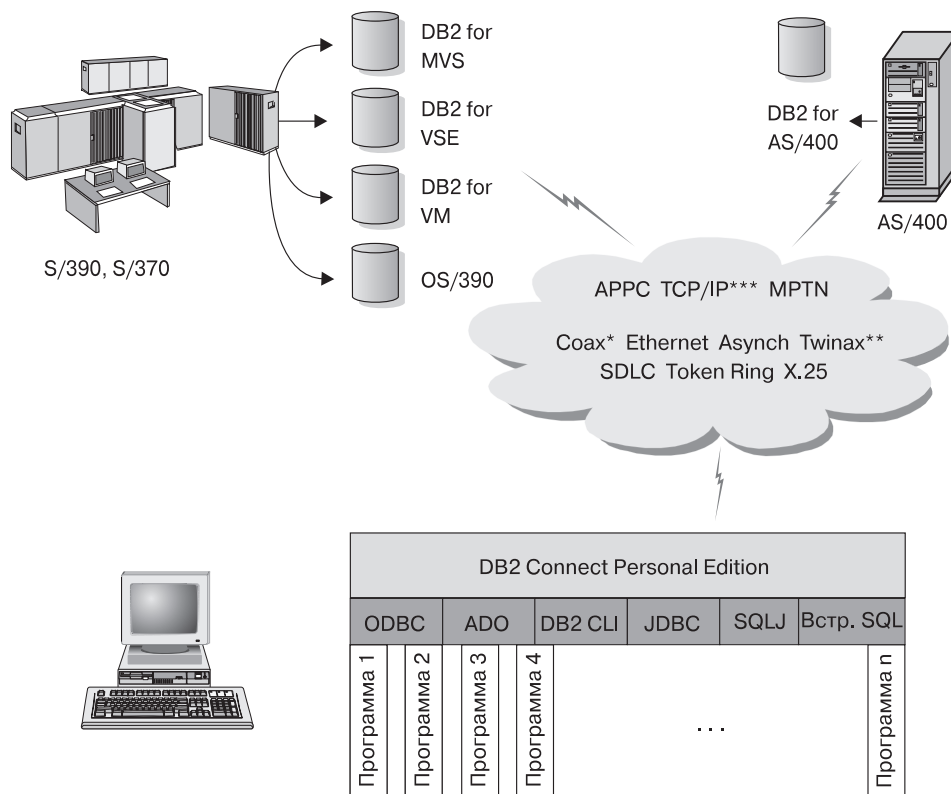
Прямой доступ к базам данных

Базовая возможность DB2 Connect – обеспечение прямого соединения с базой данных хоста из прикладных программ, выполняющихся на рабочих станциях с 32–битными системами Windows, Linux или OS/2. Самый простой способ реализовать такое решение – использовать DB2 Connect Personal Edition.

Каждая рабочая станция, на которой установлена DB2 Connect Personal Edition, может устанавливать прямое соединение TCP/IP с серверами DB2 для OS/390, DB2/400 и DB2 UDB для Windows NT, Windows 2000, UNIX и OS/2. Кроме этого, прикладные программы могут в одной транзакции соединяться с несколькими базами данных и изменять в них данные; при этом протокол двухфазного принятия обеспечивает целостность всех данных.

DB2 Connect Personal Edition также имеет встроенную поддержку APPC для связи с DB2 для MVS и с другими базами данных хоста и AS/400, для работы с которыми требуется APPC. Однако, если доступна собственная поддержка TCP/IP, настоятельно рекомендуется вместо SNA использовать TCP/IP.

На рис. 3 на стр. 23 показаны рабочие станции, непосредственно связанные с сервером баз данных хоста или AS/400. На каждой из рабочих станций установлена DB2 Connect Personal Edition.



Не все протоколы поддерживаются на всех платформах.

* Только для соединений хоста

** Для AS/400

*** Для связи TCP/IP необходима DB2 for OS/390 V5R1, DB2 for AS/400 V4R2 или DB2 for VM V6.1

Рисунок 3. Прямое соединение между DB2 Connect и сервером баз данных хоста или AS/400

Примечания:

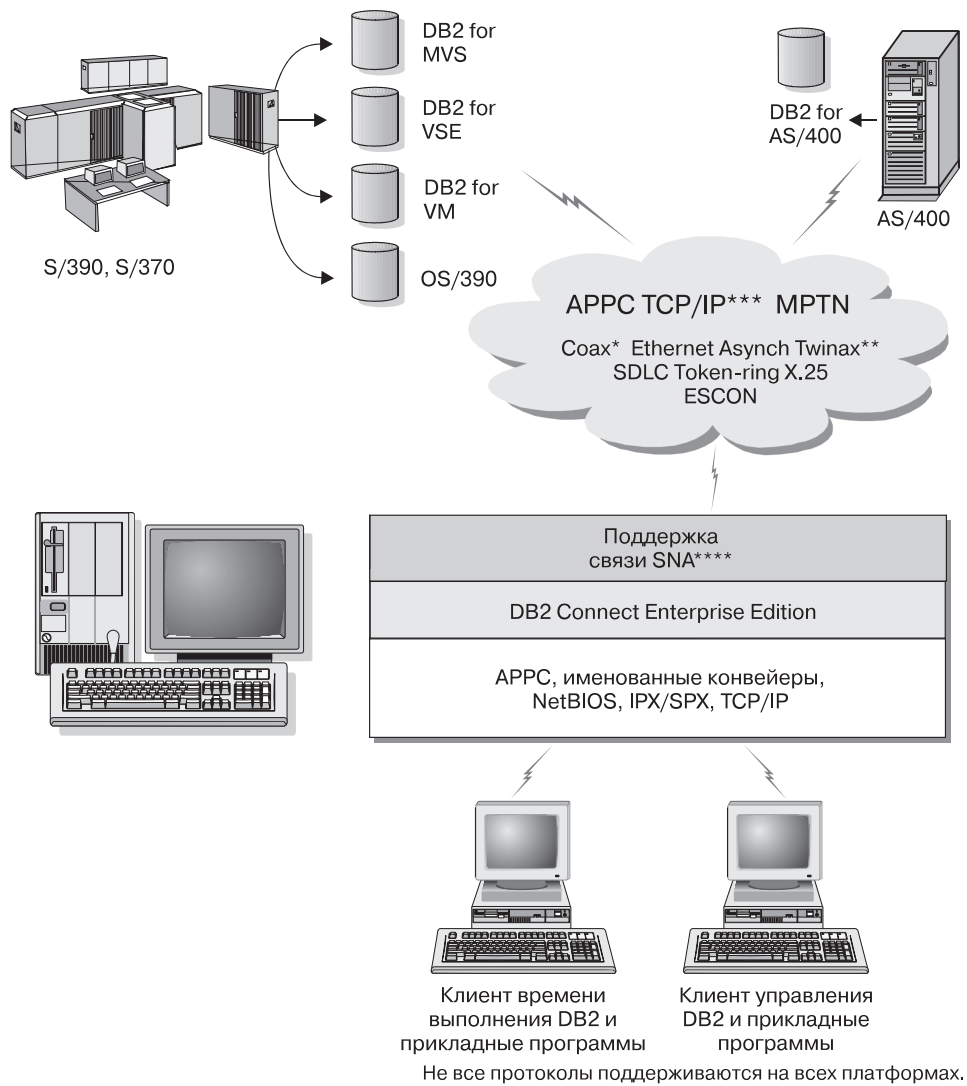
1. Информацию о протоколах, поддерживаемых конкретными реквестерами прикладных программ DRDA, и используемых для них системах DRDA хоста или AS/400 смотрите в соответствующей книге *DB2 Connect Quick Beginnings*.
2. На рабочей станции DB2 Connect установка DB2 Universal Database необязательна. Но, если на рабочей станции DB2 Connect нужна полная система управления реляционными базами данных, закажите DB2 Universal Database.

3. В пакет DB2 Connect теперь входит клиент разработки программ DB2, который можно установить и использовать для разработки прикладных программ. Кроме этого, в DB2 Connect теперь входит построитель хранимых процедур, который можно использовать для построения, тестирования и размещения хранимых процедур для DB2 для OS/390.
4. Программисты, разрабатывающие на языке C программы для Windows, в которых используется Microsoft ODBC, OLE DB или ActiveX Data Objects (ADO), должны использовать *Microsoft Open Database Connectivity Software Development Kit*. Программисты, разрабатывающие программы на языке Java, могут использовать любую среду разработки программ Java, например, VisualAge for Java фирмы IBM.

DB2 Connect Enterprise Edition в качестве сервера связи

Сервер DB2 Connect позволяет нескольким клиентам соединиться с базами данных хоста или AS/400 и может значительно уменьшить затраты на организацию доступа к данным предприятия. На рис. 4 на стр. 25 показано предлагаемое IBM решение для случаев, когда клиент DB2 должен устанавливать не прямое соединение с сервером баз данных хоста или AS/400 через DB2 Connect Enterprise Edition.

В следующем примере можно вместо сервера DB2 Connect использовать сервер DB2 UDB Enterprise Edition или Enterprise – Extended Edition, для которого установлен компонент поддержки сервера DB2 Connect.



- * Только для соединений хоста
- ** Для AS/400
- *** Для соединений TCP/IP необходима DB2 for OS/390 V5R1, DB2 for AS/400 V4R2, или DB2 for VM V6.1
- **** Поддержка связи SNA специфична для каждой операционной системы и требуется, только если связь TCP/IP недоступна.

Рисунок 4. DB2 Connect Enterprise Edition

DB2 Connect и программы Web

Браузер Web быстро становится стандартным интерфейсом для самых различных программ: от электронных каталогов до программ локальной сети. Для простых программ Web может хватить одного сервера Web. Для сложных программ, которым может требоваться доступ к базам данных и обработка транзакций, IBM предлагает для обработки очень большого числа одновременных транзакций через Web использовать DB2 Connect.

В этом разделе описываются коммерческие системы для работы в Web, для которых выгодно использовать DB2 Connect.

Преимущества и ограничения традиционного программирования с использованием CGI

Типичные прикладные программы для электронного бизнеса в WWW используют для обращения пользователей к базам данных интерфейс CGI. Многие компании также используют для внутренних нужд программы Web; обычно эти программы используются и для работы с базами данных.

Пользователи заполняют формы на Web-странице; эти формы передаются через CGI прикладным программам или сценариям на сервере Web. Эти сценарии в свою очередь используют API конкретной базы данных, чтобы передать требования SQL базе данных хоста. Этот же сценарий может затем создавать Web-страницу (HTML) с результатами выполнения требования и посылать ее назад, чтобы браузер Web вывел ее на экран пользователя. Пример – электронный каталог, в котором пользователь может запросить информацию о наличии и текущей цене для конкретных товаров или услуг.

Прикладные программы CGI легко создаются и несложны в обслуживании. Поскольку стандарт CGI независим как от операционной системы, так и от языка программирования, CGI доступен почти на всех компьютерных платформах. Программы CGI могут быть написаны на C++ или на языке сценариев (например, на Perl).

Хотя CGI может показаться идеальным решением для прикладных программ Web, этот интерфейс имеет серьезные недостатки. Среда разработки программ для CGI не настолько развита, как для других API. Кроме того, существуют проблемы масштабирования, существенные для коммерческих систем большого масштаба. При каждом запуске программы CGI на сервере Web создается новый процесс. Каждый экземпляр должен создать свое собственное соединение с базой данных, и каждый экземпляр выдает свое собственное требование к базе данных. В системах с большим числом транзакций это может значительно ухудшать производительность.

Для создания современных прикладных программ электронного бизнеса большого объема можно использовать DB2 Connect с сервером Web. DB2 Connect позволяет улучшить производительность прикладных программ для

Web. Хранимые процедуры (смотрите раздел “DB2 Connect на сервере Web” на стр. 27) позволяют пользователям DB2 Connect уменьшить число требований, посылаемых базе данных.

Группировка соединений в пулы (смотрите раздел “Пул соединений” на стр. 28) сокращает количество операций установления и завершения соединений с базой данных. Для крупных систем, в которых ограничения CGI становятся существенными, можно использовать IBM Net.Data (смотрите раздел “Net.Data” на стр. 29) и WebSphere (смотрите раздел “IBM WebSphere” на стр. 30), обеспечивающие соединения других типов (не CGI) для прикладных программ крупных предприятий.

DB2 Connect на сервере Web

Со всеми продуктами IBM DB2 Connect для OS/2, UNIX, Windows NT и Windows 2000 поставляются серверы Web (HTTP). DB2 Connect Enterprise Edition обеспечивает дополнительную поддержку для серверов Web Apache или Lotus Domino Go и может также работать с любым другим сервером Web, например, с Microsoft Internet Information Server или Netscape Enterprise Server.

Для работы с базами данных семейства DB2, работающими в системах OS/390, AS/400, VM и VSE, на сервере Web должна быть установлена DB2 Connect Enterprise Edition. DB2 Connect Enterprise Edition обеспечивает библиотеки и интерфейсы связи, позволяющие серверам Web обращаться к платформам хоста и AS/400. Для связи между сервером Web и базой данных, работающей в системе OS/390, AS/400, VM или VSE, можно использовать TCP/IP или SNA.

Примечание: Предлагаемые IBM решения для Web позволяют работать с несколькими базами данных в одном сценарии CGI или внутри одной транзакции сценария CGI.

В следующих двух разделах описываются возможности повышения производительности прикладных программ CGI, обращающихся к базам данных DB2. В последующих разделах будут рассмотрены альтернативы стандарту CGI, такие как Java.

Хранимые процедуры

Так же как для систем клиент–сервер, для прикладных программ Web важно минимизировать поток данных между сервером HTTP и базой данных. Это особенно важно при большом числе транзакций, характерном для большинства прикладных программ электронного бизнеса.

Рекомендуется использовать комбинацию прикладной программы CGI и программных модулей, оформленных в виде хранимых процедур. Во всех продуктах DB2 Universal Database для OS/2, UNIX и Windows и DB2 для

OS/390, AS/400 и VSE используется одно и то же соглашение о передаче параметров при вызовах хранимых процедур.

Как и в случае обычной программы CGI, браузер Web посылает форму на сервер Web, на котором выполняется сценарий CGI. Однако базе данных посылаются не отдельные операторы SQL, а запрос на выполнение хранимой процедуры. Эта хранимая процедура содержит множество операторов SQL, которые в противном случае пришлось бы выполнять по отдельности. При использовании хранимой процедуры уменьшается число сообщений, передаваемых между сценарием CGI и базой данных.

Главное преимущество использования хранимых процедур — уменьшение потока данных между сервером HTTP и базой данных DB2. Дополнительную информацию о хранимых процедурах смотрите в руководстве *Application Development Guide* и в электронной справке строителя хранимых процедур DB2.

Пул соединений

На установление соединения между сервером DB2 Connect и хостом тратятся вычислительные ресурсы и время. В системах, в которых тысячи клиентов часто устанавливают и завершают соединения с хостом через сервер DB2 Connect, значительная часть времени процессора тратится на установление и завершение соединений.

Использование пула соединений DB2 Connect позволяет значительно улучшить производительность в таких системах. DB2 Connect поддерживает в открытом состоянии ряд соединений с базой данных. Когда клиент запрашивает соединение, ему может быть предоставлено уже открытое соединение из этого пула. Такая схема позволяет существенно сократить затраты ресурсов на установление и завершение соединений.

Дополнительную информацию о том, как работает пул соединений, смотрите в разделе “Поддержка пула соединений” на стр. 140.

DB2 Connect в качестве сервера прикладных программ Java

Многих недостатков CGI можно избежать, если использовать вместо CGI язык Java. Фирма IBM поставляет апплеты и прикладные программы Java, позволяющие использовать Java вместо CGI на каждом шаге транзакции Web. Предлагаемые IBM решения позволяют использовать различные техники — для работы с DB2 можно использовать сценарии (например, для Net.Data или Microsoft Active Server Pages) или более совершенные реализации с использованием сервера прикладных программ Java (например, IBM WebSphere).

Для программ Java используются два интерфейса прикладного программирования (API). Первый API, JDBC, применяется при разработке

апплетов Java для работы с данными, прикладных программ Java, а также сервлетов Java, Java Server Pages (JSP) и Enterprise Java Beans (EJB). В API JDBC используются вызовы методов. Другой API Java – SQLJ. SQLJ позволяет задавать операторы SQL прямо в программе Java. DB2 может использовать для транзакции Web оба эти API, как на стороне клиента, так и на стороне сервера.

На стороне клиента поддерживаются апплеты, апплеты для работы с данными и прикладные программы. На стороне базы данных для Java поддерживаются объекты баз данных, такие как пользовательские функции и хранимые процедуры.

Для DB2 для OS/390, DB2 для VSE и VM и DB2 для OS/400 можно использовать прикладные программы Java двумя способами. Можно использовать прямое соединение, обеспечиваемое DB2 Connect Personal Edition с TCP/IP или SNA, или же соединение через сервер DB2 Connect Enterprise Edition, обеспечивающий связь с мэйнфреймом или системой AS/400.

В обоих случаях пользователю Web для доступа к базе данных не требуется никаких специальных программ помимо стандартного браузера Web. Единственное, что нужно установить – это сервер DB2 Connect и любой стандартный сервер Web. Если сервер Web и DB2 Connect расположены на разных компьютерах, клиент DB2 должен быть установлен на компьютере сервера Web.

Главный компонент для DB2 для OS/390 – это DB2 Connect Enterprise Edition, работающий на сервере среднего уровня. Именно этот компонент обеспечивает возможности использования сервера JDBC, а также связь с сервером DB2 для OS/390, VSE и VM или AS/400. Еще раз подчеркнем, что для браузера Web клиента не требуется никакого специально программного обеспечения.

IBM предоставляет большой набор инструментов для разработки прикладных программ и апплетов Java. DB2 Developer's Edition содержит полный набор инструментов для разработки программ для работы с базами данных: VisualAge for Java Professional Edition, WebSphere Application Server, Net.Data, а также DB2 Universal Database и DB2 Connect для тестирования программ. IBM VisualAge for Java Enterprise Edition также содержит инструменты разработки для прикладных программ крупных предприятий. Можно также использовать для работы с базами данных IBM продукты других производителей (например, Borland JBuilder или Symantec Visual Cafe).

Net.Data

Компонент Net.Data, входящий в семейство DB2 Universal Database и DB2 Connect, представляет собой набор инструментов разработки прикладных программ, помогающий создавать и поддерживать прикладные программы для транзакций через Web. Net.Data можно использовать для доступа к данным и для изменения данных, хранимых в DB2 UDB для OS/2, Windows

NT, Windows 2000, UNIX, OS/390, VM, VSE и OS/400. Созданные при помощи Net.Data прикладные программы хранятся на сервере Web и могут активироваться браузером Web.

В Net.Data используются шаблоны, позволяющие пользователям, знакомым с основами языков HTML и SQL, строить очень сложные программы Web. Такой шаблон представляет собой текстовый файл, который может содержать программы на языке Java, сценарии Java, теги HTML и встроенные функции. Эти шаблоны можно использовать для динамической генерации Web-страниц, использующих заранее определенные компоновку страницы, переменные и функции.

Базовый шаблон Net.Data содержит семь специальных разделов:

- Общие разделы, используемые в основном для документации, используемой программистом
- Раздел определений, где задаются объявления переменных
- Раздел функций, где находится основной код программы
- Раздел отчета, где задаются команды форматирования выходных данных шаблона Net.Data
- Раздел HTML, где находится большинство команд HTML, используемых в Web-странице
- Раздел включаемых файлов, позволяющий использовать в шаблоне общие команды, вынесенные в отдельный шаблон
- Раздел сообщений, где определяется обработка ошибок

Главное достоинство Net.Data, особенно при работе с DB2, состоит в том, что на клиенте не нужно размещать дополнительных программ. На клиенте используется только браузер Web.

На рабочих станциях с системой Windows NT, Windows 2000, OS/2 или UNIX процессор Net.Data устанавливается вместе с DB2 Universal Database, там же устанавливается сервер Web. Для соединений с DB2 для OS/390, DB2 для VSE и VM и DB2 для OS/400 все компоненты Net.Data размещаются на сервере DB2 Connect, вместе с сервером Web.

IBM WebSphere

IBM WebSphere позволяет реализовать более полные решения для электронного бизнеса по сравнению с использованием традиционных программ CGI. Серверы прикладных программ WebSphere не только поддерживают сценарии CGI, но позволяют также реализовать сложные системы для работы через Web, в которых используются сервлеты, Active Server Pages и Enterprise JavaBeans. WebSphere позволяет:

- Использовать индустриальные стандарты, ускоряющие разработку и облегчающие обслуживание программ;

- Использовать инструментальные технологии и системы прикладных программ других производителей;
- Анализировать производительность работы и использование содержимого системы Web;
- Легко увеличивать масштаб системы при необходимости работать с большим числом пользователей и поддерживать большой поток данных;
- Размещать программы в целом ряде основных операционных систем (IBM AIX, HP-UX, Linux, Novell NetWare, IBM OS/2, IBM OS/390, IBM OS/400, Sun Solaris, Microsoft Windows NT и Windows 2000);
- Использовать существующий сервер Web (например, серверы Web производства Apache, IBM, Netscape и Microsoft).

WebSphere – это не один продукт, а семейство из трех продуктов, предназначенных для разных сегментов рынка. Центральная часть системы WebSphere – сервер прикладных программ WebSphere.

Сервер прикладных программ WebSphere обеспечивает среду для трех типов объектов. Первый тип – это Java Server Pages, аналогичные Active Server Pages. Второй тип объектов – это сервлеты Java; третий – Enterprise JavaBeans. Enterprise JavaBeans становятся стандартом для современных прикладных программ очень большого масштаба.

Кроме этого, Data Access JavaBeans реализуют очень сложные функции баз данных, созданные специально для DB2. К DB2 можно также обращаться напрямую, используя JDBC и SQLJ. Также поддерживаются COM+ и CORBA.

Прикладные программы WebSphere могут размещаться на той же платформе, что и сервер Web и DB2 Universal Database. При использовании DB2 для OS/390, VM, VSE и AS/400 WebSphere размещается на той же платформе, что и DB2 Connect Enterprise Edition.

Есть несколько вариантов WebSphere, а также дополнительные продукты Web Studio и WebSphere Performance Packs. Три продукта WebSphere:

Standard Edition

Этот сервер позволяет разработчикам систем Web использовать сервлеты Java и технологию JSP, чтобы в системах и порталах Web быстро и легко перейти от статических страниц к использованию исходных данных для динамического создания настраиваемых Web-страниц. Он также включает поддержку передового промышленного стандарта XML, предназначенного для совместного использования информации и данных в группах компьютеров или предприятий, и встроенную технологию анализа системы, позволяющую получить информацию о производительности и использовании системы, что помогает максимально повысить отдачу от капиталовложений вашей компании в систему Web.

Advanced Edition

Этот высокопроизводительный сервер EJB позволяет разработчикам прикладных программ использовать компоненты EJB. Он обеспечивает масштабируемую защищенную связь и поддержку Java и включает все возможности версии Standard Edition.

Enterprise Edition

Этот сервер объединяет отдельные системы предприятия в единое целое, позволяя создать мощные прикладные программы электронного бизнеса и добиться максимально эффективного использования ресурсов. Enterprise Edition включает возможности передовых технологий IBM TXSeries– и Component Broker. Он также включает все возможности версий Advanced Edition и Standard Edition.

Использование DB2 Connect с серверами прикладных программ

С появлением прикладных программ типа клиент–сервер разработчики программ смогли создавать программы, более удобные в использовании и требующие меньших затрат на обучение персонала, в которых используется графический интерфейс пользователя и которые работают на таких платформах, как Windows и OS/2. В то же время такая структура программ позволяет передать функции управления базами данных надежным серверам баз данных, работающим в различных операционных системах и на разных аппаратных платформах.

Модель клиент–сервер, в которой прикладные алгоритмы размещаются на рабочей станции клиента, обычно называют *двухуровневой моделью клиент–сервер*. В двухуровневой модели прикладная программа располагается на уровне клиента, а сервер баз данных реализует уровень сервера или уровень базы данных. Как показано в разделе “Прямой доступ к базам данных” на стр. 22, DB2 Connect обеспечивает полную поддержку двухуровневых прикладных программ клиент–сервер при использовании в качестве серверов баз данных DB2 для OS/390, DB2 для MVS/ESA, DB2/400 или DB2 для VM и VSE.

С увеличением размера прикладных программ клиент–сервер становится очевидно, что двухуровневая модель клиент–сервер имеет значительные ограничения. Размещение большого числа прикладных программ на сотнях или даже тысячах рабочих станций клиентов делает сложным и дорогим обслуживание программ. Любые изменения алгоритмов требуют изменения клиентских программ. Часто такие изменения прикладных программ должны выполняться на всех клиентских рабочих станциях организации одновременно, чтобы обеспечивалось согласованное изменение алгоритмов работы.

Другой недостаток двухуровневой модели клиент–сервер становится очевидным при росте объема ресурсов, используемых прикладными программами. Размещение сотен или тысяч *толстых клиентов* (как часто

называют клиенты двухуровневой модели клиент–сервер) требует большей вычислительной мощности и емкости на каждой из рабочих станций клиентов. Более того, требования к серверу баз данных также сильно возрастают, так как каждый клиент использует отдельное соединение и потребляет ресурсы на поддержание такого соединения. Неудобства двухуровневой модели клиент–сервер, связанные с размещением прикладных программ на компьютерах клиентов, можно в некоторой степени ослабить, широко используя хранимые процедуры, но избавиться от других недостатков можно, только перейдя к другой модели.

Сервер прикладных программ

По мере роста цены и сложности двухуровневых прикладных программ клиент–сервер большинство крупных прикладных программ переводятся на многоуровневую модель клиент–сервер. В такой модели роль базы данных не меняется. Однако к уровню клиента добавляется один или несколько промежуточных уровней (обычно один, поэтому такая модель называется *трехуровневой*).

В трехуровневой модели клиент используется только для взаимодействия с пользователем и не содержит коммерческих алгоритмов. Промежуточный слой состоит из одного или нескольких серверов прикладных программ. Назначение сервера прикладных программ – обеспечивать надежную и недорогую реализацию алгоритмов, необходимых для работы предприятия. Как и в случае двухуровневой модели, для реализации алгоритмов часто используются хранимые процедуры, позволяющие улучшить производительность.

Поскольку рабочая станция клиента более не содержит прикладных алгоритмов и используется только для поддержания взаимодействия с пользователем, требования к ресурсам для уровня клиента значительно снижаются. Поэтому уровень клиента в трехуровневой модели часто называют *"тонким" клиентом*. Кроме этого, так как централизованный сервер прикладных программ обслуживает требования от всех клиентов, он может использовать для всех клиентов общие ресурсы, например, соединения с базой данных. В итоге сервер баз данных более не должен поддерживать отдельные соединения для каждого пользователя.

В настоящее время существует множество примеров промышленной реализации трехуровневых серверов прикладных программ. Почти все производители систем Enterprise Resource Planning (ERP) используют для своих прикладных программ трехуровневую модель (например, в прикладных программах SAP R/3 и PeopleSoft V7). Другой пример – ведущие производители систем Enterprise Relationship Management, например, Siebel и Vantive.

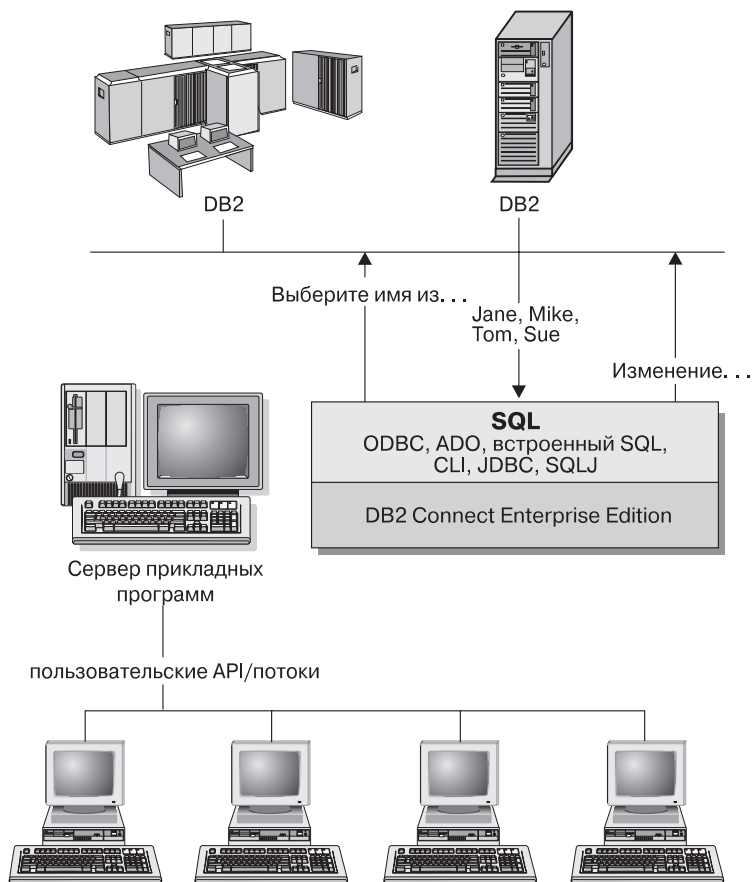
Серверы прикладных программ и DB2 Connect

Серверы DB2 Connect Enterprise Edition обеспечивают полную поддержку многоуровневых прикладных программ. DB2 Connect обеспечивает поддержку множества API, которые могут использоваться при разработке прикладных программ (ODBC, ADO, DB2 CLI, Embedded SQL, JDBC и SQLJ), а также содержит полную инфраструктуру связи для взаимодействия с серверами баз данных семейства DB2.

DB2 Connect поддерживает также системы, в которых уровень баз данных состоит из нескольких серверов баз данных семейства DB2. Это позволяет серверам прикладных программ выполнять в одной транзакции обновление данных, расположенных на нескольких серверах баз данных.

Целостность данных при таких распределенных транзакциях обеспечивается поддерживаемым DB2 Connect протоколом двухфазного принятия. Например, прикладная программа может в одной транзакции изменять данные в базе данных DB2 для OS/390 и в базе данных DB2 UDB в Windows NT. Если установлена и включена поддержка распределенных требований, прикладная программа может в одной и той же транзакции читать данные из базы данных Oracle и изменять данные в базе данных семейства DB2.

На следующей диаграмме показаны API и механизмы связи между сервером прикладных программ и серверами баз данных, обеспечиваемые DB2 Connect Enterprise Edition.



Дополнительные возможности DB2 Connect, такие как динамические соединения (смотрите раздел “Поддержка пула соединений” на стр. 140) и концентратор соединений (смотрите раздел “Концентратор соединений DB2 Connect” на стр. 141), существенно уменьшают использование ресурсов прикладными программами и упрощают реализацию сервера прикладных программ.

DB2 Connect и конфигурации сервера прикладных программ

Для работы с серверами прикладных программ требуется продукт DB2 Connect Enterprise Edition (который можно приобрести отдельно или в пакете продукта DB2 Connect Unlimited Edition). DB2 Connect Personal Edition не поддерживается и не лицензируется для работы с серверами прикладных программ. Кроме того, покупатели, собирающиеся использовать серверы прикладных программ, должны ознакомиться с условиями лицензии на приобретаемую копию DB2 Connect, чтобы понять, сколько пользовательских лицензий нужно приобрести.

DB2 Connect можно разместить в среде сервера прикладных программ двумя способами. DB2 Connect Enterprise Edition устанавливается:

- на компьютере сервера прикладных программ или
- на отдельном компьютере сервера связи.

В большинстве случаев копию DB2 Connect лучше устанавливать на том же компьютере, что и сервер прикладных программ. Это позволяет DB2 Connect участвовать во всех схемах защиты от сбоев и распределения нагрузки, которые могут использоваться сервером прикладных программ. Такое размещение программ может улучшить производительность, поскольку в этом случае не используется дополнительная линия связи, которая требуется, если DB2 Connect установлен на отдельном компьютере. При этом упрощается также управление системой, так как не нужно устанавливать и обслуживать дополнительный сервер.

Если недоступна версия DB2 Connect Enterprise Edition для операционной системы или аппаратной платформы, где работает сервер прикладных программ, DB2 Connect можно установить на отдельном сервере. Например, если сервер прикладных программ размещен на сервере Silicone Graphics (SGI) или SCO UnixWare, DB2 Connect можно разместить только на отдельном сервере, так как версий DB2 Connect Enterprise Edition для этих платформ нет.

Использование DB2 Connect с мониторами транзакций

В предыдущем разделе было рассмотрено использование DB2 Connect с сервером прикладных программ. Сервер прикладных программ позволяет большому числу пользователей выполнять программы, используя минимум системных ресурсов.

Возможности сервера прикладных программ можно расширить, разрешив прикладным программам на сервере прикладных программ выполнять скоординированные транзакции. Для координации транзакций используется монитор транзакций. Монитор транзакций работает вместе с сервером прикладных программ.

Транзакцией можно считать обычную операцию в текущем ежедневном функционировании организации (обычно это запрос на обслуживание). Мониторы транзакций предназначены для обеспечения правильной обработки транзакций.

В каждой организации имеются свои правила и процедуры, описывающие требуемый алгоритм работы. Пользовательские прикладные программы, реализующие эти правила, можно назвать *коммерческими алгоритмами*. Выполнение транзакций такими коммерческими программами часто называют обработкой транзакций или диалоговой обработкой транзакций (OLTP).

Основные характеристики коммерческих систем OLTP:

Большое число пользователей

Обычно обработка транзакций используется при работе большей части работников организации, поскольку деятельность каждого из них влияет на текущее состояние бизнеса.

Повтор операций

В большинстве случаев при работе с компьютером вновь и вновь выполняются одни и те же операции. Например, ежедневно многократно вводятся заказы или обрабатываются платежи.

Короткие взаимодействия

Для большинства операций используются лишь кратковременные взаимодействия с системой обработки транзакций.

Совместно используемые данные

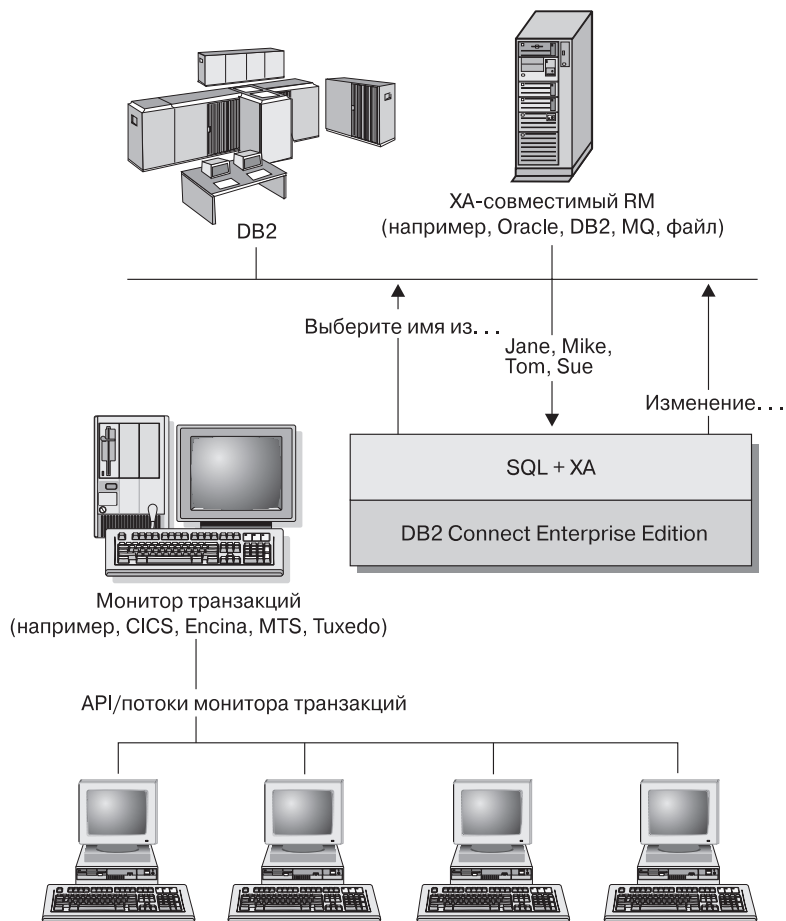
Поскольку данные отражают состояние дел в организации, может использоваться только одна копия данных.

Целостность данных

Данные должны отражать текущее состояние дел в организации и должны быть внутренне непротиворечивы. Например, каждый заказ должен быть связан с информацией о покупателе.

Низкие затраты на одну транзакцию

Поскольку обработка транзакций входит в непосредственные затраты на деятельность организации, стоимость системы должна быть минимальной. DB2 Connect позволяет прикладным программам на сервере прикладных программ в системах UNIX, Windows NT, Windows 2000 или OS/2 выполнять транзакции с удаленными серверами баз данных в локальной сети, на хосте или AS/400 и позволяет координировать транзакции при помощи монитора транзакций.



На этом рисунке показаны API и механизм связи между сервером прикладных программ и серверами баз данных, обеспечиваемые DB2 Connect Enterprise Edition.

Примеры мониторов транзакций

Наиболее распространенные в настоящее время мониторы транзакций:

- IBM TxSeries CICS
- IBM TxSeries Encina Monitor
- BEA Tuxedo.

В транзакциях, координируемых этими мониторами, могут использоваться серверы баз данных Microsoft Transaction Server Remote S/390, AS/400 и серверы баз данных локальной сети.

Tuxedo и DB2 Connect

При работе с DB2 Connect Версии 6 и более ранними версиями прикладные программы, использующие Tuxedo, могут обращаться к серверам баз данных хоста и AS/400 только в режиме чтения. Это ограничение устранено в DB2 Connect Версии 7. Теперь прикладные программы могут изменять данные на серверах баз данных хоста и AS/400, используя Tuxedo для координации транзакций. При этом есть некоторые особые требования к конфигурации и ограничения. Дополнительную информацию смотрите в разделе “Концентратор соединений DB2 Connect” на стр. 141.

Модель распределенной обработки транзакций (DTP) X/Open

Реализующая коммерческий алгоритм прикладная программа может внутри одной транзакции требовать изменения данных в нескольких базах данных. Например, банковская прикладная программа, выполняющая перевод денег с одного счета на другой, требует записать информацию об уменьшении счета в одной базе данных (для счета, с которого снимаются деньги) и информацию об увеличении счета в другой базе данных (для счета, на который эти деньги переводятся).

При этом две базы данных могут быть базами данных разных производителей. Например, одна база данных – это DB2 Universal Database для OS/390, а вторая – база данных Oracle. Чтобы не реализовывать в каждом мониторе транзакций специальные интерфейсы транзакций для всех типов баз данных, был разработан общий интерфейс для транзакций между монитором транзакций и любыми ресурсами, к которым могут обращаться прикладные программы. Этот интерфейс называется *интерфейсом XA*. Монитор транзакций, использующий интерфейс XA, называется *менеджером транзакций (TM) стандарта XA*. Обновляемый ресурс, поддерживающий интерфейс XA, называется *менеджером ресурсов (RM) стандарта XA*.

Все перечисленные выше мониторы транзакций представляют собой менеджеры транзакций стандарта XA. Серверы баз данных удаленного хоста, AS/400 и DB2 UDB в локальной сети при использовании для доступа к ним DB2 Connect представляют собой менеджеры ресурсов стандарта XA. Поэтому любой монитор транзакций, имеющий TM стандарта XA, может при выполнении транзакций коммерческих прикладных программ работать с серверами баз данных хоста, AS/400 и DB2 UDB в локальной сети.

Как использовать DB2 Connect с менеджером транзакций стандарта XA

В этом разделе описываются необходимые шаги конфигурирования для использования серверов баз данных S/390 и AS/400 с монитором транзакций. В этом разделе подразумевается, что имеется работающий монитор транзакций и установленная DB2, а также сконфигурировано и протестировано

соединение с сервером баз данных хоста или AS/400. Более подробную информацию смотрите в книге *DB2 Connect Quick Beginnings*.

Шаги конфигурирования наиболее часто используемых мониторов транзакций, описаны в руководстве *Administration Guide*. Нет никаких различий между конфигурированием доступа к серверу баз данных DB2 UDB в локальной сети и конфигурированием доступа к серверу баз данных хоста или AS/400. Далее описаны основные шаги конфигурирования для мониторов транзакций, не перечисленных в руководстве *Administration Guide*.

Чтобы сконфигурировать DB2 Connect для использования серверов баз данных S/390 или AS/400 в мониторе транзакций, выполните следующие шаги:

1. Сконфигурируйте монитор транзакций для доступа к коммутатору XA DB2. Коммутатор XA DB2 сообщает монитору транзакций адреса API XA DB2 Connect. Для каждого типа мониторов транзакций это делается разным способом. Информацию о том, как задать коммутатор XA DB2 для монитора транзакций, смотрите в руководстве *Administration Guide*.
2. Сконфигурируйте монитор транзакций для работы со строкой XA_OPEN DB2. Для каждого типа мониторов транзакций это делается своим способом. Информацию о строке XA OPEN для DB2 Connect смотрите в руководстве *Administration Guide*. Информацию о том, как сконфигурировать использование строки XA OPEN DB2 монитором транзакций, смотрите в документации по конкретному монитору транзакций.
3. При необходимости измените значения по умолчанию параметров конфигурации менеджера точек синхронизации (SPM) DB2 Connect. Серверы баз данных хоста и AS/400 пока не поддерживают интерфейс XA.

SPM – это компонент DB2 Connect, преобразующий протокол двухфазного принятия стандарта XA в протокол двухфазного принятия, используемый серверами баз данных хоста и AS/400. По умолчанию экземпляр DB2 имеет предопределенные значения для параметров конфигурации SPM. Самый важный параметр – это параметр конфигурации менеджера баз данных SPM_NAME. По умолчанию значение этого параметра генерируется на основе первых семи символов TCP/IP-имени хоста.

Если для соединения с DB2 для OS/390 используется TCP/IP, для параметров не нужно изменять значения по умолчанию. В этом случае изменение конфигурации SPM не требуется, так как SPM уже будет работать. Если для доступа к серверам баз данных хоста или AS/400 используется SNA, нужно убедиться, что значение параметра SPM_NAME представляет собой правильное имя SNA LU в данной сети. Если значение по умолчанию параметра SPM_NAME не подходит, нужно при помощи мастера многоузлового обновления изменить его.

Глава 4. Программирование в среде DB2 Connect

Этот раздел содержит сведения о создании прикладных программ, использующих DB2 Connect. Дополнительную информацию смотрите в книгах *CLI Guide and Reference*, *Command Reference* и *Application Development Guide*.

Программирование в распределенной среде

DB2 Connect позволяет прикладной программе обращаться к данным в базах данных DB2 на серверах System/390 и AS/400. Например, прикладная программа, работающая под Windows, может обращаться к данным в базе данных DB2 Universal Database for OS/390. Можно создавать новые или изменять существующие прикладные программы для запуска в среде хоста или AS/400. Можно также разрабатывать программы в одной среде и переносить их в другую.

DB2 Connect дает возможность использовать с программными продуктами для баз данных хоста, такими как DB2 Universal Database for OS/390, следующие API, при условии, что их поддерживает программный продукт баз данных хоста:

- Встроенный SQL, статический и динамический
- Интерфейс уровня вызовов DB2
- Microsoft ODBC API
- JDBC

Некоторые операторы SQL различаются в разных программных продуктах для реляционных баз данных. Операторы SQL делятся на следующие категории:

- Общие для всех используемых программных продуктов для баз данных, независимо от стандартов
- Перечисленные в документе *SQL Reference* и поэтому присутствующие во всех программных продуктах фирмы IBM для реляционных баз данных
- Уникальные для одной из систем баз данных, к которой вы обращаетесь

Операторы SQL первых двух категорий хорошо переносятся, а для операторов третьей категории в первую очередь потребуются изменения. В общем случае, операторы SQL языка описания данных (Data Definition Language – DDL) переносятся хуже, чем операторы языка манипулирования данными (Data Manipulation Language – DML).

DB2 Connect воспринимает несколько операторов SQL, не поддерживаемых DB2 Universal Database. DB2 Connect передает такие операторы на сервер хоста

или AS/400. Информацию об ограничениях для разных платформ, таких как максимальная длина столбца, смотрите в справочнике *SQL Reference*.

Если вы переносите прикладную программу CICS из OS/390 или VSE для работы под другим программным продуктом CICS (например, CICS for AIX), она также может обращаться к базе данных OS/390 или VSE, используя DB2 Connect. Подробности смотрите в руководствах *CICS/6000 Application Programming Guide* и *CICS Customization and Operation*.

При программировании в среде хоста или AS/400 рассмотрите следующие вопросы:

- Использование языка определения данных (DDL)
- Использование языка манипулирования данными (DML)
- Использование языка управления данными (Data Control Language – DCL)
- Соединение и разъединение
- Прекомпиляцию
- Определение порядка сортировки
- Управление реляционной целостностью
- Блокировка
- Различия кодов SQLCODE и SQLSTATE
- Использование системных каталогов
- Уровни изоляции
- Хранимые процедуры
- Составные операторы SQL NOT ATOMIC
- Распределенные единицы работы
- Операторы SQL, поддерживаемые и неподдерживаемые DB2 Connect.

Использование языка определения данных (DDL)

Операторы DDL различаются в разных программных продуктах для баз данных фирмы IBM, поскольку хранение в разных системах устроено по-разному. В системах серверов хоста или AS/400 между проектированием базы данных и выполнением оператора CREATE TABLE может быть несколько стадий. Например, проект логических объектов в их физическое представление в памяти могут транслировать несколько операторов.

При прекомпиляции для базы данных сервера хоста или AS/400 прекомпилятор передает много таких операторов DDL серверу хоста или AS/400. Эти операторы не будут прекомпилироваться для базы данных в системе, где запущена эта прикладная программа. Например, в OS/2 оператор прикладной программы CREATE STORGROUP будет успешно прекомпилироваться для базы данных DB2 Universal Database for OS/390, но не для базы данных DB2 для OS/2.

Использование языка манипулирования данными (DML)

В общем случае операторы DML обладают высокой переносимостью. Операторы SELECT, INSERT, UPDATE и DELETE схожи для всех программных продуктов для реляционных баз данных фирмы IBM. Большинство прикладных программ в первую очередь используют операторы SQL DML, поддерживаемые программой DB2 Connect.

Числовые типы данных

При передаче DB2 Universal Database числовых данных их тип может меняться. Числовые и зонные десятичные SQLTYPE (поддерживаемые DB2 Universal Database for AS/400) преобразуются в фиксированные (упакованные) десятичные SQLTYPE.

Смешанные данные

Смешанные данные в одном и том же столбце могут содержать символы расширенного набора символов UNIX (EUC), набора двухбайтных символов (DBCS) и набора однобайтных символов (SBCS). В системах, где данные хранятся в коде EBCDIC (OS/390, OS/400, VSE и VM), символы переключения на дополнительный и на стандартный регистры отмечают начало и конец двухбайтных данных. В системах, где данные хранятся в коде ASCII (таких как OS/2 и UNIX), символы переключения на дополнительный и стандартный регистры не требуются.

Если прикладная программа передает смешанные данные из системы ASCII в систему EBCDIC, надо выделить достаточно места для символов переключения регистров. Для каждого переключения с данных SBCS на DBCS добавьте к длине данных по 2 байта. Для лучшей переносимости в прикладных программах, использующих смешанные данные, используйте строки переменной длины.

Длинные поля

Длинные поля (строки длиннее 254 символов) по-разному обрабатываются в разных системах. Сервер хоста или AS/400 для длинных полей может поддерживать только некоторые скалярные функции; например, DB2 Universal Database for OS/390 поддерживает для длинных полей только функции **LENGTH** и **SUBSTR**. Кроме того, серверу хоста или AS/400 для отдельных операторов SQL может потребоваться разная обработка; например, DB2 for VSE & VM требует, чтобы с оператором INSERT использовалась только переменная хоста, значение SQLDA или NULL.

Тип данных большой объект (LOB)

DB2 Connect поддерживает тип данных LOB.

Пользовательские типы (UDT)

DB2 Connect поддерживает только пользовательские определенные типы. Абстрактные типы данных не поддерживаются.

Тип данных ROWID

DB2 Connect обрабатывает тип данных ROWID как VARCHAR для битовых данных.

Тип данных 64-битное целое (BIGINT)

Восьмибайтные (64-битные) целые поддерживаются DB2 Connect. Внутренний тип данных BIGINT используется для обеспечения поддержки мощности очень больших баз данных при сохранении точности данных.

Использование языка управления данными (DCL)

В каждой системе управления реляционными базами данных фирмы IBM уровень детальности для операторов SQL GRANT и REVOKE свой. Уточнить, как различные операторы SQL используются в конкретных системах управления базами данных, можно по соответствующим публикациям.

Соединение и разъединение

DB2 Connect поддерживает для оператора CONNECT как версию CONNECT TO, так и версию CONNECT RESET, а также CONNECT без параметров. Если прикладная программа вызывает оператор SQL без предварительного выполнения явного оператора CONNECT TO, выполняется *неявное* соединение с используемым по умолчанию сервером прикладных программ (если он определен).

При связи с базой данных информация, идентифицирующая систему управления реляционными базами данных, возвращается в поле SQLERRP SQLCA. Если сервер прикладных программ представляет собой реляционную базу данных IBM, в первых трех байтах SQLERRP содержится:

DSN DB2 Universal Database for OS/390

ARI DB2 for VSE & VM

QSQ DB2 Universal Database for AS/400

SQL DB2 Universal Database

Если при использовании DB2 Connect выполнить оператор CONNECT TO или CONNECT без параметров, для кода страны или элемента территории в поле SQLERRMC SQLCA возвращаются пробелы; CCSID сервера прикладных программ возвращается в кодовой странице или элементе кодового набора.

Разорвать соединение можно явным образом, используя оператор CONNECT RESET (для соединения типа 1), операторы RELEASE и COMMIT (для соединения типа 2) или оператор DISCONNECT (для любого типа соединения, но не в среде монитора транзакций).

Если соединение не разорвано явным образом и прикладная программа завершилась нормально, DB2 Connect принимает результаты неявным образом.

Примечание: Прикладная программа может получить SQLCODE, указывающий на ошибки, но тем не менее завершиться нормально; DB2 Connect в этом случае выполнит принятие. Если вы не хотите этого, необходимо выполнить команду ROLLBACK.

Команда FORCE позволяет отключить от базы данных определенных или всех пользователей. Она поддерживается для баз данных серверов хоста или AS/400; пользователь может быть принудительно отключен от рабочей станции DB2 Connect.

Прекомпиляция

Существуют определенные различия в прекомпиляторах для различных систем реляционных баз данных фирмы IBM. Прекомпилятор для DB2 Universal Database отличается от прекомпиляторов сервера хоста или AS/400 следующими особенностями:

- Он делает только один проход по прикладной программе.
- При связывании в отношении баз данных DB2 Universal Database для успешного связывания должны существовать объекты. VALIDATE RUN не поддерживается.

Блокирование

Программа DB2 Connect поддерживает опции связывания блокирования менеджера баз данных DB2:

UNAMBIG

Блокируются только однозначные указатели (по умолчанию).

ALL Блокируются неоднозначные указатели.

NO Указатели не блокируются.

Программа DB2 Connect использует для поля RQRIOBLK размер блока, определенный в файле конфигурации менеджера баз данных DB2. Текущие версии DB2 Connect поддерживают блоки размером до 32767. Если в файле конфигурации менеджера баз данных DB2 указаны большие значения, DB2 Connect использует значение 32767, но не переустанавливает файл конфигурации менеджера баз данных DB2. Блокирование для динамического и

для статического SQL обрабатываются одинаково, с использованием одного и того же размера блока.

Примечание: Большая часть систем серверов хоста или AS/400 рассматривает динамические указатели как неоднозначные, однако системы DB2 Universal Database рассматривают некоторые динамические указатели как однозначные. Во избежание недоразумений с DB2 Connect можно задать BLOCKING ALL.

Задавайте размер блока в файле конфигурации менеджера баз данных DB2, используя CLP, Центр управления или API, как сказано в книгах *Administrative API Reference* и *Command Reference*.

Атрибуты пакетов

У пакета есть следующие атрибуты:

ID собрания

ID пакета. Должен быть задан командой PREP.

Владелец

ID авторизации владельца пакета. Может быть задан командой PREP или BIND.

Создатель

Имя пользователя, связывающее пакет.

Спецификатор

Неявный спецификатор для объектов в пакете. Может быть задан командой PREP или BIND.

В каждой системе сервера хоста или AS/400 есть ограничения на использование этих атрибутов:

DB2 Universal Database for OS/390

Все четыре атрибута могут быть разными. Для использования другого спецификатора требуются специальные привилегии администратора. Дополнительную информацию об условиях использования этих атрибутов смотрите в книге *Command Reference* для DB2 Universal Database for OS/390.

DB2 for VSE & VM

Все атрибуты должны быть идентичными. Если USER1 создает файл связывания (с PREP), а реальное связывание выполняет USER2, USER2 требуется привилегия DBA для связывания для USER1. Для атрибутов используется только имя пользователя USER1.

DB2 Universal Database for AS/400

Спецификатор указывает имя собрания. Взаимоотношение между спецификаторами и владением влияет на предоставление и аннулирование привилегий в отношении объекта. Зарегистрированное

имя пользователя является создателем и владельцем, если только оно не специфицировано ID собрания; в этом случае владельцем является ID собрания. ID собрания должен существовать до того, как будет использован в качестве спецификатора.

DB2 Universal Database

Все четыре атрибута могут быть разными. Для использования другого имени владельца необходимы привилегии администратора, а у связывающего должна быть привилегия CREATEIN для схемы (если она уже существует).

Примечание: DB2 Connect поддерживает команды *SET CURRENT PACKAGESET* для DB2 Universal Database for OS/390 и DB2 Universal Database.

Строки языка C с нулевым символом–ограничителем

Опция связывания CNULREQD переопределяет обработку строк с нулевым символом–ограничителем, указанную с использованием опции LANGLEVEL.

Описание того, как обрабатываются строки с нулевым символом–ограничителем в случае подготовки с установленными для опции LANGLEVEL значениями MIA или SAA1, смотрите в разделе *Application Development Guide*.

По умолчанию для CNULREQD задается YES. Это приводит к тому, что строки с нулевым символом–ограничителем интерпретируются в соответствии со стандартами MIA. При подключении к серверу DB2 Universal Database for OS/390 настоятельно рекомендуется для CNULREQD задать YES. Прикладные программы с кодировкой по стандартам SAA1 (в отношении строк с нулевым символом–ограничителем) необходимо связывать, установив для опции CNULREQD значение NO. В противном случае строки с нулевым символом–ограничителем будут интерпретированы в соответствии со стандартами MIA, даже если они подготовлены с использованием для LANGLEVEL установки SAA1.

Автономные SQLCODE и SQLSTATE

Автономные переменные SQLCODE и SQLSTATE, как определено в ISO/ANS SQL92, поддерживаются через опцию прекомпиляции LANGLEVEL SQL92E. Во время прекомпиляции будет выдано предупреждение SQL0020W, указывающее на то, что LANGLEVEL не поддерживается. Это предупреждение применимо только к функциям, перечисленным под LANGLEVEL MIA в книге *Command Reference*, представляющим собой поднабор LANGLEVEL SQL92E.

Определение порядка сортировки

Различия между EBCDIC и ASCII приводят к различиям в порядках сортировки в разных программных продуктах для баз данных, а также влияют на условия ORDER BY и GROUP BY. Один из путей минимизации таких различий – создать пользовательскую последовательность слияния, соответствующую порядку сортировки EBCDIC. Последовательность слияния можно задавать только при создании новой базы данных. Дополнительную информацию смотрите в книгах *Application Development Guide*, *Administrative API Reference* и *Command Reference*.

Примечание: В DB2 Universal Database for OS/390 таблицы баз данных теперь могут храниться в формате ASCII. Это ускоряет обмен данными между DB2 Connect и DB2 Universal Database for OS/390, а также устраняет необходимость задавать процедуры полей, которые в противном случае пришлось бы использовать для преобразования данных и изменения их последовательности.

Поддержание реляционной целостности

В разных системах реляционные ограничения обрабатываются по-разному:

DB2 Universal Database for OS/390

Для того чтобы можно было создать внешний ключ с использованием первичного ключа, предварительно на первичном ключе должен быть создан индекс. Таблицы могут ссылаться сами на себя.

DB2 for VSE & VM

Для внешнего ключа автоматически создается индекс. Таблицы не могут ссылаться сами на себя.

DB2 Universal Database for AS/400

Для внешнего ключа автоматически создается индекс. Таблицы могут ссылаться сами на себя.

DB2 Universal Database

Для баз данных DB2 Universal Database автоматически создается индекс для уникального ограничения, включая первичный ключ. Таблицы могут ссылаться сами на себя.

Другие правила меняются в зависимости от уровней каскадности.

Блокировка

На некоторые прикладные программы может оказывать влияние способ выполнения блокировки сервером баз данных. Например, прикладные программы, разработанные на основе блокировки на уровне строки и стабильности уровня изоляции указателя, не переносятся напрямую в системы, выполняющие блокировку на уровне страницы. При таких кардинальных различиях прикладным программам может потребоваться настройка.

У программных продуктов DB2 Universal Database for OS/390 и DB2 Universal Database есть возможность устанавливать максимальное время блокировки и возвращать прикладным программам код ошибки при его истечении.

Различия в SQLCODE и SQLSTATE

Различные программные продукты для реляционных баз данных фирмы IBM не всегда выдают одинаковые SQLCODE для похожих ошибок. С этим можно бороться двумя способами:

- Использовать SQLSTATE вместо SQLCODE для конкретной ошибки.

SQLSTATE имеют примерно одинаковое значение в разных программных продуктах для баз данных, и эти программные продукты выдают SQLSTATE, соответствующие SQLCODE.

- Преобразовать SQLCODE из одной системы в другую.

По умолчанию DB2 Connect отображает SQLCODE и элементы из каждой системы сервера хоста или AS/400 фирмы IBM в вашу систему DB2 Universal Database. Если вы хотите переопределить используемое по умолчанию отображение или вы используете сервер баз данных, у которого нет отображения SQLCODE (сервер баз данных не IBM), можно задать свой собственный файл отображения SQLCODE. Отображение SQLCODE можно также отключить.

Дополнительную информацию смотрите в разделе Глава 11, “Отображение SQLCODE” на стр. 123.

Использование системных каталогов

Системные каталоги различаются в разных программных продуктах для баз данных фирмы IBM. Многие различия можно маскировать при помощи производных таблиц. Информацию смотрите в документации к используемому вами серверу баз данных.

Функции каталогов в CLI обходят эту проблему, поддерживая одни и те же API и одни и те же наборы результатов для запросов каталогов во всем семействе DB2.

Переполнения при численных преобразованиях в процессе присваивания

Переполнения при численных преобразованиях в процессе присваивания могут обрабатываться разными программными продуктами для реляционных баз данных фирмы IBM различными способами. Рассмотрим, например, выборку столбца типа float в переменную хоста integer из DB2 Universal Database for OS/390 и из DB2 Universal Database. При преобразовании значения float в значение integer может произойти переполнение. По умолчанию DB2 Universal Database for OS/390 возвратит прикладной программе SQLCODE

предупреждения и нулевое значение. DB2 Universal Database возвратит ошибку переполнения преобразования. Рекомендуется, чтобы прикладные программы избегали переполнений при числовых преобразованиях в процессе присваивания, используя переменные хоста надлежащего размера.

Уровни изоляции

При подготовке или связывании прикладной программы DB2 Connect принимает следующие уровни изоляции:

- RR** Многократное чтение
- RS** Стабильность чтения
- CS** Стабильность на уровне указателя
- UR** Чтение непринятого
- NC** Нет принятия

Уровни изоляции перечислены в порядке убывания защиты. Если сервер хоста или AS/400 не поддерживает заданный уровень изоляции, используется ближайший наиболее высокий поддерживаемый уровень.

В таблице Табл. 2 показаны результаты применения уровней изоляции на каждом из серверов прикладных программ хоста или AS/400.

Таблица 2. Уровни изоляции

DB2 Connect	DB2 Universal Database for OS/390	DB2 for VSE & VM	DB2 Universal Database for AS/400	DB2 Universal Database
RR	RR	RR	примечание 1	RR
RS	примечание 2	RR	COMMIT(*ALL)	RS
CS	CS	CS	COMMIT(*CS)	CS
UR	примечание 3	CS	COMMIT(*CHG)	UR
NC	примечание 4	примечание 5	COMMIT(*NONE)	UR

Примечания:

1. На DB2 Universal Database for AS/400 нет эквивалентной опции COMMIT, совпадающей с RR. DB2 Universal Database for AS/400 поддерживает RR путем блокировки всей таблицы.
2. Приводит к RR для Версии 3.1 и к RS для Версии 4.1 с APAR PN75407 или Версии 5.1.
3. Приводит к CS для Версии 3.1 и к UR для Версии 4.1 или Версии 5.1.
4. Приводит к CS для Версии 3.1 и к UR для Версии 4.1 с APAR PN60988 или Версии 5.1.
5. Уровень изоляции NC не поддерживается с DB2 for VSE & VM.

С DB2 Universal Database for AS/400 можно обращаться к таблице без журнала, если прикладная программа связана с уровнем изоляции UR и для блокировки установлено ALL или если для уровня изоляции установлено NC.

Хранимые процедуры

- Вызов

Программа клиента может вызывать программу сервера при помощи оператора SQL CALL. В этом случае все серверы работают несколько по-разному.

OS/390

Имя схемы не должно быть длиннее 8 байт, имя процедуры не должно быть длиннее 18 байт, а хранимая процедура должна быть определена на сервере в каталоге SYSIBM.SYSPROCEDURES.

VSE или VM

Имя процедуры не должно быть длиннее 18 байт и должно быть определено на сервере в каталоге SYSTEM.SYSROUTINES.

OS/400

Имя процедуры должно быть идентификатором SQL. Для задания реального полного имени (имени схемы или имени собрания) с целью локализации хранимой процедуры можно также использовать операторы DECLARE PROCEDURE или CREATE PROCEDURE.

Все операторы CALL, посылаемые DB2 for AS/400 из REXX/SQL, должны быть подготовлены динамически и выполнены прикладной программой, так как оператор CALL в REXX/SQL отображается на CALL USING DESCRIPTOR.

Синтаксис оператора SQL CALL смотрите в справочнике *SQL Reference*. Информацию о том, как использовать хранимые процедуры при написании прикладных программ, смотрите в книге *Application Development Guide*.

Программу сервера в DB2 Universal Database можно вызвать с тем же самым соглашением о параметрах, которое используют программы серверов в DB2 Universal Database for OS/390, DB2 Universal Database for AS/400 или DB2 for VSE & VM. Дополнительную информацию о вызове хранимых процедур DB2 Universal Database смотрите в книге *Application Development Guide*. Про соглашения о параметрах на других платформах можно узнать из документации к программным продуктам DB2 для этих платформ.

Все операторы SQL в хранимой процедуре выполняются как часть рабочей единицы SQL, запущенной программой клиента SQL.

- Не передавайте значения индикаторов со специальным значением хранимым процедурам и не получайте от них таких значений.

Между разными DB2 Universal Database передаются любые значения индикаторных переменных. Однако при использовании DB2 Connect в индикаторных переменных можно передавать только 0, -1 и -128.

- Необходимо определить параметр для возврата любых ошибок, обнаруженных прикладной программой сервера.

Программа сервера в DB2 Universal Database может обновлять SQLCA для возвращения любой ошибки или предупреждения, но у хранимой процедуры в DB2 Universal Database for OS/390 или DB2 Universal Database for AS/400 такой возможности нет. Если вы хотите возвращать от хранимой процедуры код ошибки, его необходимо передавать как параметр. SQLCODE и SQLCA устанавливаются сервером только для ошибок, обнаруженных системой.

- Из серверов прикладных программ хоста или AS/400 только DB2 for VSE & VM Версии 7 или более новой и DB2 Universal Database for OS/390 Версии 5.1 могут в настоящее время возвращать наборы результатов хранимых процедур.

Построитель хранимых процедур

Построитель хранимых процедур DB2 предоставляет удобную среду разработки для создания, установки и тестирования хранимых процедур. Он позволяет сосредоточиться на создании логики хранимой процедуры вместо подробностей регистрации, построения и установки хранимых процедур на сервере DB2. Кроме того, при помощи построителя хранимых процедур можно разрабатывать хранимые процедуры в одной операционной системе, а создавать их затем в другой операционной системе сервера.

Построитель хранимых процедур представляет собой графическую программу, поддерживающую быструю разработку. При помощи построителя хранимых процедур можно выполнять следующие задачи:

- Создавать новые хранимые процедуры
- Создавать хранимые процедуры на локальных и удаленных серверах DB2
- Изменять и перестраивать существующие хранимые процедуры
- Тестировать и отлаживать существующие хранимые процедуры

Построитель хранимых процедур можно запускать как в виде отдельной прикладной программы из группы программ DB2 Universal Database, так и из одной из следующих сред разработки:

- Microsoft Visual Studio
- Microsoft Visual Basic
- IBM VisualAge for Java

Построитель хранимых процедур можно также запускать из Центра управления для DB2 for OS/390. В Центре управления построитель хранимых процедур можно запустить в качестве отдельного процесса из меню Tools (Инструменты), из панели инструментов или из папки хранимых процедур. Кроме того, из окна проекта построителя хранимых процедур можно экспортировать одну или несколько выбранных хранимых процедур SQL, построенных для сервера DB2 for OS/390, в заданный файл, который можно запускать из командной строки.

Построитель хранимых процедур управляет вашей работой посредством использования проектов. Каждый из проектов построителя хранимых процедур сохраняет ваши соединения в особых базах данных, например на серверах DB2 for OS/390. Кроме этого, можно создавать фильтры для вывода поднаборов хранимых процедур в каждой из баз данных. При открытии нового или существующего проекта построителя хранимых процедур можно фильтровать хранимые процедуры так, чтобы просматривать их на основе имен, схем, языка или ID собрания (только для OS/390).

Информация о соединении сохраняется в проекте построителя хранимых процедур; поэтому, когда вы открываете существующий проект, вам автоматически предлагается ввести ID пользователя и пароль для базы данных. Используя мастер вставки хранимых процедур SQL, можно строить хранимые процедуры SQL на сервере DB2 for OS/390. Для построения хранимой процедуры SQL на сервере DB2 for OS/390 можно задать конкретную среду компиляции, предварительной компоновки, компоновки, связывания, времени выполнения, WLM, а также дополнительные опции безопасности.

Дополнительно можно получить информацию о стоимости SQL для хранимой процедуры SQL, включая время работы процессора и другие параметры DB2, имеющие отношение к потоку, в котором работает данная хранимая процедура SQL. В частности, можно получить информацию о стоимости для времени ожидания блокировок содержимого, числа полученных страниц, числа чтений ввода/вывода и числа записей ввода/вывода.

Для получения информации о стоимости построитель хранимых процедур связывается с сервером DB2 for OS/390, выполняет оператор SQL и вызывает хранимую процедуру (DSNWSPM), определяющую, сколько процессорного времени использует хранимая процедура SQL.

Составной SQL NOT ATOMIC

Составной SQL позволяет сгруппировать несколько операторов SQL в один исполнимый блок. Это может сократить передачу данных по сети и уменьшить время ответа.

DB2 Connect поддерживает составные операторы SQL NOT ATOMIC. Это означает, что обработка составного SQL продолжается после ошибки. (Для

составного оператора SQL ATOMIC, который не поддерживается DB2 Connect, ошибка приводит к отмене всей группы составного SQL.)

Операторы будут выполняться вплоть до остановки сервером прикладных программ. В общем случае выполнение составного оператора SQL будет остановлено только в случае серьезных ошибок.

Составной оператор SQL NOT ATOMIC можно использовать со всеми поддерживаемыми серверами прикладных программ хоста или AS/400.

Если происходит несколько ошибок SQL, в поле SQLERRMC SQLCA возвращаются коды SQLSTATE первых семи ошибочных операторов с сообщением о многочисленных ошибках. Дополнительную информацию смотрите в книге *SQL Reference*.

Многоузловое изменение с DB2 Connect

DB2 Connect позволяет выполнять многоузловое изменение (другое название – двухфазное принятие). Многоузловое изменение – это изменение нескольких баз данных в одной распределенной единице работы. Для использования этой возможности должны соблюдаться некоторые условия:

- Ваша прикладная программа должна быть прекомпилирована с опциями CONNECT 2 и SYNCPOINT TWOPHASE.
- При сетевых соединениях SNA можно использовать поддержку двухфазного принятия, предоставляемую функцией менеджера точек синхронизации DB2 Connect Enterprise Edition Версии 7 в AIX, OS/2 и Windows NT. Это дает возможность участвовать в распределенной единице работы следующим серверам баз данных хостов:
 - DB2 for AS/400 Версии 3.1 или более новой
 - DB2 for MVS/ESA Версии 3.1 или более новой
 - DB2 for OS/390 Версии 5.1 или более новой
 - DB2 for VM & VSE Версии V5.1 или более новой

Сказанное выше верно для собственных прикладных программ UDB DB2 и прикладных программ, скоординированных внешним монитором транзакций, таким как IBM TXSeries, CICS for Open Systems, BEA Tuxedo, Encina Monitor и Microsoft Transaction Server.

Примечание: Дополнительную информацию о BEA Tuxedo смотрите в разделе “Использование DB2 Connect с мониторами транзакций” на стр. 36. Дополнительную информацию о концентраторе XA смотрите в разделе “Концентратор соединений DB2 Connect” на стр. 141.

- Для сетевых соединений TCP/IP сервер DB2 for OS/390 Версии 5.1 или более новой может участвовать в распределенной единице работы. Если прикладная программа управляется монитором управления транзакциями,

таким как IBM TXSeries, CICS for Open Systems, Encina Monitor или Microsoft Transaction Server, необходимо использовать менеджер точек синхронизации.

Если общий сервер DB2 Connect Enterprise Edition используется для доступа к данным хоста по соединениям TCP/IP как собственными прикладными программами DB2, так и прикладными программами монитора транзакций, необходимо использовать менеджер точек синхронизации.

Если для доступа к данным хоста с использованием сетевых протоколов как SNA, так и TCP/IP используется один сервер DB2 Connect Enterprise Edition и требуется двухфазное принятие, необходимо использовать менеджер точек синхронизации. Это справедливо как для прикладных программ DB2, так и для прикладных программ монитора транзакций.

Операторы SQL сервера хоста или AS/400, поддерживаемые DB2 Connect

Следующие операторы успешно компилируются для обработки сервером хоста или AS/400, но не системой DB2 Universal Database:

- ACQUIRE
- DECLARE (модификатор.(спецификатор.)имя_таблицы TABLE ...
- LABEL ON

Эти операторы также поддерживаются процессором командной строки.

Следующие операторы поддерживаются для обработки сервера хоста или AS/400, но не добавляются в связанный файл или пакет и не поддерживаются процессором командной строки:

- DESCRIBE имя_оператора INTO имя_описателя USING NAMES
- PREPARE имя_оператора INTO имя_описателя USING NAMES FROM...

Прекомпилятор при работе предполагает, что:

- Переменные хоста являются входными переменными
- Оператор назначается уникальному номеру раздела.

Операторы SQL сервера хоста или AS/400, не поддерживаемые DB2 Connect

Следующие операторы SQL не поддерживаются DB2 Connect и не поддерживаются процессором командной строки:

- COMMIT WORK RELEASE
- DECLARE имя_состояния, имя_оператора STATEMENT
- DESCRIBE имя_оператора INTO имя_описателя USING xxxx (где xxxx — ANY, BOTH или LABELS)

- PREPARE имя_оператора INTO имя_описателя USING xxxx FROM :переменная_хоста (где xxxx – ANY, BOTH или LABELS)
- PUT ...
- ROLLBACK WORK RELEASE
- SET :переменная_хоста = CURRENT ...

Расширенные динамические операторы SQL DB2 for VSE & VM вызывают генерацию кодов SQLCODE –104 и синтаксических ошибок.

Реализация учета оплаты в DB2 Universal Database for OS/390

Часто при работе с DB2 Universal Database for OS/390 применяются методы слежения за ресурсами, позволяющие системным администраторам связывать использование ресурсов с доступом отдельных пользователей. Это позволяет требовать оплату за потребляемые ресурсы с отдельных пользователей или подразделений. Такая практика часто называется *учетом оплаты*.

Продукты DB2 Connect позволяют системным администраторам следить за ресурсами мэйнфрейма, потребляемыми пользователями, которые обращаются к базам данных через DB2 Connect. Учетные строки можно использовать для отправки учетных данных от DB2 Connect на DB2 для сервера баз данных. В учетной строке сочетаются данные, сгенерированные системой, и данные пользователя. Эти данные позволяют системным администраторам связывать использование ресурсов с каждым обращением пользователя и соответственно требовать от пользователей оплаты.

Учетная строка посылается с помощью параметра DRDA PRDDTA. Поскольку содержимое этого параметра не определено в архитектуре DRDA, нет гарантии, что сервер прикладных программ распознает в этих данных учетные данные. В данный момент PRDDTA поддерживается только на системах MVS и OS/390. Строка записывается как учетная запись.

Учетная строка состоит из 56 байтов, сгенерированных DB2 Connect (префикс), за которыми следуют до 199 байт, задаваемых пользователем (суффикс), максимальная длина – 255.

В Табл. 3 показаны поля, генерируемые системой. Каждое из этих полей дополняется справа пробелами.

Таблица 3 (Стр. 1 из 2). Поля учетной строки, генерируемые DB2 Connect		
Имя поля	Длина	Описание
acct_str_len	1	Шестнадцатеричное значение, соответствующее длине учетной строки минус 1. Например, X'3C'.

<i>Таблица 3 (Стр. 2 из 2). Поля учетной строки, генерируемые DB2 Connect</i>		
Имя поля	Длина	Описание
client_prdid	8	ID продукта для программы клиента. Например, ID продукта для DB2 Universal Database, Версия 7 – SQL07010.
client_platform	18	Платформа клиента, например AIX, OS/2, DOS или Windows.
client_appl_name	20	Первые 20 символов имени программы пользователя, например, payroll.
client_authid	8	authid прикладной программы пользователя, например, SMITH.
suffix_len	1	Шестнадцатеричное значение длины предоставляемого пользователем суффикса. X'00' означает, что предоставляемый пользователем суффикс отсутствует.

Предоставляемый пользователем суффикс может быть одним из следующих:

- Значение, указанное прикладной программой с помощью API sqlsact()
- Значение переменной среды DB2ACCOUNT
- Значение параметра конфигурации DFT_ACCOUNT_STR (учетная строка по умолчанию)
- Пустая строка

Если суффикс содержит более 199 символов, он усекается. Чтобы строка была правильно преобразована при передаче на сервер хоста или AS/400, нужно использовать символы A–Z, 0–9 и символ подчеркивания.

Рекомендуется заполнять учетную строку с помощью вызова API. Прикладная программа должна вызвать API до установления соединения с базой данных. Если вы хотите изменить учетную строку в прикладной программе (например, чтобы послать другую строку перед соединением с другой базой данных), вызовите API снова. В противном случае значение будет действительно до конца работы программы.

Если API `sqlsact()` не был вызван до первого соединения с базой данных, используется переменная среды `DB2ACCOUNT`. Это значение действует до конца работы программы или фонового процесса процессора командной строки. Чтобы задать новый суффикс учетной строки после установления первого соединения с базой данных, используйте API `sqlsact()` или же завершите программу или фоновый процесс процессора командной строки и перезапустите их, задав для `DB2ACCOUNT` новое значение.

Если `DB2ACCOUNT` не задана, используется параметр конфигурации системы `DFT_ACCOUNT_STR`. Это умолчание может быть полезно для клиентов баз данных, у которых нет возможности отправить учетную строку `DB2 Connect`. Если и этот параметр не задан, посылается пустая строка.

Ниже приведены примеры учетных строк:

<code>x'3C'SQL070100S/2</code>	<code>cheque</code>	<code>SMITH</code>	<code>x'05'DEPT1</code>
<code>x'37'SQL070100S/2</code>	<code>cheque</code>	<code>SMITH</code>	<code>x'00'</code>

В первом примере задаваемый пользователем суффикс – `DEPT1`. Во втором примере это пустая строка.

Отправка учетной информации на сервер DB2 for OS/390

Многие заказчики, работающие на мэйнфреймах, считают детальный учет ресурсов, используемых различными прикладными программами, важной частью рабочих процедур. DB2 for OS/390 предоставляет богатые возможности для получения отчетов об использовании ресурсов, позволяющих подразделениям информационных систем взимать плату с отдельных подразделений–пользователей за расходуемые ими ресурсы мэйнфреймов. Этот процесс обычно называют *учетом оплаты*. Продукты DB2 Connect с помощью существующих отчетов и процедур учета использования ресурсов позволяют вести детальный учет ресурсов хоста, используемых программами PC и UNIX.

DB2 Connect реализует гибкий подход для этой возможности, предоставляя:

- Учетную строку по умолчанию для любых применений, генерируемую определенным сервером DB2 Connect Enterprise Edition
- Два механизма для задания отдельными пользователями или программами счета, на который должны быть записаны расходы

Задание учетной строки

Учетная строка по умолчанию задается в параметре конфигурации `dft_account_str` на рабочей станции DB2 Connect. Этот механизм полезен для клиентов баз данных, не имеющих возможности отправить учетную строку

DB2 Connect. Пример такого клиента – программы, разработанные до появления продуктов Версии 2.

Есть два способа переопределить учетную строку по умолчанию, заданную на сервере DB2 Connect:

- При помощи API задания учетной строки: `sqlsact()`

API `sqlsact()` вызывается перед установлением программой соединения с базой данных. Этот способ имеет следующие преимущества:

- Вызов API позволяет избежать издержек, связанных с чтением значения реестра.
- Нет необходимости вызывать этот API снова, пока вам не потребуется задать новую учетную строку для последующих требований соединения. Если используется значение из реестра, необходимо завершить процесс программы, переопределить `DB2ACCOUNT` и затем запустить процесс снова.

Дополнительную информацию об использовании этого API смотрите в руководстве *Administrative API Reference*.

- Используя значение реестра `DB2ACCOUNT` на рабочей станции клиента.

Если до первого соединения с базой данных API `sqlsact()` не был вызван, читается значение реестра `DB2ACCOUNT`. Учетная строка используется для всех последующих требований соединения.

Примечание: При задании учетной строки необходимо соблюдать следующие правила:

1. Использовать только символы A–Z, 0–9 или '_' (подчеркивание).
2. Не превышать максимальную длину строки (199 байт); более длинные строки усекаются.

Литература

Следующие публикации могут помочь при разработке прикладных программы для работы в распределенной среде:

- Книги по прикладному программированию для конкретных систем баз данных могут содержать сведения, различающиеся для разных продуктов.
- Книги по SQL для конкретных продуктов баз данных помогут определить, какие операторы SQL поддерживаются данным продуктом, и уточнить их синтаксис.
- *DB2 Universal Database for OS/390 Reference for Remote DRDA Requesters and Servers* содержит последнюю информацию об учете оплаты для пользователей DB2 Universal Database for OS/390.

- В справочнике *SQL Reference* описываются на высоком уровне различия между разными продуктами реляционных баз данных IBM и рассказывается, как избежать влияния некоторых различий.
- Публикации по DRDA содержат сведения по планированию, связи, программированию и диагностике неисправностей в среде DRDA. Список заголовков и номеров заказов смотрите в разделе “Электронные публикации, связанные с DRDA” на стр. 19.

Глава 5. Запуск собственных прикладных программ

К базам данных DB2 могут обращаться различные типы прикладных программ:

- Прикладные программы, разработанные при помощи клиента разработки программ DB2, который включает в себя встроенный SQL, интерфейсы прикладного программирования (API), хранимые процедуры, пользовательские функции и вызовы CLI DB2
- Прикладные программы ODBC, например Lotus Approach
- Прикладные программы и апплеты JDBC
- Макрокоманды Net.Data, содержащие HTML и SQL

Прикладная программа на клиенте DB2 может обращаться к удаленной базе данных, не зная ее физического расположения. Клиент DB2 определяет расположение этой базы данных, управляет передачей запросов серверу баз данных и возвращает результаты.

Обычно для запуска прикладной программы (клиента баз данных), используются следующие шаги:

Шаг 1. Убедитесь, что сервер сконфигурирован и запущен.

Убедитесь, что менеджер баз данных запущен на сервере баз данных, с которым соединяется прикладная программа. Если это не так, нужно перед запуском прикладной программы дать на сервере команду **db2start**.

Шаг 2. Убедитесь, что установлено соединение с базой данных, которую использует прикладная программа.

Шаг 3. Свяжите утилиты и прикладные программы с базой данных. Информацию о связывании утилит смотрите в разделе “Связывание утилит баз данных.”

Шаг 4. Запустите прикладную программу.

Связывание утилит баз данных

Перед использованием утилит баз данных (`import`, `export`, `reorg`, процессор командной строки) и DB2 CLI с базой данных их необходимо связать с этой базой данных. В сетевой среде, если работает несколько клиентов, выполняющихся в разных операционных системах или использующих различные версии или служебные уровни DB2, необходимо связать утилиты один раз для каждой комбинации операционная система – версия DB2.

При связывании утилит создается *пакет* — объект, включающий всю необходимую информацию для обработки конкретных операторов SQL из одиночного исходного файла.

Связанные файлы группируются в различные файлы .lst в подкаталоге bnd каталога установки. Каждый файл относится к конкретному серверу.

Выполнение программ CLI/ODBC

Среда выполнения интерфейса уровня вызовов DB2 (CLI) и драйвер DB2 CLI/ODBC — необязательные компоненты клиентов DB2, которые можно выбрать во время установки.

Такая поддержка позволяет программам, разработанным с использованием API ODBC и DB2 CLI работать с любым сервером DB2. Поддержку разработки программ DB2 CLI обеспечивает клиент разработки программ DB2, поставляемый вместе с сервером DB2.

Чтобы прикладная программа DB2 CLI или ODBC могла обращаться к DB2, пакеты DB2 CLI должны быть связаны с сервером. Хотя это произойдет автоматически при первом соединении (если пользователь обладает необходимыми полномочиями на связывание пакетов), рекомендуется, чтобы администратор сделал это первым для каждой версии клиента на каждой платформе, которая будет обращаться к серверу.

Чтобы позволить прикладным программам DB2 CLI и ODBC обращаться к базам данных DB2, нужно выполнить следующие основные действия в системе клиента. В этих инструкциях подразумевается, что успешно установлено соединение с DB2 с использованием правильных ID пользователя и пароля. В зависимости от типа платформы многие из этих шагов выполняются автоматически. Полное описание смотрите в разделе, посвященном конкретной используемой платформе.

Шаг 1. При помощи Ассистента конфигурирования клиента (ССА) добавьте базу данных (если для клиента и сервера используются разные компьютеры), чтобы экземпляры и базы данных были известны Центру управления, затем добавьте экземпляры и базы данных для этой системы. Если эта программа недоступна, можно использовать команду **catalog** процессора командной строки.

Шаг 2. Драйвер DB2 CLI/ODBC — необязательный компонент, который можно выбрать при установке клиента DB2 на платформах Windows. Убедитесь, что этот компонент был выбран при установке. В OS/2 необходимо при помощи значка **Установка драйвера ODBC** установить драйвер DB2 CLI/ODBC и менеджер драйверов ODBC. На платформах UNIX драйвер DB2 CLI/ODBC устанавливается автоматически вместе с клиентом.

Шаг 3. Для обращения к базе данных DB2 из ODBC:

- a. Должен быть установлен менеджер драйверов ODBC Microsoft или другого производителя (по умолчанию это делается только при установке DB2 в 32-битных системах Windows).
- b. Базы данных DB2 должны быть зарегистрированы в качестве источников данных ODBC. Менеджер драйверов ODBC не читает информацию каталога DB2 – вместо этого он использует собственный список источников данных.
- c. Если для таблицы DB2 не существует индекс уникальности, многие прикладные программы ODBC будут открывать эту таблицу только для чтения. Для каждой таблицы DB2, которая должна обновляться прикладными программами ODBC, нужно создать индекс уникальности. Смотрите описание оператора **CREATE INDEX** в справочнике *SQL Reference*. При помощи Центра управления можно изменить параметры таблицы, затем выбрать закладку **Первичный ключ** и переместить один или несколько столбцов из списка доступных столбцов в список столбцов первичного ключа. Все столбцы, выбираемые как компоненты первичного ключа, должны быть определены как NOT NULL.

Шаг 4. При необходимости можно задать различные параметры конфигурации CLI/ODBC, чтобы изменить поведение DB2 CLI/ODBC и поведение использующих их прикладных программ.

Если выполнены описанные выше действия для установки поддержки ODBC и базы данных DB2 добавлены в качестве источников данных ODBC, прикладные программы ODBC могут обращаться к этим базам данных.

Подробности доступа CLI/ODBC для конкретных платформ



Подробное описание того, как на конкретных платформах обеспечить прикладным программам DB2 CLI и ODBC доступ к DB2, содержит следующие разделы:

- “Доступ к DB2 с использованием CLI/ODBC из клиентов 32-битных операционных систем Windows”
- “Доступ к DB2 с использованием CLI/ODBC из клиентов 32-битных операционных систем Windows” на стр. 66

Доступ к DB2 с использованием CLI/ODBC из клиентов 32-битных операционных систем Windows

Чтобы прикладные программы DB2 CLI и ODBC могли успешно обращаться к базе данных DB2 из клиента Windows, выполните в системе клиента следующие действия:

Шаг 1. Необходимо внести в каталог базу данных DB2 (и узел, если это удаленная база данных). Используйте для этого CCA (или процессор командной строки).

Дополнительную информацию смотрите в электронной справке CCA (или в описаниях команд **CATALOG DATABASE** и **CATALOG NODE** в руководстве *Command Reference*).

- Шаг 2. Убедитесь, что установлены Microsoft ODBC Driver Manager и драйвер DB2 CLI/ODBC. В 32-битных системах Windows оба эти компонента устанавливаются вместе с DB2, если только установка компонента ODBC не была отменена вручную. DB2 не запишет новую версию менеджера драйверов ODBC Microsoft, если этот менеджер найден в системе.

Чтобы проверить, установлены ли на компьютере оба эти компонента:

- a. Щелкните по значку Microsoft ODBC Data Sources на Панели управления или введите команду **odbcad32.exe** в командной строке.
- b. Щелкните по закладке **Drivers** (Драйверы).
- c. Проверьте, есть ли в списке драйверов драйвер "IBM DB2 ODBC DRIVER".

Если Microsoft ODBC Driver Manager или драйвер IBM DB2 CLI/ODBC не установлен, повторно запустите в 32-битной системе Windows процесс установки DB2 и выберите компонент ODBC.

- Шаг 3. Зарегистрируйте базу данных DB2 в качестве *источника данных* для менеджера драйверов ODBC. В 32-битных системах Windows можно сделать источник данных доступным для всех пользователей системы (системный источник данных) или только для текущего пользователя (пользовательский источник данных). Используйте для добавления источника данных один из этих методов:

- При помощи CCA:
 - a. Выберите алиас базы данных DB2, которую нужно добавить в качестве источника данных.
 - b. Нажмите кнопку **Свойства**. Откроется окно Свойства базы данных.
 - c. Включите переключатель **Зарегистрировать эту базу данных для ODBC**.
 - d. В 32-битных системах Windows при помощи радиокнопок определите источник данных как пользовательский или как системный источник данных.
- При помощи программы **Microsoft 32-bit ODBC Administration tool** (запустить ее можно при помощи значка на Панели управления или команды **odbcad32.exe** в командной строке):
 - a. В 32-битных системах Windows по умолчанию появляется список пользовательских источников данных. Если нужно добавить системный источник данных, нажмите кнопку

System DSN (Системный DSN) или выберите закладку **System DSN** (в зависимости от платформы).

- b. Нажмите кнопку **Добавить**.
 - c. Дважды щелкните по строке IBM DB2 ODBC Driver в списке.
 - d. Выберите базу данных DB2, которую нужно добавить, и нажмите кнопку **ОК**.
- Чтобы зарегистрировать базу данных DB2 в качестве источника данных для менеджера драйверов ODBC, в 32-битных системах Windows можно ввести команду в процессоре командной строки. Администратор может создать сценарий процессора командной строки для регистрации нужных баз данных. Этот сценарий можно затем выполнить на всех компьютерах, которые должны обращаться к базам данных DB2 через ODBC.

В руководстве *Command Reference* содержится дополнительная информация о команде CATALOG:

```
CATALOG [ user | system ] ODBC DATA SOURCE
```

Шаг 4. Сконфигурируйте драйвер DB2 CLI/ODBC при помощи CCA: (Это не обязательно)

- a. Выберите алиас базы данных DB2, которую нужно сконфигурировать.
- b. Нажмите кнопку **Свойства**. Откроется окно Свойства базы данных.
- c. Нажмите кнопку **Параметры**. Откроется окно Параметры CLI/ODBC.
- d. Нажмите кнопку **Дополнительные**. Откроется окно, в котором можно задать параметры конфигурации. Эти параметры относятся к данному *алиасу* базы данных и влияют на работу всех прикладных программ DB2 CLI/ODBC, обращающихся к этой базе данных. Значения всех этих параметров объясняются в электронной справке, в электронном руководстве *Дополнение по установке и настройке*.

Шаг 5. Если установлен доступ ODBC (как описано выше), можно обращаться к данным DB2, используя прикладные программы ODBC. Запустите прикладную программу ODBC и перейдите в окно Open (Открыть). Выберите тип файла **ODBC databases** (Базы данных ODBC). В этом списке можно выбрать базы данных DB2, добавленные в качестве источников данных ODBC. Если для таблицы не существует индекс уникальности, многие прикладные программы ODBC будут открывать эту таблицу только для чтения.

Доступ к DB2 с использованием CLI/ODBC из клиентов 32-битных операционных систем Windows

Чтобы прикладные программы DB2 CLI и ODBC могли успешно обращаться к базе данных DB2 из клиента OS/2, выполните в системе клиента следующие действия:

1. Необходимо внести в каталог базу данных DB2 (и узел, если это удаленная база данных). Используйте для этого CCA (или процессор командной строки).

Дополнительную информацию смотрите в электронной справке по CCA (или в описании команд **CATALOG DATABASE** и **CATALOG NODE** в книге *Command Reference*).

2. Если для обращения к данным DB2 используются прикладные программы ODBC, выполните следующие действия. (Если используются только прикладные программы CLI, пропустите этот шаг и перейдите к следующему шагу.)

- a. Проверьте, установлен ли менеджер драйверов ODBC. Менеджер драйверов ODBC не устанавливается вместе с DB2; мы предлагаем использовать менеджер драйверов ODBC, поставляемый с используемой прикладной программой ODBC. Убедитесь также, что установлен драйвер DB2 CLI/ODBC:

- 1) Запустите средство управления ODBC, как описано в его документации. Обычно это делают одним из следующих способов:

- Дважды щелкните по папке **ODBC** в системе OS/2 и затем дважды щелкните по значку **ODBC Administrator** (Администратор ODBC).
- Введите **odbcadm.exe** в командной строке.

Откроется окно Data Sources (Источники данных).

- 2) Нажмите кнопку **Drivers** (Драйверы). Откроется окно Drivers (Драйверы).
- 3) Проверьте, есть ли в списке драйверов драйвер "IBM DB2 ODBC DRIVER".

Если менеджер драйверов ODBC не установлен, выполните инструкции по его установке для используемой прикладной программы ODBC.

Если драйвер IBM DB2 CLI/ODBC не установлен, дважды щелкните по значку **Install ODBC Driver** (Установить драйвер ODBC) в папке DB2, чтобы установить драйвер DB2 CLI/ODBC.

- b. Зарегистрируйте базу данных DB2 в качестве *источника данных* для менеджера драйверов ODBC, используя один из следующих методов:
 - При помощи CCA:
 - 1) Выберите алиас базы данных DB2, которую нужно добавить в качестве источника данных.

- 2) Нажмите кнопку **Свойства**.
- 3) Включите переключатель **Зарегистрировать эту базу данных для ODBC**.
- При помощи менеджера драйверов ODBC:
 - 1) Запустите менеджер драйверов ODBC, как описано в его документации. Обычно это делают одним из следующих способов:
 - Дважды щелкните по папке **ODBC** в системе OS/2 и затем дважды щелкните по значку **ODBC Administrator** (Администратор ODBC).
 - Введите **odbcadm.exe** в командной строке.
 - 2) В окне Data Sources (Источники данных) нажмите кнопку **Add** (Добавить). Откроется окно Add Data Source (Добавить источник данных).
 - 3) Дважды щелкните по строке IBM DB2 ODBC DRIVER в списке драйверов.
 - 4) Выберите базу данных DB2, которую нужно добавить, и нажмите кнопку **ОК**.
3. Сконфигурируйте драйвер DB2 CLI/ODBC при помощи CCA: (Это не обязательно)
 - a. Выберите алиас базы данных DB2, которую нужно сконфигурировать.
 - b. Нажмите кнопку **Свойства**. Откроется окно Свойства базы данных.
 - c. Нажмите кнопку **Параметры**. Откроется окно Параметры CLI/ODBC.
 - d. Нажмите кнопку **Дополнительные**. Откроется окно, в котором можно задать параметры конфигурации. Эти параметры относятся к данному алиасу базы данных и влияют на работу всех прикладных программ DB2 CLI/ODBC, обращающихся к этой базе данных. Значения всех этих параметров объясняются в электронной справке, а также в руководстве *Дополнение по установке и настройке*.
4. Если установлен доступ ODBC (как описано выше), можно обращаться к данным DB2, используя прикладные программы ODBC. Запустите прикладную программу ODBC и перейдите в окно Open (Открыть). Выберите тип файла **ODBC databases** (Базы данных ODBC). В этом списке можно выбрать базы данных DB2, добавленные в качестве источников данных ODBC. Если для таблицы не существует индекса уникальности, многие прикладные программы ODBC будут открывать эту таблицу только для чтения.

Подробная информация о конфигурации

В разделе “Подробности доступа CLI/ODBC для конкретных платформ” на стр. 63 изложена вся необходимая информация. Дополнительную информацию о настройке и использовании прикладных программ DB2 CLI и

ODBC смотрите в руководстве *Дополнение по установке и настройке*. (Электронное руководство *Дополнение по установке и настройке* находится в каталоге `x:\doc\ru\html`, где `x` – буква используемого дисковода CD-ROM, а `ru` – двухсимвольный код языка, например, `ru` для русского.) Эта информация может понадобиться, если недоступна поддержка инструментов DB2, а также полезна администраторам, которым нужны более подробные сведения.

В электронном руководстве *Дополнение по установке и настройке* рассмотрены следующие темы:

- Как связать драйвер DB2 CLI/ODBC с базой данных
- Как задать параметры конфигурации CLI/ODBC
- Задание конфигурации в файле `db2cli.ini`

Выполнение программ Java

Используя соответствующий набор инструментов для разработки программ Java (JDK) в системах AIX, HP-UX, Linux, OS/2, PTX, Silicon Graphics IRIX, Solaris Operating Environment или 32-битных системах Windows. JDK содержит Java Database Connectivity (JDBC) – API динамического SQL для языка Java.

Для поддержки DB2 JDBC необходимо включить компонент DB2 Java Enablement при установке клиента DB2. Используя поддержку DB2 JDBC, можно строить и выполнять прикладные программы и апплеты JDBC. Такие программы содержат только динамические операторы SQL и используют интерфейс вызовов Java для передачи операторов SQL к DB2.

Клиент разработки программ DB2 обеспечивает поддержку встроенных операторов SQL для языка Java (SQLJ). Используя поддержку DB2 SQLJ и поддержку DB2 JDBC, можно строить и выполнять прикладные программы и апплеты SQLJ. Такие программы содержат статические операторы SQL и используют встроенные операторы SQL, которые связаны с базой данных DB2.

Можно также использовать язык Java для создания на сервере хранимых процедур и пользовательских функций (UDF) JDBC и SQLJ.

Для построения и выполнения разных типов программ Java требуются различные компоненты DB2:

- Для построения прикладных программ JDBC необходимо установить клиент DB2 с компонентом DB2 Java Enablement. Для выполнения прикладных программ JDBC используемый клиент DB2 с компонентом DB2 Java Enablement должен соединиться с сервером DB2.
- Для построения прикладных программ SQLJ необходимо установить клиент разработки программ DB2 и клиент управления DB2 с компонентом DB2 Java Enablement. Для выполнения прикладных программ

SQLJ используемый клиент DB2 с компонентом DB2 Java Enablement должен соединиться с сервером DB2.

- Для построения апплетов JDBC необходимо установить клиент DB2 с компонентом DB2 Java Enablement. Для выполнения апплетов JDBC на компьютере клиента не требуются какие-либо компоненты DB2.
- Для построения апплетов SQLJ необходимо установить клиент разработки программ DB2 и клиент управления DB2 с компонентом DB2 Java Enablement. Для выполнения апплетов SQLJ на компьютере клиента не требуются какие-либо компоненты DB2.

Подробную информацию о построении и выполнении программ JDBC и SQLJ смотрите в руководстве *Application Building Guide*. Дополнительную информацию о создании программ DB2 на языке Java смотрите в руководстве *Application Development Guide*. В нем описываются создание и выполнение прикладных программ, апплетов, хранимых процедур и пользовательских функций JDBC и SQLJ.

Чтобы получить самую свежую информацию о DB2 Java, посмотрите Web-сайт по адресу: <http://www.ibm.com/software/data/db2/java>.

Настройка среды

Чтобы строить программы DB2 на языке Java, необходимо установить и сконфигурировать на используемом для разработки компьютере соответствующую версию набора инструментов для разработки программ Java (JDK). Чтобы выполнять программы DB2 Java, надо установить и сконфигурировать на компьютере разработки соответствующую версию Java Runtime Environment (JRE) или JDK. В следующей таблице перечислены JDK для различных компьютеров разработки:

AIX	IBM AIX Developer Kit, Java Technology Edition, Версия 1.1.8. На системах AIX, где не установлен JDK, этот JDK устанавливается автоматически при установке клиента разработки программ DB2.
HP-UX	HP-UX Developer's Kit for Java, Выпуск 1.1.8, фирмы Hewlett-Packard
Linux	IBM Developer Kit for Linux, Java Technology Edition, Версия 1.1.8
OS/2	IBM Java Development Kit for OS/2, версия 1.1.8 (она записана на компакт-диске продукта)
PTX	ptx/JSE, Версия 1.2.1, IBM
SGI IRIX	Java 2 Software Development Kit for SGI IRIX, версия 1.2.1, фирмы SGI

Solaris Operating Environment Java Development Kit for Solaris, версия 1.1.8, фирмы Sun Microsystems

32–битные операционные системы Windows

IBM Developer Kit для 32–битных операционных систем Windows, Java Technology Edition, Версия 1.1.8. При установке клиента разработки программ DB2 этот JDK автоматически устанавливается в каталог `sqllib\java\jdk`.

Информацию об установке и конфигурировании каких–либо из указанных выше JDK смотрите на странице Web по адресу:
<http://www.ibm.com/software/data/db2/java>.

Для всех поддерживаемых платформ необходимо также установить и сконфигурировать клиент DB2 с компонентом DB2 Java Enablement. Для связывания с базой данных программ SQLJ необходимо установить и сконфигурировать клиент администратора DB2 с компонентом DB2 Java Enablement.

Для выполнения написанных на языке Java хранимых процедур или пользовательских процедур DB2 необходимо также изменить конфигурацию менеджера баз данных DB2, указав в ней каталог, в котором на компьютере разработчика установлен JDK версии 1.1.

Можно проверить, задано ли в конфигурации менеджера баз данных DB2 правильное значение поля `JDK11_PATH`, введя следующую команду:

```
db2 get dbm cfg
```

Чтобы упростить просмотр выходных данных этой команды, их можно направить в файл. Поле `JDK11_PATH` находится недалеко от начала выходных данных. Дополнительную информацию об этих командах смотрите в руководстве *Command Reference*.



В среде Solaris Operating Environment некоторые реализации виртуальных машин Java (JVM) не будут правильно работать в программах, выполняющихся в среде "setuid". Может возникнуть ошибка загрузки совместно используемой библиотеки `libjava.so`, содержащей интерпретатор Java. Чтобы обойти эту ситуацию, можно создать символические ссылки для всех требуемых JVM совместно используемых библиотек в `/usr/lib` с помощью команд, подобных следующей (в зависимости от того, где на данном компьютере установлена поддержка языка Java):

```
ln -s /opt/jdk1.1.3/lib/sparc/native_threads/*.so /usr/lib
```

Чтобы получить дополнительную информацию об этом и других возможных способах исправления ситуации, посетите страницу Web по адресу: <http://www.ibm.com/software/data/db2/java>.

Чтобы можно было выполнять программы Java, при установке DB2 в операционных системах OS/2 и Windows и при создании экземпляра на платформах UNIX автоматически обновляются следующие переменные среды.

На платформах UNIX:

- В переменную среды CLASSPATH включается "." и файл sqllib/java/db2java.zip.
- В AIX, Linux, PTX, Silicon Graphics IRIX и Solaris Operating Environment: в LD_LIBRARY_PATH включается каталог sqllib/lib.
- В HP-UX: в SHLIB_PATH включается каталог sqllib/lib.
- Только в Solaris Operating Environment: для THREADS_FLAG задается "native".

На платформах Windows и OS/2:

- В переменную среды CLASSPATH включается "." и файл %DB2PATH%\java\db2java.zip.

Чтобы можно было построить и выполнять программы SQLJ, автоматически обновляется переменная среды CLASSPATH; в ее значение включаются следующие файлы:

На платформах UNIX:

- sqllib/java/sqlj.zip (требуется для построения программ SQLJ)
- sqllib/java/runtime.zip (требуется для выполнения программ SQLJ)

На платформах Windows и OS/2:

- %DB2PATH%\java\sqlj.zip (требуется для построения программ SQLJ)
- %DB2PATH%\java\runtime.zip (требуется для выполнения программ SQLJ)

Прикладные программы Java

Прикладная программа запускается с рабочего стола или из командной строки, для чего используется команда, запускающая интерпретатор Java для этой программы:

```
java имя_программы
```

где имя_программы – имя этой программы.

Драйвер DB2 JDBC обрабатывает вызовы JDBC API от прикладной программы и использует клиент DB2 для передачи требования на сервер и получения результатов. Перед выполнением апплет SQLJ должен быть связан с базой данных.

Апплеты Java

Поскольку апплеты Java передаются через Web, на компьютере (сервера или клиента) необходимо установить сервер Web.

Чтобы выполнить апплет, убедитесь, что используемый файл .html правильно сконфигурирован. Запустите сервер апплетов JDBC на порту, заданном в этом файле .html. Например, если задано:

```
param name=port value='6789'
```

нужно ввести команду:

```
db2jstrt 6789
```

Необходимо убедиться, что рабочий каталог доступен для браузера Web. Если это не так, скопируйте файлы .class и .html для этого апплета в доступный для браузера каталог. Для апплетов SQLJ необходимо также скопировать файлы профилей .class и .ser.

Скопируйте файл sqllib/java/db2java.zip в тот же каталог, что и другие файлы. Для апплетов SQLJ в этот каталог нужно также скопировать файл sqllib/java/runtime.zip. Затем запустите на компьютере клиента браузер Web (который поддерживает JDK 1.1) и загрузите этот файл .html.

Когда апплет вызывает JDBC API для соединения с DB2, драйвер JDBC устанавливает отдельные связи с базой данных DB2 через сервер апплетов JDBC, находящийся на сервере DB2. Перед выполнением апплет SQLJ должен быть связан с базой данных.

Часть 2. Справочник и устранение неисправностей

Глава 6. Изменение каталогов базы данных

Для управления информацией о базах данных, с которыми соединяется DB2 Connect, используются следующие каталоги:

- *Каталог узла*, содержащий информацию о сетевом адресе и протоколе связи для каждого сервера баз данных хоста или AS/400, к которым обращается DB2 Connect.
- *Каталог DCS*, содержащий информацию, специфичную для баз данных хоста или AS/400.
- *Системный каталог баз данных*, содержащий информацию об имени, узле и аутентификации для каждой базы данных, к которой DB2 Connect получает доступ.

Примечания:

1. Перед внесением изменений в эти каталоги надо сконфигурировать связь на сервере баз данных хоста или AS/400 и на рабочих станциях. Дополнительную информацию смотрите в книге *Дополнение по установке и настройке*.
2. Обратите внимание, что в OS/2 и 32-битных средах Windows каталоги баз данных можно изменить с помощью Ассистента конфигурирования клиента DB2 Universal Database.

На всех остальных платформах каталоги базы данных следует изменять с помощью процессора командной строки DB2.

3. Пример синтаксиса команды приводится в разделе “Изменение каталогов” на стр. 86. Дополнительную информацию смотрите в руководстве *Command Reference*.
4. При использовании DCE эти каталоги нужно изменить для каждой базы данных сервера баз данных хоста или AS/400, с которыми вы соединяетесь, или записать эквивалентную информацию в глобальном каталоге DCE. Дополнительную информацию о DCE смотрите в разделе Приложение D, “Использование служб каталога DCE” на стр. 201 и руководстве *Administration Guide*. В этом разделе считается, что службы каталога DCE не используются.

Сбор информации

В разделе Приложение В, “Листок настройки каталогов” на стр. 195 перечислена информация, которую необходимо собрать. Полезно сделать копию этого рабочего листа и вписать в нее данные вашей системы.

Каталог узла

В каталоге узла можно задать следующую информацию:

Имя узла

Псевдоним для системы сервера баз данных хоста или AS/400, на которой находится удаленная база данных. Это имя задает пользователь. Его следует записать и в таблицу параметров каталога узла, и в таблицу параметров системного каталога баз данных.

Формат: 1–8 однобайтных алфавитно-цифровых символов, включая символы #, @, \$ и подчеркивание (_). Это имя не может начинаться с символов подчеркивания или #.

Протокол

Может быть APPC или TCP/IP.

Имя символического назначения

При задании узла APPC используйте имя символического назначения, которое было задано в таблице информации о стороне связи CPI (например, имя свойств символического назначения CPI-C при использовании Microsoft SNA Server). Это значение нужно получить у того, кто устанавливал и/или конфигурировал SNA. Имя символического назначения регистрозависимо (в случае несоответствия имен, заданных в верхнем и нижнем регистре, можно получить код возврата SQL1338).

Тип защиты

Тип защитной проверки, который будет использоваться. Для узлов APPC допустимы опции: SAME, PROGRAM и NONE. Для узлов TCP/IP – опция SECURITY SOCKS, задающая включение на узле защиты SOCKS; в этом случае переменные среды SOCKS_NS и SOCKS_SERVER обязательно должны быть установлены, чтобы защита SOCKS стала возможной. Дополнительную информацию смотрите в разделе Глава 10, “Защита” на стр. 113 и в справочнике *Command Reference*.

Имя удаленного хоста или IP-адрес TCP/IP

При задании узла TCP/IP это или удаленное имя хоста TCP/IP, или удаленный адрес TCP/IP. Если задается имя хоста, оно должно быть разрешено на рабочей станции DB2 Connect либо через таблицу поиска DNS, либо с помощью записи в локальном файле хостов TCP/IP.

Для удаленных хостов DB2 for OS/390 имя хоста выводится в сообщении DSNL004I (DOMAIN=имя_хоста) при запуске DDF.

Имя службы или номер порта TCP/IP

При задании узла TCP/IP это или имя службы TCP/IP, или номер порта. Его нужно задать для TCP/IP на удаленном хосте. Для DRDA в качестве номера порта по умолчанию зарегистрирован номер порта 446.

Для удаленных хостов DB2 for OS/390 номер порта задается в наборе данных начальной загрузки (BSDS) как PORT и также появляется в сообщении DSNL004I (TCPSPORT=номер_порта) при запуске DDF.

Примечание: Второй порт, используемый для операций ресинхронизации двухфазного принятия через соединения TCP/IP, назначается сервером. Например, набор данных начальной загрузки DB2 Universal Database for OS/390 назначает номер порта (RESPORT), который будет использоваться только для ресинхронизации входящих соединений с DB2 Universal Database for OS/390. Имя службы для него задавать не надо.

Каталог DCS

В каталоге DCS можно задать следующую информацию:

Имя базы данных

Задаваемый пользователем псевдоним для сервера баз данных хоста или AS/400. Используйте это имя и в таблице параметров каталога DCS, и в таблице параметров системного каталога баз данных.

Формат: 1–8 однобайтных алфавитно-цифровых символов, включая символы #, @, \$ и подчеркивание (_). Это имя не может начинаться с символов подчеркивания или #.

Имя базы данных назначения

База данных в системе сервера баз данных хоста или AS/400:

MVS/ESA

Подсистема DB2 Universal Database for OS/390, определяемая своим именем LOCATION NAME.

Чтобы узнать LOCATION NAME, зарегистрируйтесь в TSO и выполните при помощи доступных средств запросов следующий запрос SQL:

```
select current server from sysibm.sysdummy1
```

LOCATION NAME указывается также в наборе данных MVS/ESA Boot Strap Data Set (BSDS) и в сообщении DSNL004I (LOCATION=имя_положения), появляющемся при запуске DDF.

OS/390

Подсистема DB2 Universal Database for OS/390, определяемая своим именем LOCATION NAME.

Чтобы узнать LOCATION NAME, зарегистрируйтесь в TSO и выполните при помощи доступных средств запросов следующий запрос SQL:

```
select current server from sysibm.sysdummy1
```

LOCATION NAME указывается также в наборе данных Boot Strap Data Set (BSDS) и в сообщении DSNL004I (LOCATION=имя_положения), появляющемся при запуске DDF.

VSE или VM

Имя базы данных (DBNAME)

OS/400

Имя реляционной базы данных (RDBNAME)

Другие Для OS/2, Windows NT, Windows 2000 и систем на основе UNIX – алиас базы данных в каталоге баз данных.

Имя реквестера прикладных программ

Имя реквестера прикладных программ для передачи требований SQL на серверы программ DRDA. Реквестер прикладных программ обрабатывает требования для прикладной программы.

Формат: AR <имя_реквестера_программ>

По умолчанию используется реквестер прикладных программ DB2 Connect.

Строка параметров

При желании изменить значения по умолчанию задайте произвольное сочетание приведенных ниже параметров в заданном порядке. Строку параметров нельзя задать с помощью Ассистента конфигурирования клиента, а при использовании процессора командной строки строка параметров должна заключаться либо в одиночные кавычки (например в OS/2 или Windows NT), либо в двойные кавычки (например, в AIX):

файл отображения

Имя файла отображения SQLCODE для замены отображения по умолчанию SQLCODE. Чтобы отключить отображение SQLCODE, задайте **NOMAP**. Дополнительную информацию смотрите в разделе Глава 11, “Отображение SQLCODE” на стр. 123.

,D Это второй позиционный параметр. Если его задать, программа будет отсоединяться от базы данных хоста или базы данных сервера баз данных при получении одного из следующих SQLCODE:

SQL3000N
SQL3004N
SQL3005N
SQL30051N
SQL30053N
SQL3006N
SQL3007N
SQL30071N
SQL30072N
SQL30073N
SQL30074N
SQL3009N

Если не задать параметр отсоединения **,D**, отсоединение будет выполняться только при получении следующих SQLCODE:

SQL3002N
SQL30021N
SQL30041N
SQL30061N
SQL30081N

Объяснение этих кодов смотрите в книге *Справочник по сообщениям*.

Примечание: Если отсоединение DB2 Connect происходит из-за ошибки, автоматически выполняется откат.

„INTERRUPT_ENABLED

Это третий позиционный параметр. Если параметр INTERRUPT_ENABLED сконфигурирован в каталоге DCS на рабочей станции DB2 Connect и программа клиента выдаст прерывание во время соединения с хостом или сервером баз данных AS/400, DB2 Connect обработает это прерывание, разорвав соединение и выполнив откат единицы работы. Такая обработка прерывания поддерживается в AIX, OS/2, Windows NT и Windows 2000.

Программа получит SQLCODE (-30081), указывающий на разрыв соединения с сервером. Чтобы обработать дополнительные требования базы данных, программа после этого должна установить с этим хостом или сервером баз данных AS/400 новое соединение. На всех платформах, кроме AIX, Версия 4.1 или новее, SNA Server, Версия 3.1 или новее, OS/2, Windows NT и Windows 2000, DB2 Connect не поддерживает опцию

автоматического отсоединения, когда использующая ее программа получает требование на прерывание.

Примечание: Для TCP/IP эта поддержка работает на всех платформах. Клиент может удалить гнездо, но, в зависимости от реализации сервера, при этом прием может не произойти. DB2 Universal Database for OS/390 использует асинхронные вызовы гнезд и поэтому способна обнаружить потерю соединения и выполнить откат долго выполняющихся операторов SQL, которые находятся в обработке.

,,,,,SYSPLEX

Этот шестой позиционный параметр можно использовать для явного включения поддержки SYSPLEX в DB2 Connect для отдельной базы данных.

Введена новая переменная профиля (среды или реестра) DB2SYSPLEX_SERVER, которую можно использовать для отключения поддержки SYSPLEX на уровне рабочей станции.

,,,,,LOCALDATE="*<значение>*"

Этот седьмой позиционный параметр используется для поддержки формата даты DB2 Connect. Поддержка осуществляется с помощью маски даты для *<значения>* следующим образом:

Предположим, вы ввели следующие операторы командной строки:

```
catalog appc node nynode remote nycpic security program
catalog dcs database nydb1 as new_york
catalog database nydb1 as newyork1 at node nynode
authentication dcs
```

Алиас базы данных *newyork1* будет использоваться для доступа к базе данных хоста без преобразования даты, так как маска даты не была задана.

Однако с новой возможностью форматирования даты можно воспользоваться командами процессора командной строки, показанными ниже. В этом случае, поскольку используется процессор командной строки, а строка параметров сама задается в двойных кавычек, значение LOCALDATE нужно заключить в две пары двойных кавычек. Обратите внимание на символ "\"

(обратная косая черта), который используется для того, чтобы двойные кавычки не удалялись из спецификации LOCALDATE. Смотрите также раздел “Задание строки параметров” на стр. 84.

```
catalog dcs database nydb2 as new_york
  parms "\",,,,,,LOCALDATE=\"\"YYYYMMDD\"\"\"
catalog database nydb2 as newyork2 at node nynode
  authentication dcs
```

Алиас базы данных "newyork2" используется для доступа к той же базе данных хоста, но у него задана маска формата даты. Этот пример показывает, что маска формата даты задается с использованием ключевого слова LOCALDATE в седьмом позиционном параметре поля PARMs записи каталога DCS.

Чтобы маска даты была допустимой, должны быть выполнены ВСЕ следующие условия:

1. В маске должно быть не более одной последовательности символов Y, символов M и символов D (где Y – цифры года, M – цифры месяца, а D – цифры числа).
2. Максимальное число символов Y в последовательности – 4.
3. Максимальное число символов M в последовательности – 2.
4. Максимальное число символов D в последовательности – 2.

Например, следующие маски дат допустимы:

"YYyyMmDd" – (Y, M и D можно задавать в любом регистре)
"MM+DD+YYYY" – Маска может быть длиннее 10 байт
и содержать символы, отличные
от Y, M и D
"abcYY+MM" – Последовательность D может отсутствовать.

Следующие маски дат недопустимы:

"YYYYyMMDD" – недопустима, в последовательности 5 Y
"YYYYMDDM" – недопустима, есть 2 последовательности M

Если маска даты недопустима, сообщение об ошибке не выводится. Такая маска просто будет проигнорирована. Но если маска даты допустима, это еще не значит, что она будет использоваться. Преобразование формата даты по допустимой маске производится, только если выполнены ВСЕ следующие условия:

1. Нет ошибок SQL.
2. Вывод – значение даты в формате типа ISO (то есть ISO или JIS).
3. Длина области вывода данных – по крайней мере 10 байт. Это минимальный размер области выходных данных для хранения значения даты, даже если преобразование формата данных НЕ производится. Это требование применяется даже в тех случаях, когда маска формата даты короче 10 байт.
4. В записи каталога DCS есть допустимая маска формата даты, по длине не превышающая область вывода данных.

,,,,,,CHGPWD_SDN=<имя>

Этот восьмой позиционный параметр служит для задания имени символического назначения, используемого для управления сроком действия пароля (PEM). Значение <имя> регистрозависимо.

В разделе “Изменение пароля MVS” на стр. 118 приводится пример каталогизации каталога базы данных с использованием CHGPWD_SDN:

```
catalog dcs database db1 as dsn_db_1 parms
    " , , , , , , , CHGPWD_SDN=pempgm "
```

,,,,,,BIDI=<ccsid>

Этот девятый позиционный параметр служит для задания (BiDi) CCSID (coded character set identifier – идентификатор кодового набора символов) с двумя направлениями письма, который переопределяет заданный по умолчанию CCDID с двумя направлениями письма базы данных сервера. Например:

```
" , , , , , , , BIDI=xyz "
```

где xyz заменяет значение CCSID (смотрите примечание 1 на стр. 84).

Список поддерживаемых CCSID с двумя направлениями письма и их типы строк смотрите в руководстве *Administration Guide*.

Для правильной обработки данных с двумя направлениями письма на разных платформах требуются следующие атрибуты BiDi:

- Начертание цифр (ARABIC либо HINDI)
- Направление (RIGHT-TO-LEFT либо LEFT-TO-RIGHT)
- Изменение формы (SHAPED либо UNSHAPED)

- Симметричное обращение (YES или NO)
- Тип текста (LOGICAL либо VISUAL)

Поскольку установки по умолчанию на разных платформах не одинаковы, существует проблема пересылки данных DB2 с одной платформы на другую. Например, на платформах Windows используются данные в формате LOGICAL UNSHAPED, а данные на MVS и OS/390 обычно имеют формат SHAPED VISUAL. Следовательно, без поддержки атрибутов BiDi данные, переданные от DB2 for MVS или от DB2 for OS/390 на DB2 Connect, в Windows будут выведены неправильно.

Когда DB2 Connect и база данных на сервере обмениваются данными, преобразование данных обычно выполняется принимающей стороной. Как правило, это же соглашение применимо и для преобразования с двумя направлениями письма в дополнение к обычному преобразованию кодовой страницы. Однако в настоящее время поддержка CCSID с двумя направлениями письма и преобразования вида текста с двумя направлениями письма никакими продуктами DB2 хоста не выполняется. Поэтому в DB2 Connect была добавлена необязательная возможность выполнять преобразование текста с двумя направлениями письма для данных, подготавливаемых к передаче в базу данных сервера помимо преобразования данных, принимаемых от базы данных сервера.

Для выполнения в DB2 Connect преобразования текста с двумя направлениями письма для данных, передаваемых в базу данных сервера, нужно заменить CCSID с двумя направлениями письма базы данных сервера (смотрите раздел 2 на стр. 84). Это выполняется с помощью параметра BIDI в поле PARMS записи каталога базы данных DCS для базы данных сервера.

Использование этой функции лучше всего показать на примере.

Рассмотрим клиент DB2 с поддержкой иврита; его CCSID 62213 (строки с двумя направлениями письма типа 5); требуется доступ к базе данных хоста DB2, который использует CCSID 424 (строки с двумя направлениями письма типа 4). Но при этом известно, что в самих данных в базе хоста DB2 используется CCSID 8616 (строки с двумя направлениями письма типа 6).

В этой ситуации существует две проблемы. Во-первых, базе данных хоста не известно различие между типами строк CCSID 424 и 8616. Во-вторых, база данных хоста DB2 не распознает CCSID 62213 клиента DB2. Она поддерживает только CCSID 862, который использует ту же кодовую страницу, что и CCSID 62213.

Сначала нужно убедиться, что данные, передаваемые на базу данных хоста DB2, имеют формат строки с двумя направлениями письма типа 6, а также сообщить DB2 Connect, что надо выполнять преобразование вида строк с двумя направлениями письма, принимаемых от базы данных хоста DB2. Выполните для базы данных хоста следующую команду занесения в каталог:

```
catalog dcs database nydb1 as TELAVIV parms ",,,,,,,BIDI=8616"
```

Для DB2 Connect это указание заменить CCSID 424 базы данных хоста DB2 на CCSID 8616. Эта замена заключается в следующем:

1. DB2 Connect соединится с базой данных хоста DB2, используя CCSID 862.
2. DB2 Connect будет выполнять преобразование строк с двумя направлениями письма для данных, подготавливаемых к передаче на базу данных хоста DB2, заменяя CCSID 62213 (строки с двумя направлениями типа 5) на CCSID 62221 (строки с двумя направлениями типа 6).
3. DB2 Connect будет выполнять преобразование строк с двумя направлениями для данных, получаемых от базы данных хоста DB2, заменяя CCSID 8616 (строки с двумя направлениями письма типа 6) на CCSID 62213 (строки с двумя направлениями письма типа 5).

Примечания:

1. Чтобы параметр BIDI вступил в силу, для переменной среды или реестра DB2BIDI нужно установить значение YES.
2. Если от DB2 Connect требуется выполнять преобразование для данных перед передачей на базу данных хоста DB2, даже без замены их CCSID, в поле PARMS каталога базы данных DCS все равно нужно добавить параметр BIDI. В этом случае требуемое значение CCSID будет значением по умолчанию CCSID базы данных хоста.
3. В некоторых случаях использование CCSID с двумя направлениями письма может привести к изменению самого запроса SQL так, что он может быть не опознан сервером DB2. Поэтому следует избегать использовать CCSID IMPLICIT CONTEXTUAL и IMPLICIT RIGHT-TO-LEFT, когда могут быть использованы разные типы строк. CCSID CONTEXTUAL могут привести к непредсказуемым результатам, если запрос SQL содержит строки в кавычках. Избегайте использовать в операторах SQL строки в кавычках; вместо этого используйте, где возможно, переменные хоста.

Если использование некоторого CCSID с двумя направлениями письма приводит к возникновению ошибок, которые нельзя исправить, следуя этим рекомендациям, установите для переменной среды или реестра DB2BIDI значение NO.

Задание строки параметров

Ниже приведены примеры некоторых допустимых строк параметров.

Например, можно задать любую из следующих строк, где "\" (обратная косая черта) – символ исключения операционной системы:

В системе AIX:

```

NOMAP
/u/username/sql1lib/map/dcs1new.map,D
,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE="\\"YYMMDD\\"",,

```

В OS/2, Windows NT или Windows 2000:

```

NOMAP
d:\sql1lib\map\dcs1new.map,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE="\\"YYMMDD\\"",,

```

Если не задавать строку параметров, будут использоваться значения по умолчанию.

Примечание: Так как при задании в строке параметров маски LOCALDATE необходимы две пары двойных кавычек, требуется использовать символ исключения операционной системы "\"" (обратная косая черта), например:

```
db2 catalog dcs db x as y parms \",,,,,,LOCALDATE=\\\"YYMMDD\\\"\",
```

В результате запись каталога DCS будет выглядеть так:

DCS 1 entry:

Local database name	= X
Target database name	= Y
Application requestor name	=
DCS parameters	= ,,,,,,LOCALDATE=\\\"YYMMDD\\\"
Comment	=
DCS directory release level	= 0x0100

Системный каталог баз данных

В системном каталоге баз данных можно задать следующую информацию:

Имя базы данных

То же имя, которое вы занесли в таблицу параметров каталога DCS.

Алиас базы данных

Алиас для сервера баз данных AS/400. Это имя будет использоваться программой, получающей доступ к базе данных. По умолчанию используется значение, которое задается для имени базы данных.

Формат: 1–8 однобайтных алфавитно-цифровых символов, включая символы #, @, \$ и подчеркивание (_). Это имя не может начинаться с символов подчеркивания или #.

Имя узла

То же имя, которое вы занесли в таблицу параметров каталога узла.

Аутентификация

Этот параметр задает, где будет проверяться имя пользователя и пароль. Допустимы следующие опции: SERVER, SERVER_ENCRYPT, CLIENT, DCE, DCS и DCS_ENCRYPT. Дополнительную информацию смотрите в разделе Глава 10, “Защита” на стр. 113.

Задание нескольких записей для одной базы данных

Для каждой базы данных надо задать по крайней мере одну запись в каждом из трех каталогов (каталоге узла, каталоге DCS и каталоге базы данных). Иногда может потребоваться задать для какой-нибудь базы данных несколько записей.

Например, можно отключить отображение SQLCODE для программ, которые были перенесены с сервера баз данных AS/400, но принять отображение по умолчанию для программ, разработанных для среды клиент-сервер. Это можно сделать следующим образом:

- Задайте одну запись в каталоге узла.
- Задайте две записи в каталоге DCS с разными именами базы данных. Для одной записи в строке параметров задайте NOMAP.
- Задайте две записи в системном каталоге баз данных с разными алиасами базы данных и двумя именами базы данных, которые вы задали в каталоге DCS.

Оба алиаса предназначены для доступа к одной базе данных, один – с отображением SQLCODE, а другой – без отображения SQLCODE.

Изменение каталогов

В любой системе DB2 Connect можно использовать команду CATALOG, а в OS/2 и в 32-битных средах Windows, кроме того – мастер по добавлению баз данных в CCA. Если установлен Клиент разработки программ DB2, можно также написать программу для каталогизации записей. Информацию об API смотрите в справочниках *Administrative API Reference* и *Command Reference*.

Примечание: Для каталогизации базы данных у вас должны быть права *системного администратора*.

Чтобы изменить каталоги с помощью процессора командной строки, сделайте следующее:

1. Для изменения каталога узла воспользуйтесь одной из следующих команд:

- Для узла, использующего соединение APPC:
db2 CATALOG APPC NODE *имя_узла*
REMOTE *имя_символического_назначения* SECURITY *тип_защиты*

Например:


```
db2 CATALOG APPC NODE DB2NODE REMOTE DB2CPIC SECURITY PROGRAM
```

- Для базы данных DB2 Universal Database for OS/390, Версия 5.1 или DB2 Universal Database for AS/400, Версия 4.2, использующей соединение TCP/IP:

```
db2 CATALOG TCPIP NODE имя_узла  
REMOTE имя_хоста или IP_адрес  
SERVER имя_службы или номер_порта  
SECURITY тип_защиты
```

Например:

```
db2 CATALOG TCPIP NODE MVSIPNOD REMOTE MVSHOST SERVER DB2INSTC
```

Для соединений TCP/IP номер порта DRDA по умолчанию 446.

2. Для изменения каталога DCS воспользуйтесь следующей командой:

```
db2 CATALOG DCS DATABASE имя_базы_данных AS имя_базы_данных_назначения  
[AR реквестер_программ]  
[PARMS "строка_параметров"]
```

Например:

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3
```

или в OS/2, Windows NT или Windows 2000:

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3 PARMS "NOMAP,D"
```

или (для AIX):

```
db2 CATALOG DCS DATABASE DB2DB AS NEW_YORK3 PARMS "'NOMAP,D'"
```

Примечание: Информацию об использовании символа исключения операционной системы при задании в строке параметров маски LOCALDATE смотрите в разделе “Задание строки параметров” на стр. 84.

3. Для изменения системного каталога баз данных воспользуйтесь следующей командой:

```
db2 CATALOG DATABASE имя_базы_данных AS алиас AT NODE имя_узла  
AUTHENTICATION тип_аутентификации
```

Например:

```
db2 CATALOG DATABASE DB2DB AS NYC3 AT NODE DB2NODE AUTHENTICATION DCS
```

Если есть удаленные клиенты, на каждом из них также требуется изменить каталоги. Дополнительную информацию смотрите в соответствующей книге *DB2 Connect Quick Beginnings* (Быстрый старт).

Глава 7. Связывание прикладных программ и утилит

Прикладные программы, использующие встроенный SQL, должны быть связаны с каждой базой данных, с которой они будут работать. Это можно сделать при помощи командного центра или ассистента конфигурирования клиента (на тех платформах, где они доступны).

Связывание должно быть проведено один раз для каждой программы и базы данных. При связывании для каждого оператора SQL, который будет выполнен, записываются схемы доступа к базе данных. Эти схемы доступа поставляются разработчиками программы и хранятся в *файлах связывания*, которые создаются при прекомпиляции. Связывание состоит в обработке этих файлов связывания сервером баз данных хоста или AS/400. Дополнительную информацию о связывании смотрите в руководстве *Application Development Guide*.

Поскольку некоторые утилиты, поставляемые вместе с DB2 Connect, разработаны с использованием встроенного SQL, они должны быть связаны с сервером баз данных хоста или AS/400, чтобы их можно было использовать с этой системой. Если вы не используете утилиты и интерфейсы DB2 Connect, перечисленные в Табл. 4 на стр. 92, вам не нужно связывать их с каждым вашим сервером баз данных хоста или AS/400. Списки файлов связывания, необходимых для этих утилит, содержатся в следующих файлах:

ddcsmvs.lst

Для MVS или OS/390

ddcsvse.lst

Для VSE

ddcsvm.lst

Для VM

ddcs400.lst

Для OS/400

Связывание одного из этих файлов с базой данных свяжет каждую отдельную утилиту с этой базой данных.

Если установлена DB2 Connect Enterprise Edition, утилиты DB2 Connect должны быть связаны с каждым сервером баз данных хоста или AS/400; это надо сделать один раз с каждого типа платформы клиента, чтобы их можно было использовать с этой системой.

Например, если у вас 10 клиентов OS/2, 10 клиентов Windows и 10 клиентов AIX, подключенных к DB2 Universal Database for OS/390 через сервер DB2 Connect Enterprise Edition под Windows NT, поступайте следующим образом:

1. Свяжите ddcsmvs.lst с одного из клиентов Windows.
2. Свяжите ddcsmvs.lst с одного из клиентов OS/2.
3. Свяжите ddcsmvs.lst с одного из клиентов AIX.
4. Свяжите ddcsmvs.lst с сервера DB2 Connect.

Примечание: При этом подразумевается, что все клиенты имеют один уровень обслуживания. Если это не так, может дополнительно понадобится выполнить связывание с клиента каждого уровня обслуживания. Если у вас есть клиенты DB2 ранних версий (до 2.1), посмотрите раздел Приложение E, “Связывание утилит для клиентов ранних версий” на стр. 213.

Кроме утилит DB2 Connect, надо связать с каждой используемой базой данных все остальные программы, использующие встроенный SQL. При запуске несвязанной программы обычно выводится сообщение об ошибке SQL0805N. Возможно, вы захотите создать дополнительный файл списка связывания для ваших программ, которые нужно связать.

Для каждого сервера баз данных хоста или AS/400, с которым вы выполняете связывание, сделайте следующее:

1. Проверьте, есть ли у вас достаточные полномочия доступа в системе управления сервером баз данных хоста или AS/400:

MVS или OS/390

Необходимые полномочия:

- SYSADM или
- SYSCTRL или
- BINDADD и CREATE IN COLLECTION NULLID

Примечание: Привилегии BINDADD и CREATE IN COLLECTION NULLID дают достаточные полномочия **только** в том случае, когда пакеты еще не существуют — например, если вы создаете их в первый раз.

Если пакеты уже существуют и вы связываете их заново, полномочия, необходимые для выполнения заданий, зависят от того, кто выполнял первоначальное связывание.

A Если вы создавали первоначальное связывание и делаете связывание снова, любые из перечисленных полномочий позволят вам выполнить связывание.

B Если первоначальное связывание выполнял кто-то еще, а повторное связывание выполняете вы, вам понадобятся полномочия SYSADM или SYSCTRL.

Полномочий BINDADD и CREATE IN COLLECTION NULLID в этом случае недостаточно. Но и не имея привилегий SYSADM или SYSCTRL, можно создать пакет. В таком случае вам понадобятся привилегии BIND для каждого из существующих пакетов, которые вы намереваетесь заменить.

VSE или VM

Необходимы полномочия DBA. Если вы хотите использовать опцию GRANT в команде bind (чтобы избежать индивидуального предоставления доступа к каждому пакету DB2 Connect), пользовательский идентификатор NULLID должен иметь право предоставлять права другим пользователям для следующих таблиц:

- system.syscatalog
- system.syscolumns
- system.sysindexes
- system.systabauth
- system.syskeycols
- system.syssynonyms
- system.syskeys
- system.syscolauth

В системе VSE или VM вы можете ввести:

```
grant select on таблица to nullid with grant option
```

OS/400

Полномочия *CHANGE или выше для набора NULLID.

2. Введите команды, аналогичные приведенным ниже:

```
db2 connect to алиас_базы_данных user ID_пользователя using пароль
db2 bind путь@ddcsmvs.lst blocking all
      sqlerror continue messages ddcsmvs.msg grant public
db2 connect reset
```

Где *алиас_базы_данных*, *ID_пользователя* и *пароль* относятся к серверу баз данных хоста или AS/400, ddcsmvs.lst – это файл списка связывания MVS, а *путь* – это каталог, где находится файл списка связывания.

Например, *диск*: \sql11b\bnd\ для операционных систем платформ Intel, или *НАЧАЛЬНЫЙ_КАТАЛОГ*/sql11b/bnd/ для операционных систем UNIX, где *диск* – это логический диск, на котором установлена DB2 Connect, а *НАЧАЛЬНЫЙ_КАТАЛОГ* – это начальный каталог вашей копии DB2 Connect.

При помощи опции **grant** команды **bind** можно дать права EXECUTE всем пользователям (PUBLIC) или задать идентификатор пользователя или ID группы. Если вы не используете опцию **grant** команды **bind**, надо давать полномочия на выполнение – GRANT EXECUTE (RUN) – индивидуально.

Чтобы получить имена пакетов для файлов связывания, введите следующую команду:

```
ddcspkgn @bindfile.lst
```

Например, результат выполнения команды

```
ddcspkgn @ddcsmvs.lst
```

может быть таким:

Bind File	Package Name
f:\sqllib\bnd\db2ajgrt.bnd	SQLAB6D3

Для справки в Табл. 4 показаны файлы связывания и имена пакетов, которые используются различными компонентами DB2 Connect. В некоторых случаях для различных операционных систем используются различные файлы связывания и пакеты.

Таблица 4 (Стр. 1 из 2). Файлы связывания и пакеты

Компонент	Файл связывания	Пакет	MVS или OS/390	VSE	VM	OS/400
Утилита связывания (используется опцией связывания GRANT)	db2ajgrt.bnd	sqlabxxx	да	да	да	да
Интерфейс уровня вызовов DB2						
Уровень изоляции CS	db2clics.bnd	sql1xxx	да	да	да	да
Уровень изоляции RR	db2clirr.bnd	sql2xxx	да	да	да	да
Уровень изоляции UR	db2cliur.bnd	sql3xxx	да	да	да	да
Уровень изоляции RS	db2clirs.bnd	sql4xxx	да	да	да	да
Уровень изоляции NC	db2clinc.bnd	sql5xxx	нет	нет	нет	да
Использование имен таблиц MVS	db2clims.bnd	sql7xxx	да	нет	нет	нет
Использование имен таблиц OS/400 (OS/400 3.1 или более новой)	db2clias.bnd	sql1axxx	нет	нет	нет	да
Использование имен таблиц VSE/VM	db2clivm.bnd	sql8xxx	нет	да	да	нет
Процессор командной строки						
Уровень изоляции CS	db2clpcs.bnd	sqlc2xxx	да	да	да	да
Уровень изоляции RR	db2clprr.bnd	sqlc3xxx	да	да	да	да

Таблица 4 (Стр. 2 из 2). Файлы связывания и пакеты						
Компонент	Файл связывания	Пакет	MVS или OS/390	VSE	VM	OS/400
Уровень изоляции UR	db2clpur.bnd	sqlc4xxx	да	да	да	да
Уровень изоляции RS	db2clprs.bnd	sqlc5xxx	да	да	да	да
Уровень изоляции NC	db2clpnc.bnd	sqlc6xxx	нет	нет	нет	да
REXX						
Уровень изоляции CS	db2arxcs.bnd	sqla1xxx	да	да	да	да
Уровень изоляции RR	db2arxrr.bnd	sqla2xxx	да	да	да	да
Уровень изоляции UR	db2arxur.bnd	sqla3xxx	да	да	да	да
Уровень изоляции RS	db2arxrs.bnd	sqla4xxx	да	да	да	да
Уровень изоляции NC	db2arxnc.bnd	sqla5xxx	нет	нет	нет	да
Утилиты						
Экспорт	db2uexpm.bnd	sqlubxxx	да	да	да	да
Импорт	db2uimpb.bnd	sqlufxxx	да	да	да	да

Чтобы определить эти значения для DB2 Connect, запустите утилиту *ddcspkgn*, например:

```
ddcspkgn @ddcsmvs.lst
```

Эту утилиту можно использовать также для определения имени пакета отдельного файла связывания, например:

```
ddcspkgn bindfile.bnd
```

Если на вашей системе DB2 for MVS/ESA установлены исправления APAR PN60988 (или ваш выпуск новее, чем Версия 3 Выпуск 1), можно также добавлять файлы связывания для уровня изоляции NC в файл *ddcsmvs.lst*.

Дополнительную информацию об опциях команды *bind* смотрите в книге *Command Reference*.

Примечания:

- a. Использование опции связывания `sqlerror continue` обязательно, однако, если вы связываете программы при помощи средств DB2 или процессора командной строки, эта опция задается автоматически. Установка этой опции превращает ошибки связывания в предупреждения, таким образом, при связывании файла с ошибками пакет создается, несмотря на эти ошибки. Это позволяет использовать один файл связывания для нескольких серверов, даже если какая-то отдельная реализация сервера отмечает синтаксис SQL другой реализации как неправильный. В частности, при связывании любых

файлов списков `ddcsxxx.lst` для любого конкретного сервера баз данных хоста или AS/400 можно ожидать появления предупреждений. Например, при связывании для DB2 for VM может появиться множество предупреждающих сообщений, так как DB2 for VM не разрешает объявлять указатели с условием "WITH HOLD".

- b. Если вы подключаетесь к базе данных DB2 Universal Database через DB2 Connect, используйте список связывания `db2ubind.lst` и не указывайте опцию `sqlerror continue`, которая действует только при подключении к серверу баз данных хоста или AS/400. Мы рекомендуем также при подключении к базе данных DB2 Universal Database использовать клиенты DB2, поставляемые вместе с DB2, а не DB2 Connect.
3. Используйте одинаковые операторы для связывания каждой программы или списка программ.
4. Если у вас есть удаленные клиенты из предыдущего выпуска DB2, вам может понадобиться связать утилиты этих клиентов с DB2 Connect. Дополнительную информацию смотрите в разделе Приложение E, "Связывание утилит для клиентов ранних версий" на стр. 213.

Команда BIND

Команда **bind** DB2 связывает программу с конкретной базой данных. Если вы выполняете операции прекомпиляции и связывания по отдельности, опции, указанные при связывании, переопределяют опции, указанные на шаге прекомпиляции.

В справочнике *Command Reference* описан синтаксис команды BIND, который используется при связывании программы с сервером баз данных хоста или AS/400 через DB2 Connect. Убедитесь, что вы пользуетесь описанием именно для DRDA.

Примечание: Некоторые параметры команды BIND могут не поддерживаться вашим сервером баз данных хоста или AS/400. Дополнительную информацию смотрите в документации, поставляемой вместе с RDBMS вашего сервера баз данных хоста или AS/400.

Повторное связывание

Связав свою программу (и создав пакет на сервере баз данных хоста или AS/400), вы можете обнаружить, что необходимо создать пакет заново. Это можно сделать без первоначального файла связывания, используя команду **REBIND PACKAGE** командной строки или соответствующий API.

Преимущества использования этой команды:

- Можно улучшить свою систему, переоптимизировав ее и перестроив новые части пакета без первоначального файла связывания.
- Можно пересоздать пакеты, ставшие негодными для использования или неправильными.
- Можно пересоздать пакеты, которые стали непригодными из-за перенастройки.
- Можно увеличить производительность, используя явное пересвязывание, а не неявное пересвязывание или связывание.
- Можно изменить характеристики пакета. Например, в DB2 Universal Database for OS/390 можно изменить спецификатор неспецифицированных таблиц для тестирования или для перенастройки.

Если вы хотите изменить программу, опции связывания или любую другую информацию, надо использовать команду **BIND**. Если пакета не существует в базе данных или если вы хотите увидеть все ошибки связывания (а не только первую обнаруженную ошибку), также надо использовать команду **BIND**.

Чтобы запустить эту команду, вам нужен уровень полномочий, требуемый вашим сервером баз данных хоста или AS/400. Если вы не соединены с базой данных, команда выполнит неявное соединение с базой данных по умолчанию (если у вас есть привилегии на это соединение).

Синтаксис этой команды описан в справочнике *Command Reference*.

Чтобы определить имя пакета, соответствующего файлу связывания, введите следующую команду: **ddcspkgn bindfile.bnd**

Глава 8. Монитор системы баз данных

В этой главе приводится сводка возможностей монитора системы DB2 для пользователей DB2 Connect. В DB2 Connect Версии 7 введены следующие улучшения монитора системы баз данных:

- Монитор снимков. Моментальные снимки системы дают информацию о ее состоянии на определенный момент времени. Снимок – это картина текущего состояния действий менеджера баз данных с определенным объектом или группой объектов. Монитор позволяет получить пять основных видов снимков информации баз данных DCS.
- Графический интерфейс пользователя, эквивалентный команде LIST DCS APPLICATIONS CLP. Команда LIST дает более точную картину состояния системы по сравнению с этим снимком. Графический вариант команды LIST вызывается через командный центр DB2. Смотрите раздел “Команда LIST DCS APPLICATIONS EXTENDED” на стр. 105

Дополнительную информацию о мониторе баз данных смотрите в руководстве *System Monitor Guide and Reference*.

Слежение за соединениями с удаленными клиентами

При помощи монитора системы баз данных и DB2 Connect Enterprise Edition можно следить за соединениями с удаленными клиентами. Чтобы следить за клиентами, локальными для сервера DB2 Connect, которые работают на самой системе, надо задать следующую переменную среды:

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

Например, когда ошибка возникает на системе хоста или AS/400, системный администратор может определить, была ли причиной проблема на рабочей станции DB2 Connect. Монитор системы баз данных дает следующую информацию:

- Маркер соответствия DRDA (CRRTKN) для незащищенных диалогов
- Идентификатор логической единицы работы (LUWID) для двухфазных диалогов, если для защиты используется менеджер точек синхронизации SNA (SPM)
- Идентификатор единицы работы (UOWID) для двухфазных соединений, если для защиты используется менеджер точек синхронизации DRDA-3 (как в соединениях TCP/IP)
- Идентификатор соединения DB2 Connect (ID программы)

В этой информации показывается, какое из соединений DB2 Connect создало проблему, что позволяет администратору системы отключить одну программу клиента, не мешая другим клиентам использовать соединение DB2 Connect.

Переключатели монитора для DB2 Connect

Монитор системы всегда активен. Но, если вы хотите получить более подробный результат от команды GET SNAPSHOT, нужно использовать те или иные переключатели монитора. К DB2 Connect имеют отношение два переключателя – STATEMENT (для информации уровня операторов языка) и UOW (для информации уровня транзакций).

Для управления переключателями монитора используйте команду **db2 update monitor switches**. Синтаксис этой команды смотрите в руководстве *Command Reference*. В следующем примере выводится статистика монитора системы DB2 для единиц работы (UOW):

```
db2 update monitor switches using uow on
```

Вывод состояния переключателей монитора

Чтобы вывести состояние переключателей монитора, используйте команду **db2 get monitor switches**.

Использование команд GET SNAPSHOT

Монитор DB2 следит за важными для системы показателями. В любой момент можно получить сводку состояния системы, введя команду GET SNAPSHOT. Получить снимок монитора можно, если у вас есть полномочия SYSMAINT, SYSCTRL или SYSADM для экземпляра менеджера баз данных, для которого вам нужен снимок.

Для слежения за информацией DCS полезны пять команд снимков:

- GET SNAPSHOT FOR ALL DCS DATABASES
- GET SNAPSHOT FOR ALL DCS APPLICATIONS
- GET SNAPSHOT FOR DCS APPLICATION ...
- GET SNAPSHOT FOR DCS DATABASE ON алиас_бд
- GET SNAPSHOT FOR DCS APPLICATIONS ON алиас_бд

Каждая из этих команд генерирует подробный отчет о указанной в ней области.

Например, команда GET SNAPSHOT FOR DCS DATABASE ON DCSDB порождает следующий отчет:

Снимок базы данных DCS

Имя базы данных DCS	= DCSDB
Имя базы данных хоста	= GILROY
Временная отметка первого соединения с базой	= 12-15-1999 10:28:24.596495
Текущее время соединения	= 0.950561
Текущая продолжительность соединения	= 0.000000
Время ответа хоста (сек.мс)	= 0.000000
Временная отметка последнего сброса	=
Число попыток для операторов SQL	= 2
Попыток принятия операторов	= 1
Попыток отката операторов	= 0
Операторов, обработанных неудачно	= 0
Общее число соединений шлюза	= 1
Текущее число соединений шлюза	= 1
Соединений шлюза ждет ответа хоста	= 0
Соединений шлюза ждет требования клиента	= 1
Ошибок связи шлюза с хостом	= 0
Отметка времени последней ошибки связи	= Нет
Высший уровень соединений шлюза	= 1
Выбрано строк	= 0
Отправлено исходящих байтов	= 140
Принято исходящих байтов	= 103

Этот отчет дает информацию о соединениях базы данных, производительности, числе ошибок и пропускной способности обработки требований SQL. Можно делать и более детальные снимки монитора DB2. Например, если вы введете команду GET SNAPSHOT FOR ALL DCS APPLICATIONS, вы получите примерно следующий снимок:

Снимок программы DCS

ID программы клиента	= 09150F74.B6A4.991215152824
Последовательный номер	= 0001
ID авторизации	= SMITH
Имя прикладной программы	= db2bp
Хэндл прикладной программы	= 1
Состояние программы	= ожидание требования
Время изменения состояния	= 12-15-1999 10:29:06.707086
Узел клиента	= sys143
Уровень выпуска клиента	= SQL06010
Платформа клиента	= AIX
Протокол клиента	= TCP/IP
Кодовая страница клиента	= 850
ID процесса программы клиента	= 49074
ID регистрации клиента	= smith
ID программы хоста	= G9150F74.B6A5.991215152825
Последовательный номер	= 0000
Алиас базы данных на шлюзе	= MVSDDB
Имя базы данных DCS	= DCSDDB
Имя базы данных хоста	= GILROY
Уровень версии хоста	= DSN05012
CCSID хоста	= 500
Адрес исходящих связей	= 9.21.21.92 5021
Протокол исходящих связей	= TCP/IP
Адрес входящих связей	= 9.21.15.116 46756
Временная отметка первого соединения с базой	= 12-15-1999 10:28:24.596495
Время ответа хоста (сек.мс)	= 0.000000
Время на обработку на шлюзе	= 0.000000
Временная отметка последнего сброса	=
Выбрано строк	= 0
Число попыток для операторов SQL	= 2
Операторов, обработанных неудачно	= 0
Операторов принятия	= 1
Операторов отката	= 0
Принято входящих байтов	= 404
Отправлено исходящих байтов	= 140
Принято исходящих байтов	= 103
Отправлено входящих байтов	= 287
Число открытых указателей	= 0
Время простоя программы	= 1 минута и 32 секунды
Статус завершения единицы работы	=
Врем. отметка заверш. пред. единицы работы	= 12-15-1999 10:28:25.592631
Временная отметка запуска единицы работы	= 12-15-1999 10:29:06.142790
Временная отметка конца единицы работы	=
Время вып.посл.заверш.единицы работы (сек.мс)	= 0.034396
Последняя операция	= Execute Immediate
Время начала последней операции	= 12-15-1999 10:29:06.142790
Время конца последней операции	= 12-15-1999 10:29:06.707053

```

Оператор = Execute Immediate
Номер раздела = 203
Создатель прикладной программы = NULLID
Имя пакета = SQLC2C07
Оценка компилятора SQL =
Оценка кардинальности компилятора SQL = 0
Временная отметка запуска оператора = 12-15-1999 10:29:06.142790
Временная отметка конца оператора = 12-15-1999 10:29:06.707053
Время ответа хоста (сек.мс) = 1.101612
Время вып.посл.заверш. оператора (сек.мс) = 0.564263
Выбрано строк = 0
Время на обработку на шлюзе = 0.013367
Входящих байтов принято для оператора = 220
Исходящих байтов послано для оператора = 130
Исходящих байтов принято для оператора = 49
Входящих байтов послано для оператора = 27
Текст оператора SQL:
create table t12 (col1 int, col2 char)

```

Дополнительную информацию о команде GET SNAPSHOT, а также о других полезных командах монитора DB2 смотрите в руководстве *System Monitor Guide and Reference*.

Вывод состояния программы DCS

Для DB2 Connect V5.2 монитор системы поддерживает три формы команды LIST DCS APPLICATIONS, а именно:

- LIST DCS APPLICATIONS
- LIST DCS APPLICATIONS SHOW DETAIL
- LIST DCS APPLICATIONS EXTENDED

Команда LIST DCS APPLICATIONS

Чтобы просмотреть регистрируемую монитором информацию уровня программы, введите команду DB2 LIST DCS APPLICATIONS. Для соединения APPC (DB2 Connect Enterprise Edition, Версия 7 с DB2 Universal Database for OS/390) будет получена следующая информация:

ID авт.	Имя программы	Хэндл программы	ID программы хоста
USERID	db2bp_41	0	CA1BOML.OMXT4H0A.A79EAA3C6E29

Для соединения TCP/IP (DB2 Connect Enterprise Edition, Версия 7 с DB2 Universal Database for OS/390) будет получена следующая информация:

ID авт.	Имя программы	Хэндл программы	ID программы хоста
USERID	db2bp_41	2	0915155C.9704.1517172201BE

ID авт. (ID авторизации)

ID авторизации, использованный при регистрации на сервере баз данных хоста или AS/400. Он показывает, кто запустил программу.

Имя программы

Имя программы клиента, под которым она известна DB2 Connect. Доступны только первые 20 байтов после последнего разделителя пути. Имя программы недоступно для программ, запущенных на DB2 for OS/2 Версии 1.

Хэндл программы

Агент, исполняемый на рабочей станции DB2 Connect. С помощью этого элемента можно связать информацию системного монитора баз данных с другой диагностической информацией. (Пример смотрите в разделе “Утилита трассировки (ddcstrc)” на стр. 167.) ID агента требуется также при использовании команды или API FORCE USERS.

ID программы хоста

Одно из следующих значений:

- Маркер соответствия DRDA (CRRTKN) для незащищенных диалогов
- Идентификатор логической единицы работы (LUWID) для двухфазных диалогов, если для защиты используется менеджер точек синхронизации SNA (SPM)
- Идентификатор единицы работы (UOWID) для двухфазных соединений, если для защиты используется менеджер точек синхронизации DRDA-3 (как в соединениях TCP/IP)

Это уникальный идентификатор, создаваемый в момент соединения программы с сервером баз данных хоста или AS/400. С помощью этого элемента и ID программы можно сопоставлять ту информацию о программе, которая относится к клиенту, с той информацией, которая относится к серверу.

LIST DCS APPLICATIONS SHOW DETAIL

Если команда выдана в формате DB2 LIST DCS APPLICATIONS SHOW DETAIL, будет показана дополнительная информация:

ID авт.	Имя программы	Хэндл программы	ID программы-клиента	Ном.
NEWTON	db2bp	0	09151251.07D3.980925183850	0001

Алиас клиента	DB клиента	Узел клиента	Выпуск клиента	Код.стран. клиента	ID программы хоста	Ном.
MVSDB	antman	SQL05020	819		G9151251.G7D4.980925183851	0000

Имя DB хоста	Выпуск хоста
GILROY	DSN05011

Этот отчет не форматирован, и для вас может оказаться полезнее отчет “Команда LIST DCS APPLICATIONS EXTENDED” на стр. 105.

ID программы-клиента

Уникальный идентификатор программы, соединенной с рабочей станцией DB2 Connect. Существуют разные форматы ID программы, зависящие от протокола связи между клиентом и рабочей станцией DB2 Connect. Дополнительную информацию об этих форматах смотрите в руководстве *Administration Guide*.

Данное значение позволяет вам сопоставить, с одной стороны, соединения клиентов с рабочей станцией DB2 Connect и, с другой стороны, соединения рабочей станции DB2 Connect с сервером баз данных хоста или AS/400.

Ном. (последовательный номер клиента)

Последовательный номер клиента — это последовательный номер транзакции. Он помогает определить соответствие транзакций, захватывающих несколько систем.

Алиас DB клиента

Алиас базы данных, сообщаемый программой, которая соединяется с базой данных. При помощи этого элемента можно определить фактическую базу данных, к которой обращается программа. Соответствие между этим именем и именем базы данных можно получить при помощи каталогов баз данных на узле клиента и узле сервера менеджера баз данных.

NNAME клиента (Узел клиента)

Идентифицирует узел, на котором выполняется программа клиента. Конкретная информация зависит от используемого протокола клиента. Например, для клиента, соединенного через NetBIOS, это значение равно параметру NNAME конфигурации менеджера баз данных. Для клиента, соединенного через TCP/IP, это значение — имя хоста.

ID продукта клиента (Выпуск клиента)

Продукт и версия продукта, запущенного на клиенте. Возможные ID продукта клиента:

- SQL01010 для Версии 1 продукта DB2 for OS/2
- SQL01011 для Версии 1 продуктов DB2 и программ Client Application Enabler для UNIX
- SQL02010 для Версии 2 продуктов DB2 и программ Client Application Enabler
- SQL02020 для Версии 2.1.2 продуктов DB2 и программ Client Application Enabler
- SQL05000 для Версии 5.0 продуктов DB2 Universal Database и DB2 Connect и их клиентов
- SQL05020 для Версии 5.2 продуктов DB2 Universal Database и DB2 Connect и их клиентов
- SQL06010 для Версии 6.1 продуктов DB2 Universal Database и DB2 Connect и их клиентов
- SQL07010 для Версии 7 продуктов DB2 Universal Database и DB2 Connect и их клиентов

ID кодовой страницы (Код. стран. клиента)

Идентификатор кодовой страницы узла, на котором запущена отслеживаемая программа.

При помощи этой информации можно проверить, поддерживается ли преобразование данных между кодовой страницей программы и кодовой страницей базы данных (или для сервера баз данных хоста или AS/400, CCSID сервера баз данных хоста или AS/400).

Если у программы кодовая страница отличается от страницы, под которой запущен системный монитор баз данных, этот элемент поможет вручную преобразовать данные, переданные программой и выведенные системным монитором баз данных. Например, с его помощью можно преобразовать имя программы.

Исходящий последовательный номер (Ном.)

Последовательный номер отправляемого сообщения. С его помощью можно сопоставить транзакции на разных системах.

Имя базы данных хоста (Имя DB хоста)

Реальное имя базы данных, с которой соединена программа. В каталоге DCS это *имя базы данных назначения*.

ID продукта хоста (Выпуск хоста)

Продукт и версия продукта, запущенного на сервере. Имеет вид *PPPVVRRM*, где:

- PPP** идентифицирует продукт базы данных хоста или AS/400 (например, DSN для DB2 Universal Database for OS/390, ARI для DB2 for VSE & VM, QSQ для DB2 Universal Database for AS/400)
- VV** Двухзначный номер версии, например, 01
- RR** Двухзначный номер выпуска
- M** Цифра уровня модификации

Команда LIST DCS APPLICATIONS EXTENDED

При помощи команды LIST DCS APPLICATIONS с опцией EXTENDED можно создать расширенный отчет. В нем присутствуют все поля, выводимые с опцией SHOW DETAIL, и еще девять полей:

- Состояние прикладной программы DCS
- Время изменения состояния
- Платформа клиента
- Протокол клиента
- CCSID хоста
- ID регистрации клиента
- ID процесса программы клиента
- Алиас базы данных на шлюзе
- Имя базы данных DCS

В отличие от прежних вариантов команды, где поля перечисляются горизонтально, по строке на программу, в этом варианте они перечисляются вертикально – по полю на строку.

Вот новый синтаксис этой команды:

```
LIST DCS APPLICATIONS [SHOW DETAIL | EXTENDED ]
```

Пример результата выполнения этой команды с новой опцией EXTENDED:

Список программ DCS - Расширенный отчет

ID программы клиента	= 09151251.0AA7.981015204853
Последовательный номер	= 0001
ID авторизации	= NEWTON
Имя прикладной программы	= db2bp
Хэндл прикладной программы	= 1
Состояние программы	= ожидание требования
Время изменения состояния	= 10-15-1998 16:50:29.489160
Узел клиента	= antman
Уровень выпуска клиента	= SQL05020
Платформа клиента	= AIX
Протокол клиента	= TCP/IP
Кодовая страница клиента	= 819
ID процесса программы клиента	= 39324
ID регистрации клиента	= smith
ID программы хоста	= G9151251.GAA8.981015204854
Последовательный номер	= 0000
Алиас базы данных на шлюзе	= MVSDB
Имя базы данных DCS	= DCSDB
Имя базы данных хоста	= GILROY
Уровень версии хоста	= DSN05011
CCSID хоста	= 500

Поле состояния программы содержит одно из следующих трех значений:

1. отложенное соединение - исходящее. Послано требование на соединение с базой данных хоста, и DB2 Connect ожидает, когда будет установлено соединение.
2. ожидание требования. Соединение с базой данных хоста установлено, и DB2 Connect ожидает поступления оператора SQL от программы клиента
3. ожидание ответа. Оператор SQL послан базе данных хоста.

Обратите внимание на то, что время изменения состояния выводится в отчете, только когда переключатель UOW Монитора системы во время обработки был включен. В противном случае в отчете выводится сообщение "не собрана".

Использование центра управления DB2 для вывода расширенной информации о программах DCS

При помощи центра управления DB2 Версии 7 можно следить за работой шлюза DB2 Connect. В этом разделе показано, как при помощи центра управления вывести тот же отчет, который создается командой **list dcs applications extended**.

Чтобы просмотреть расширенный отчет о какой-нибудь программе:

1. Разверните дерево в центре управления под значком **системы: Системы** —> **Экземпляры** —> **Соединения шлюза**. Если правой кнопкой мыши щелкнуть по любому экземпляру под папкой **Соединения шлюза**, появится

всплывающее меню. Выберите в нем Программы.... Появится окно Программы. Это окно выглядит как записная книжка с закладками, одна из которых помечена **Программы**. Если в вашем экземпляре есть программы шлюза, вторая закладка будет помечена **Программы шлюза**.

2. На каждой странице главное окно содержит столбцы информации, соответствующей полям отчета LIST DCS APPLICATIONS EXTENDED. Первые шесть столбцов, видимых в окне, содержат следующие данные:

- Узел клиента
- Имя прикладной программы
- ID программы клиента
- ID программы хоста
- Алиас базы данных на шлюзе
- Состояние

Остальные поля в отчете можно просмотреть, перемещая движок горизонтальной прокрутки в нижней части окна.

Все поля, перечисляемые командой LIST DCS APPLICATIONS EXTENDED, будут выводиться и в этом окне.

Использование монитора производительности Windows

В Windows NT и Windows 2000 есть полезный инструмент для наблюдения за производительностью ваших программ DB2. Монитор производительности — одно из средств управления Windows, графически показывает производительность системы. Вы можете выбирать различные показатели, связанные с системой, базой данных и связью, и выводить их общее графическое представление.

Например, отчеты, получаемые при помощи команды **GET SNAPSHOT FOR ALL DCS DATABASES** или **GET SNAPSHOT FOR ALL DCS APPLICATIONS**, можно выводить графически в реальном времени при помощи монитора и сравнивать прямо со значениями системных показателей, таких как занятость процессора. Можно непосредственно сравнивать действие различных параметров базы данных или связи. Ваши настроенные конфигурации можно сохранить в файлах PMS для последующего использования.

Например, на следующем рисунке несколько показателей DB2 графически сравниваются с занятостью процессора. Набор отображаемых значений сохраняется в файле db2chart.pmc. Можно сохранить столько файлов PMS, сколько вам надо для отражения разных факторов, влияющих на производительность системы.

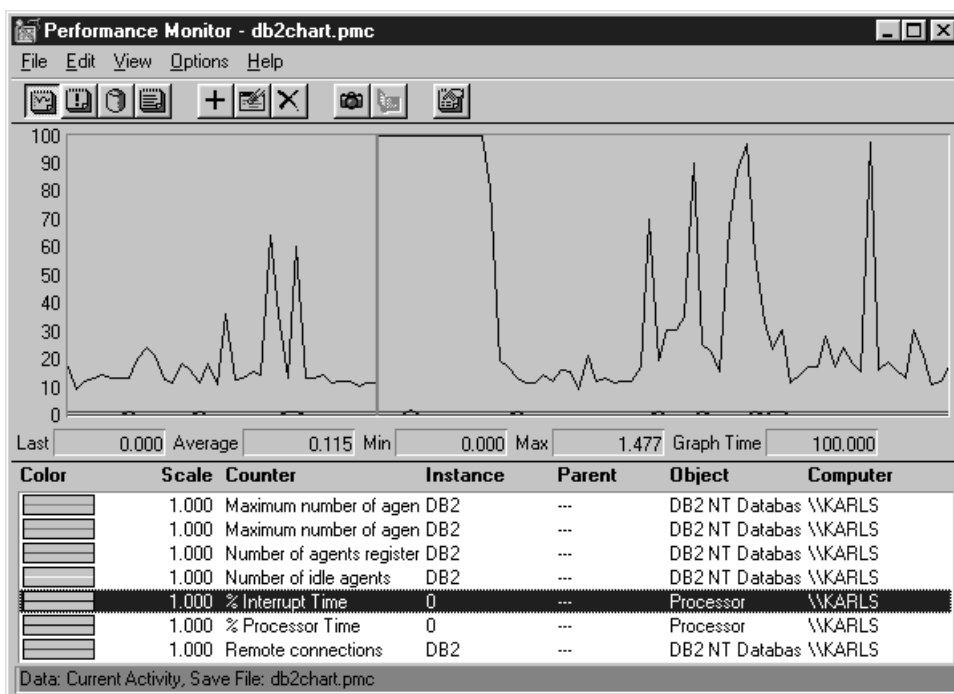


Рисунок 5. Монитор производительности

Для мониторинга локальных программ надо задать для переменной среды DB2CONNECT_IN_APP_PROCESS значение off.

Описание других команд DCS APPLICATIONS, а также дополнительную информацию об использовании монитора производительности Windows с DB2 UDB и DB2 Connect смотрите в руководстве *Administration Guide*.

Глава 9. Утилиты администратора

В этой главе описываются утилиты, помогающие выполнять задания управления. Глава состоит из следующих разделов:

- “Процессор командной строки”
- “Использование утилит импорта и экспорта” на стр. 110

Процессор командной строки

Процессор командной строки позволяет передавать серверу баз данных хоста или AS/400 операторы SQL после ключевого слова **db2**. О различиях между SQL хоста (или AS/400) и SQL DB2 Connect смотрите разделы “Операторы SQL сервера хоста или AS/400, поддерживаемые DB2 Connect” на стр. 55 и “Операторы SQL сервера хоста или AS/400, не поддерживаемые DB2 Connect” на стр. 55.

Чтобы вызвать процессор командной строки в режиме интерактивного ввода, надо:

OS/2 Щелкнуть по надписи **OS/2 Warp** и выбрать **IBM DB2** —> **Процессор командной строки** или ввести команду **db2**.

32-битные операционные системы Windows

Выбрать **Пуск**, затем **Программы** —> **IBM DB2** —> **Процессор командной строки**.

Примечание: Процессор командной строки в режиме интерактивного ввода можно также вызвать, введя в командной строке MS-DOS команду **db2cmd**, а потом команду **db2**.

UNIX Ввести в процессоре командной строки команду **db2**

Приглашение в режиме интерактивного ввода выглядит следующим образом:

```
db2 =>
```

В режиме интерактивного ввода не нужно ставить перед командой DB2 префикс **db2**; вместо этого вводится просто команда **DB2**. Например:

```
db2 => list database directory
```

Для ввода в интерактивном режиме команд операционной системы следует перед командой операционной системы ставить восклицательный знак (!). Например:

```
db2 => !dir
```

Если требуется ввести длинную команду, не помещающуюся на одну строку, введите символ продолжения командной строки \. Дойдя до конца строки, надо нажать клавишу **Enter**, чтобы продолжить ввод команды на следующей строке. Например:

```
db2 => select empno, function, firstname, lastname, birthdate, from \
db2 (cont.) => employee where function='service' and \
db2 (cont.) => firstname='Lily' order by empno desc
```

Чтобы выйти из режима интерактивного ввода, введите команду **quit**.

Чтобы использовать процессор командной строки, надо выполнить для него связывание, как описано в разделе Глава 7, “Связывание прикладных программ и утилит” на стр. 89.

Примечание: Если процессор командной строки используется в UNIX-системах не в интерактивном режиме ввода, специальные символы (например, * и ?) в операторах SQL надо заключать в двойные кавычки.

Дополнительную информацию о командах SQL смотрите в руководстве *Command Reference*.

Использование утилит импорта и экспорта

Утилиты импорта и экспорта позволяют перемещать данные из базы данных хоста S/390 или AS/400 в файл на рабочей станции DB2 Connect или наоборот. Впоследствии эти данные может использовать любая прикладная программа или RDBMS, поддерживающая данный формат импорта/экспорта. Например, можно экспортировать данные из DB2 for OS/390 в ASCII-файл с ограничителями, а потом импортировать их в базу данных DB2 UDB for Windows 2000.

Можно выполнять операции экспорта и импорта с клиента баз данных или с рабочей станции DB2 Connect.

Примечания:

1. Данные для импорта или экспорта должны удовлетворять ограничениям на размер и тип данных для обеих баз данных.
2. Для повышения производительности можно использовать составной SQL. Укажите `COMPOUND=число` в API импорта или же в строчном параметре `CLP filetype=mod`, чтобы сгруппировать в блок указанное число операторов SQL. Это может снизить расходы на передачу по сети и уменьшить время ответа.

3. Описание синтаксиса утилит импорта и экспорта, вызываемых из командной строки, смотрите в руководстве *Command Reference*.

Передача данных с рабочей станции на сервер баз данных S/390 или AS/400

Чтобы экспортировать данные на сервер баз данных S/390 или AS/400:

1. Экспортируйте строки информации из таблицы DB2 UDB for Unix, Windows NT, Windows 2000, OS/2 в файл PC/IXF.
2. Если в базе данных сервера DRDA нет таблицы с атрибутами, совместимыми с импортируемой информацией, создайте такую таблицу.
3. При помощи опции INSERT импортируйте файл PC/IXF в таблицу базы данных сервера DRDA.

Передача данных с сервера DRDA на рабочую станцию

Чтобы импортировать данные с сервера баз данных DRDA:

1. Экспортируйте строки информации из таблицы базы данных на сервере DRDA в файл PC/IXF.
2. Импортируйте файл PC/IXF в таблицу DB2.

Ограничения

При использовании программы DB2 Connect операции импорта или экспорта должны удовлетворять следующим требованиям:

- Тип файла должен быть PC/IXF.
- Определения индекса не записываются при экспорте и не используются при импорте.
- При импорте данных уже должна существовать таблица с атрибутами, совместимыми с атрибутами данных. При импорте посредством программы DB2 Connect нельзя создать таблицу, поскольку единственная поддерживаемая опция – это опция INSERT.
- При импорте нельзя указывать интервал счетчика принятия.

Если эти требования нарушены, операция завершится неудачно и будет сгенерировано сообщение об ошибке.

Смешанные однобайтные и двухбайтные данные

При импорте или экспорте смешанных данных (когда столбцы содержат как однобайтные, так и двухбайтные данные) нужно учитывать следующее:

- В системах, где данные хранятся в коде EBCDIC (MVS, OS/390, OS/400, VM и VSE), начало и конец двухбайтных данных обозначают символы переключения. При задании длин столбцов для таблиц базы данных проследите, чтобы было отведено место для этих символов.

- Рекомендуется использовать символьные столбцы переменной длины, за исключением тех случаев, когда все данные в столбце отвечают определенному шаблону. В последнем случае допустимо использовать постоянную длину.

Замена для утилиты SQLQMF

Функции утилиты SQLQMF для DDCS для OS/2 теперь выполняют функции Import/Export в DB2 Connect. Они имеют следующие преимущества:

- Не требуется QMF на хосте.
- Не нужно регистрироваться на хосте (ID TSO по-прежнему требуется для DB2 for MVS/ESA и DB2 Universal Database for OS/390).
- Поддерживаются DB2 for MVS, DB2 for OS/390, DB2 for OS/400 и DB2 for VM and VSE.
- При использовании составного SQL достигается хорошая производительность.
- Кроме ASCII, поддерживаются еще несколько форматов файлов.
- Функции можно запустить с клиентского компьютера, где не поддерживаются соединения SNA.

Дополнительную информацию об использовании этих команд смотрите в руководстве *Command Reference*.

Глава 10. Защита

В этой главе описываются возможности защиты DB2 Connect, в том числе типы аутентификации. Здесь приводятся также дополнительные советы пользователям DB2 Universal Database for OS/390, касающиеся защиты.

Дополнительные сведения про установку защиты в DCE смотрите в книге *Administration Guide*, а также в руководствах по базам данных и DCE для сервера баз данных хоста или AS/400.

Примечание: При использовании DB2 Connect с защитой DCE необходимо, чтобы программное обеспечение DCE было установлено на рабочей станции клиента DB2 и на хосте или сервере баз данных AS/400; не обязательно устанавливать его на рабочей станции DB2 Connect. Более подробная информация о требованиях DCE к программному обеспечению приводится в книге *DB2 Connect. Быстрый старт*.

Аутентификация

Администратор DB2 Connect вместе с администратором баз данных хоста или AS/400 может выяснить, где нарушены имена пользователей и пароли. Есть пять вариантов:

- Нарушение на клиенте
- Нарушение на рабочей станции DB2 Connect
- Нарушение и на рабочей станции DB2 Connect, и на сервере хоста или AS/400
- Нарушение на хосте или сервере AS/400
- Нарушение на сервере защиты DCE

Определить, где происходит нарушение, можно, задав параметр типа аутентификации в системном каталоге баз данных и параметр типа защиты в каталоге узлов для узлов APPC или APPN. Более подробная информация об изменении этих каталогов приводится в разделе Глава 6, “Изменение каталогов базы данных” на стр. 75.

Примечания:

1. Собственно DB2 Connect не производит проверки пользователей. Если необходимо, чтобы рабочая станция DB2 Connect производила проверку ID пользователя и пароля, указанных в каждом требовании CONNECT, будет использоваться локальная система защиты. Следовательно, если при установке сервера DB2 Connect Enterprise Edition задано AUTHENTICATION=SERVER, нужно также задать на системе сервера все необходимые ID пользователей и пароли.
2. Если используется служба каталога DCE, аутентификация работает иначе. Дополнительную информацию смотрите в разделе “Защита с использованием служб каталога DCE” на стр. 208.

В DB2 Connect применяются следующие типы аутентификации:

CLIENT

Имя пользователя и пароль проверяются на клиенте.

SERVER

Имя пользователя и пароль проверяются на рабочей станции DB2 Connect. Если тип аутентификации не задан, используется **SERVER**.

SERVER_ENCRYPT

Как и при аутентификации типа SERVER, имя пользователя и пароль проверяются на рабочей станции DB2 Connect, но передаваемые пароли шифруются на клиенте и расшифровываются на рабочей станции DB2 Connect.

DCS Имя пользователя и пароль проверяются на сервере баз данных хоста или AS/400.

DCS_ENCRYPT

Как и при аутентификации типа DCS, имя пользователя и пароль проверяются на сервере баз данных хоста или AS/400, но передаваемые пароли шифруются на клиенте и, в зависимости от типа аутентификации, заданного на рабочей станции DB2 Connect, расшифровываются на рабочей станции DB2 Connect или на сервере баз данных хоста или AS/400.

DCE Имя пользователя и пароль проверяются на сервере защиты DCE.

Типы аутентификации SERVER_ENCRYPT и DCS_ENCRYPT совпадают с типами аутентификации SERVER и DCS по месту проведения аутентификации. Они отличаются тем, что все передаваемые пароли будут зашифрованы в точке отправления (клиент или сервер DB2 Connect) и расшифрованы в точке прибытия (сервер DB2 Connect или сервер баз данных хоста или AS/400), как указано типом аутентификации, заданным в точке отправления.

Зашифрованные и незашифрованные значения с одинаковым местом аутентификации затем можно использовать для выбора различных сочетаний шифрования между клиентом и сервером DB2 Connect или между сервером

DB2 Connect и сервером баз данных хоста или AS/400, независимо от того, где будет проводиться аутентификация. Ниже приводятся примеры использования этого в сценарии шлюза (здесь под "шлюзом" понимается сервер DB2 Connect):

Аутентификация на клиенте	Аутентификация на шлюзе	Место аутентификации	Шифрование клиент–шлюз?	Шифрование шлюз–сервер?
SERVER_ENCRYPT	SERVER	шлюз	да	нет
DCS_ENCRYPT	DCS	сервер	да	нет
DCS	DCS_ENCRYPT	сервер	нет	да
DCS_ENCRYPT	DCS_ENCRYPT	сервер	да	да

Единственный параметр защиты APPC, поддерживаемый, когда включен режим SERVER_ENCRYPT или DCS_ENCRYPT – SECURITY=NONE.

Примечания:

1. Для любой записи из системного каталога баз данных, используемой DB2 Connect для установления соединения, если параметр аутентификации не указан, DB2 Connect будет использовать аутентификацию **SERVER**.
2. Как и для связей клиент–сервер DB2 Universal Database, тип аутентификации на удаленном клиенте, подключенном к шлюзу DB2 Connect Enterprise Edition, указывать не обязательно. Можно указать его, чтобы улучшить быстродействие (поскольку тогда его не нужно будет получать с шлюза), что сократит время, которое уходит на передачу данных.
3. При несоответствии между значением на клиенте и значением на шлюзе приоритет имеет значение, указанное на шлюзе DB2 Connect.

Типы защиты

В этом разделе перечислены различные сочетания параметров аутентификации и защиты, поддерживаемые DB2 Connect для соединений APPC и TCP/IP.

Далее описаны оба этих типа соединения.

Типы защиты для соединений APPC

Для соединений APPC доступны следующие типы защиты, определяющие передаваемую на уровне связи информацию:

SAME На сервер баз данных хоста или AS/400 передается только имя пользователя.

PROGRAM

На сервер баз данных хоста или AS/400 передаются имя пользователя и пароль.

NONE Не передается никакой информации защиты.

В Табл. 5 перечислены возможные сочетания этих значений и типа аутентификации, заданного на рабочей станции DB2 Connect, а также указано, производится ли проверка для каждого сочетания. Соединения DB2 Connect через APPC поддерживают только сочетания, перечисленные в этой таблице.

Таблица 5. Сценарии защиты, возможные при соединениях APPC

Случай	Параметр аутентификации в записи из каталога баз данных на рабочей станции DB2 Connect	Защита	Проверка
1	CLIENT	SAME	Клиент
2	SERVER	SAME	Сервер DB2 Connect
3	SERVER	PROGRAM	Сервер DB2 Connect и сервер баз данных хоста или AS/400
4	SERVER_ENCRYPT или DCS_ENCRYPT	NONE	Сервер баз данных хоста или AS/400
5	DCS	PROGRAM	Сервер баз данных хоста или AS/400
6	DCE (конечное оборудование для передачи данных DCE)	NONE	Сервер защиты DCE

Если удаленные клиенты соединяются с сервером DB2 Connect Enterprise Edition, указывайте следующие типы аутентификации и защиты:

- Если удаленный клиент соединяется с сервером DB2 Connect через APPC, на удаленном клиенте должен быть указан тип защиты NONE.
- Если в конфигурации менеджера баз данных на клиенте DB2 Connect указан тип аутентификации CLIENT, на всех удаленных клиентах также должен быть указан тип аутентификации CLIENT.
- Если на сервере DB2 Connect указан тип аутентификации SERVER, SERVER_ENCRYPT, DCS или DCS_ENCRYPT, на каждом удаленном клиенте также должен быть указан один из этих типов. (Не имеет значения, какой именно из этих 4 типов указан на удаленном клиенте.)

Примечания:

1. Для систем AIX все входящие в систему пользователи, использующие тип защиты APPC SAME, должны принадлежать к группе AIX system (системной).
2. Для систем AIX с удаленными клиентами, экземпляр программы DB2 Connect, выполняемый на рабочей станции DB2 Connect, должен принадлежать к группе AIX system (системной).
3. Доступ к серверу баз данных хоста или AS/400 управляется его собственными механизмами и подсистемами защиты, например, Virtual Telecommunications Access Method (VTAM) или Resource Access Control Facility (RACF). Доступ к защищенным объектам баз данных управляется операторами SQL **GRANT** и **REVOKE**.

Типы защиты для соединений TCP/IP

Протокол связи TCP/IP не поддерживает параметры защиты на уровне сетевого протокола. Таким образом, на то, где происходит аутентификация, влияет только тип аутентификации. Соединения DB2 Connect через TCP/IP поддерживают только сочетания, перечисленные в следующей таблице.

Случай	Параметр аутентификации в записи из каталога баз данных на рабочей станции DB2 Connect	Проверка
1	CLIENT	Клиент
2	SERVER или SERVER_ENCRYPT	Рабочая станция DB2 Connect
3	Неприменимо	Нет
4	DCS или DCS_ENCRYPT	Сервер баз данных хоста или AS/400
5	DCE (конечное оборудование для передачи данных DCE)	Сервер защиты DCE

Обсуждение типов защиты

Дальнейшее обсуждение относится и к соединениям APPC, и к соединениям TCP/IP, описанным выше и перечисленным в таблицах Табл. 5 на стр. 116 и Табл. 6. Все случаи подробно рассмотрены ниже:

- В случае 1 имя пользователя и пароль проверяются только на удаленном клиенте. (Для локального клиента имя пользователя и пароль проверяются только на сервере DB2 Connect.)

Ожидается, что пользователь проходит аутентификацию там, где он регистрируется. По сети посылается ID пользователя, но не пароль. Используйте этот тип защиты, только если на всех рабочих станциях клиентов имеются адекватные механизмы защиты, которым можно доверять.

- В случае 2 имя пользователя и пароль проверяются только на сервере DB2 Connect. Пароль посылается по сети с удаленного клиента на сервер DB2 Connect, но не на сервер баз данных хоста или AS/400.
- В случае 3 имя пользователя и пароль проверяются и на сервере DB2 Connect, и на сервере баз данных хоста или AS/400. Пароль посылается по сети с удаленного клиента на рабочую станцию DB2 Connect, а с рабочей станции DB2 Connect на сервер баз данных хоста или AS/400.

Поскольку проверка производится в двух местах, на сервере DB2 Connect и на сервере баз данных хоста или AS/400 должны быть одинаковые наборы имен пользователей и паролей.

- В случае 4 имя пользователя и пароль проверяются только на сервере баз данных хоста или AS/400. ID пользователя и пароль посылаются по сети с удаленного клиента на сервер DB2 Connect, а с сервера DB2 Connect на сервер баз данных хоста или AS/400.
- В случае 5 клиент получает с сервера защиты DCE зашифрованную квитанцию DCE. Эта квитанция передается в неизменном виде через DB2 Connect на сервер, где она проверяется с помощью служб защиты DCE.

Изменение пароля MVS

DB2 Connect теперь предоставляет возможность изменять пароль пользователя. Эта возможность особенно полезна в ситуациях, когда для аутентификации пользователей используются средства защиты хоста, например, RACF. Раньше для смены пароля на хосте пользователям приходилось запускать сеанс TSO. Новые средства поддержки пароля, предоставляемые продуктами DB2 Connect, позволяют пользователям изменить пароль на хосте, введя оператор SQL CONNECT из командной строки DB2 (CLP) или нажав кнопку Пароль в ассистенте конфигурирования клиента DB2 (CCA) или кнопку Изменить в диалоговом окне регистрации ODBC.

Для изменения пароля MVS с рабочей станции DB2 Connect, соединенной с DB2 for OS/390 V5.1 через TCP/IP, требуется, чтобы поле DB2 OS/390 Extended Security содержало значение "Yes". Это поле выводится на панели DB2 OS/390 DSNTIPR.

Для изменения пароля MVS на хост-системах, соединенных через SNA, требуется, чтобы на хосте была установлена специальная программа

управления сроками действия паролей и чтобы рабочая станция DB2 Connect была сконфигурирована для связи с этой программой хоста.

Программа управления сроками действия паролей поставляется в составе следующих программных продуктов MVS:

- MVS/ESA SP Версии 4.2 и более новых (управление сроками действия паролей входит в компонент APPC/MVS)
- CICS/ESA Версии 3.3 и более новых

(для этого должна быть установлена система IBM Resource Access Control Facility (RACF) 1.9.2)

Необходимо:

1. Сконфигурировать программу транзакций хоста, чтобы она принимала требования для управления сроками действия паролей
2. Сконфигурировать рабочую станцию DB2 Connect для связи с программой транзакций хоста

Конфигурирование рабочей станции DB2 Connect для работы с программой управления сроками действия паролей

После того как программа транзакций управления сроками действия паролей на хосте сконфигурирована, нужно сконфигурировать рабочую станцию DB2 Connect для связи с этой программой. Конфигурирование состоит из двух этапов:

1. Задание для подсистемы SNA на рабочей станции DB2 Connect символического имени назначения для программы поддержки паролей на хосте
2. Занесение этого символического имени назначения в каталог DCS для баз данных, расположенных в этой хост-системе

Этап 1. Задание символического имени назначения

Задание символического имени назначения для программы контроля срока действия паролей на хосте зависит от используемой подсистемы SNA:

- Если используется подсистема SNA, которую можно сконфигурировать при помощи ассистента конфигурирования клиента DB2 (CCA), для задания символического имени назначения нужно использовать CCA. Необходимо узнать у администратора MVS имя LU для программы управления сроками действия паролей.
- Если подсистему нельзя сконфигурировать с помощью CCA, необходимо сконфигурировать символическое имя назначения, как описано в документации по подсистеме SNA. Необходимо узнать у администратора MVS следующую информацию:
 - Сетевое имя хоста, с которым устанавливается соединение

— Имя LU программы управления сроками действия паролей на хосте

При конфигурировании символического имени назначения нужно в качестве имени программы транзакций (TP) указать x'06F3F0F1' (шестнадцатеричное число), а для параметра security (защита) задать значение NONE. Можете также указать режим, например, #INTER или любой другой допустимый в MVS режим.

Этап 2. Занесение символического имени назначения в каталог DCS

Если DB2 Connect работает на платформе, где есть ассистент конфигурирования клиента (CCA), для внесения в каталог DCS символического имени программы управления сроками действия паролей на хосте нужно использовать CCA. Это можно сделать независимо от того, какая подсистема SNA используется на рабочей станции DB2 Connect.

Для занесения символического имени назначения в каталог DCS можно также использовать команду **catalog dcs database** (из командной строки DB2).

Например:

```
catalog dcs database db1 as dsn_db_1 parms ",,,,,,CHGPWD_SDN=petpgm"
```

задает символическое имя назначения *petpgm*; оно будет использоваться для требований пользователей изменить пароль для базы данных *db1*.

Конфигурирование хоста для поддержки управления сроками действия паролей

Более подробную информацию о паролях MVS смотрите в следующих электронных публикациях:

APPC/MVS:

<http://www.s390.ibm.com/products/appc/library>

SecureWay Communications:

<http://www.ibm.com/software/network/commserver/library>

TxSeries:

<http://www.ibm.com/software/ts/txseries/library>

Дополнительные советы, касающиеся защиты

В этом разделе для пользователей DB2 Connect приводятся некоторые дополнительные советы, касающиеся защиты.

Расширенные коды защиты

В версиях DB2 Universal Database for OS/390 до Версии 5.1 требования на соединение с ID пользователя или паролем могли завершиться неудачно с

кодом причины SQL30082, равным 0, но никакой уточняющей информации о причинах ошибки не выдавалось.

В Версии 5.1 DB2 Universal Database for OS/390 была введена поддержка расширенных кодов защиты. При использовании расширенного кода защиты можно, в дополнение к коду причины, получать дополнительную диагностику, например, (PASSWORD EXPIRED) (срок действия пароля истек).

Для этого параметр установки DB2 Universal Database for OS/390 ZPARM должен иметь значение YES. На панели установки DB2 Universal Database for OS/390 DSN6SYSP задайте EXTSEC=YES. Для задания этого параметра можно также использовать панель 1 DDF (DSNTIPR). По умолчанию используется значение EXTSEC=NO. В таком случае при истечении срока действия пароля программы PC, UNIX, Apple Macintosh и программы Web, использующие DB2 Connect, получают сообщение об ошибке SQL01404.

Защита TCP/IP уже проверена

Если вы хотите обеспечить поддержку опции защиты DB2 Universal Database AUTHENTICATION=CLIENT, на панели установки DB2 Universal Database for OS/390 DSNTIP4 (панель 2 DDF) задайте в поле "TCP/IP already verified" значение YES.

Защита программ ODBC и Java на настольных системах

Программы ODBC и Java на рабочих станциях используют динамический SQL. На некоторых установках это может привести к проблемам с защитой. В DB2 Universal Database for OS/390 введена новая опция связывания DYNAMICRULES (BIND), которая позволяет по выбору выполнять динамический SQL с авторизацией владельца или же того, кто выполнял связывание. Посмотрите в справочнике *Command Reference*, как задать DYNAMICRULES через DB2 Connect.

В DB2 Universal Database и DB2 Connect в файле конфигурации DB2CLI.INI введен новый параметр конфигурации CLI/ODBC CURRENTPACKAGESET. Его значением должно быть имя схемы с соответствующими привилегиями. При каждом соединении такой программы автоматически будет выполняться оператор SQL SET CURRENT PACKAGESET схема.

Для исправления файла DB2CLI.INI используйте менеджер ODBC. Дополнительную информацию смотрите в книге *Дополнение по установке и настройке*.

Поддержка изменения пароля

Если в ответ на оператор SQL CONNECT пользователь получает сообщение об истечении срока действия его пароля, в DB2 Connect Версии 5.2 и более новых он может изменить пароль, не регистрируясь на TSO. DB2 Universal Database for OS/390 может сменить пароль для вас через DRDA.

Пользователь должен задать старый пароль, новый пароль и его подтверждение. Если на сервере DB2 Connect Enterprise Edition задана защита DCS, требование изменения пароля посылается на сервер баз данных DB2 Universal Database for OS/390. Если задана защита SERVER, изменяется пароль на сервере DB2 Connect.

Дополнительное преимущество состоит в том, что отдельное определение LU не требуется. Дополнительную информацию смотрите в руководстве DB2 Connect Enterprise Edition *Quick Beginnings*.

Глава 11. Отображение SQLCODE

Различные программы работы с реляционными базами данных IBM не всегда выдают одинаковые SQLCODE для одинаковых ошибок. Если даже SQLCODE тот же, он может сопровождаться различными элементами сообщения. Список этих элементов передается в поле SQLERRMC в SQLCA. По умолчанию DB2 Connect отображает SQLCODE и элементы сообщения каждого сервера баз данных IBM хоста или AS/400 в соответствующие SQLCODE DB2 Universal Database.

Отключение отображения SQLCODE

Чтобы отключить отображение SQLCODE, укажите NOMAP в строке параметров каталога DCS или объекта информации маршрутизации DCE. Информацию о внесении изменений в каталог DCS смотрите в разделе Глава 6, “Изменение каталогов базы данных” на стр. 75. Информацию об использовании DCE смотрите в разделе Приложение D, “Использование служб каталога DCE” на стр. 201.

Если вы переносите прикладную программу непосредственно с сервера баз данных хоста или AS/400 (например, DB2 UDB for OS/390), может понадобиться отключить отображение SQLCODE. Это позволит использовать прикладную программу, не меняя соответствующие ей SQLCODE.

Оптимизация отображения SQLCODE

По умолчанию DB2 Connect отображает SQLCODE и элементы сообщения каждого сервера баз данных IBM хоста или AS/400 в соответствующие SQLCODE DB2 UDB. Следующие файлы содержат копии отображения SQLCODE по умолчанию:

dcs1dsn.map

Отображает SQLCODE для DB2 for MVS/ESA and DB2 UDB for OS/390

dcs1ari.map

Отображает SQLCODE для DB2 for VSE & VM

dcs1qsq.map

Отображает SQLCODE для DB2 UDB for AS/400

Для систем DB2 на базе OS/2 и UNIX отображение не требуется.

Если вы хотите переопределить отображение SQLCODE по умолчанию или используете сервер баз данных хоста или AS/400, для которого нет отображения SQLCODE (сервер баз данных другого разработчика), можно

скопировать один из этих файлов и использовать его как основу для нового файла отображения SQLCODE. Этот файл лучше скопировать, а не редактировать непосредственно, чтобы осталась возможность при необходимости вернуться к первоначальному отображению SQLCODE.

В строке параметров каталога DCS или объекта информации маршрутизации DCE укажите имя нового файла отображения SQLCODE. Информацию о внесении изменений в каталог DCS смотрите в разделе Глава 6, “Изменение каталогов базы данных” на стр. 75. Информацию об использовании DCE смотрите в разделе Приложение D, “Использование служб каталога DCE” на стр. 201.

Файл отображения – это простой ASCII-файл, и его можно создавать и редактировать с помощью ASCII-редактора. При начальной установке этот файл находится в каталоге `var` пути установки.

В этом файле могут использоваться следующие особые типы строк:

- &&** Логическое начало файла. Все строки до первого появления **&&** рассматриваются как произвольные комментарии и игнорируются. Если после **&&** в файле ничего нет, отображение SQLCODE не выполняется. Отключить отображение SQLCODE можно также с помощью параметра `NOMAP`, как описано выше.
- *** Строка, начинающаяся со звездочки, воспринимается как комментарий.
- W** Как единственный символ строки, указывает, что флаги предупреждений подлежат отображению. По умолчанию передаются исходные флаги предупреждений. **W** надо задавать в верхнем регистре.

Все остальные строки после **&&** должны быть либо пустыми, либо операторами отображения в следующей форме:

```
входной_код [, выходной_код [, список_элементов]]
```

входной_код может быть:

sqlcode SQLCODE сервера баз данных хоста или AS/400.

- U** Все неопределенные отрицательные SQLCODE (те, что не перечислены в этом файле) отображаются в указанный *выходной_код*. Если в этой строке не указан *выходной_код*, используется первоначальный SQLCODE. Этот символ надо задавать в верхнем регистре.
- P** Все неопределенные положительные SQLCODE (те, что не перечислены в этом файле) отображаются в указанный *выходной_код*. Если в этой строке не указан *выходной_код*, используется первоначальный SQLCODE. Этот символ надо задавать в верхнем регистре.
- csnn** Код класса SQLSTATE сервера баз данных хоста или AS/400. *nn* может быть одним из следующих:

00	Успешное завершение без уточнения
01	Предупреждение
02	Нет данных
21	Нарушение количества элементов
22	Исключительное состояние данных
23	Нарушение ограничений
24	Недопустимое состояние указателя
26	Недопустимый идентификатор оператора SQL
40	Откат транзакции
42	Нарушение правил доступа
51	Недопустимое состояние программы
55	Объект не в требуемом состоянии
56	Различные ошибки программы или SQL
57	Ресурс недоступен или вмешательство оператора
58	Системная ошибка

Указанный *выходной_код* используется для всех SQLCODE с данным кодом класса, не указанных явно в файле отображения. Если в этой строке не указан *выходной_код*, первоначальный SQLCODE отображается сам в себя, а элементы сообщения не перекопируются.

Символы **cc** вводятся в нижнем регистре.

Если в файле отображения один и тот же *входной_код* появляется несколько раз, используется первое вхождение.

выходной_код означает выходной SQLCODE. Если никакое значение не указано, используется исходный SQLCODE.

При указании выходного кода можно указать также одно из следующего:

- (s) SQLCODE ввода плюс ID продукта (ARI, DSN или QSQ) будут помещены в поле элемента сообщений SQLCA.

Первоначальный SQLCODE возвращается как единственный элемент. Эта опция предназначена для неопределенных SQLCODE, за исключением +965 и -969. Если *выходной_код* принимает значение +965 или -969, список элементов, возвращаемый в поле SQLERRMC SQLCA, содержит исходный SQLCODE, за которым следуют идентификатор программы и первоначальный список элементов.

Символ **s** задается в нижнем регистре.

(*список_элементов*)

Список элементов через запятую. Чтобы пропустить отдельный элемент, укажите только запятую. Например, выражение *(,t2,,t4)* означает, что первый и третий элементы вывода пусты.

Каждый элемент имеет вид числа (*n*), перед которым может стоять **c** и за которым может следовать **c** или **i**. Это означает следующее:

- c** Тип данных элемента в этой позиции – CHAR (предполагается по умолчанию). Если **c** стоит перед *n*, он относится ко входному элементу, а если после *n* – к выходному элементу. Символ **c** задается в нижнем регистре.
- i** Тип данных элемента в этой позиции – INTEGER. Если **i** стоит после *n*, он относится к выходному элементу. **i** не должен стоять перед *n*, потому что программы сервера баз данных IBM хоста или AS/400 поддерживают только элементы типа CHAR. Символ **i** задается в нижнем регистре.
- n* Число или числа, указывающие, какие элементы сервера баз данных хоста или AS/400 используются. Они указывают требуемый порядок размещения элементов в выходной области SQLCA. Число означает порядковый номер элемента сервера баз данных хоста или AS/400; порядок этих чисел соответствует порядку, в котором элементы будут помещены в SQLCA.

Например, сервер баз данных хоста или AS/400 должен вернуть два элемента, 1 и 2. Если вы хотите, чтобы в выходной SQLCA второй элемент шел перед первым, укажите (2,1).

Можно записать несколько номеров элементов, соединив их точками, чтобы объединить их в один выходной элемент типа CHAR.

Выходные элементы разделяются запятыми. Если элемент перед запятой не указан, для данной позиции никакой выходной элемент в SQLCA не записывается. Любые элементы выходной SQLCA, следующие за последним указанным элементом, будут пустыми.

На рис. 6 на стр. 127 показан пример файла отображения SQLCODE.


```

&&
-007 , -007 , (1)
-010
-060 , -171 , (2)
...
-204 , -204 , (c1.2c)
...
-633 , -206 , (,c1i)

-30021 , -30021 , (c1c,c2c)

cc00 , +000
...
U , -969 , (s)
P , +965 , (s)

```

Рисунок 6. Файл отображения SQLCODE

Операторы отображения в этом файле интерпретируются так:

1. SQLCODE –007 отображается в –007. Первый входной элемент, полученный от сервера баз данных хоста или AS/400, используется в качестве первого выходного элемента, ему по умолчанию присваивается тип CHAR. Другие элементы не передаются.
2. SQLCODE –010 отображается в –010 (поскольку выходной SQLCODE не указан). Никакие элементы в выходную SQLCA не помещаются.
3. SQLCODE –060 отображается в –171. Первый входной элемент, полученный от сервера баз данных хоста или AS/400, отбрасывается. Второй используется как первый элемент выходной SQLCA с типом CHAR. В выходной SQLCA второго элемента нет.
4. SQLCODE –204 отображается в –204. Первый и второй элементы, полученные от сервера баз данных хоста или AS/400, имеют тип CHAR. Оба входных элемента объединяются в один выходной элемент типа CHAR, который будет первым элементом в SQLCA.
5. SQLCODE –633 отображается в –206. Первый входной элемент, полученный от сервера баз данных хоста или AS/400, имеет тип CHAR. Он преобразуется в INTEGER и используется как второй элемент выходной SQLCA. Первый элемент выходной SQLCA остается пустым, что указано запятой.
6. SQLCODE –30021 отображается в –30021. Первый и второй входные элементы, полученные от сервера баз данных хоста или AS/400, имеют тип CHAR и используются как первый и второй элементы выходной SQLCA.
7. Все SQLCODE в SQLCA, SQLSTATE которых принадлежит классу 00, будут отображены в SQLCODE +000.
8. Все неопределенные SQLCODE отображаются в –969. Эту опцию следует использовать, только если уже перечислены все отображаемые коды, в том числе и те, которые не изменяются и не требуют отображения. Опция (s) указывает, что список элементов, возвращаемых в поле SQLERRMC в

SQLCA, включает первоначальный SQLCODE, за которым следует продукт, где произошла ошибка, а за ней – первоначальный список элементов. Если строка **U** не включена, все не входящие в список коды передаются без отображения.

9. Все неопределенные положительные SQLCODE отображаются в +965. Эту опцию следует использовать, только если уже перечислены все отображаемые коды, в том числе и те, которые не изменяются и не требуют отображения. Опция **(s)** указывает, что список элементов, возвращаемых в поле SQLERRMC в SQLCA, включает первоначальный SQLCODE, за которым следует продукт, где произошла ошибка, а за ней – первоначальный список элементов. Если строка **P** не включена, все не входящие в список положительные коды передаются без отображения.

Глава 12. Производительность

DB2 Connect взаимодействует со множеством других продуктов, включая серверы прикладных программ DRDA, клиенты и программы связи. Ее производительность зависит от эффективности их совместной работы.

Производительность: основные понятия и средства

Производительность определяет, как компьютерная система справляется с определенным объемом работы. Производительность зависит от доступных ресурсов, их использования и распределения. Чтобы улучшить производительность, прежде всего надо решить, что под ней понимается. Можно выделить различные *показатели производительности*, в том числе:

Время ответа

Промежуток между отправкой требования к базе данных и получением ответа.

Производительность транзакций

Количество единиц работы, завершаемых в единицу времени. Единица работы может быть простой, например, поиск и обновление строки, или сложной, включающей сотни операторов SQL.

Скорость передачи данных

Объем данных в байтах, которыми обмениваются прикладная программа DB2 Connect и база данных хоста или AS/400 за единицу времени.

Производительность ограничивается программными и аппаратными ресурсами. Примеры аппаратных ресурсов – процессоры, память и сетевые адаптеры. Примеры программных ресурсов – подсистемы связи, подкачки страниц, mbuf для AIX и link для SNA.

Потоки данных

На рис. 7 на стр. 130 показан путь передачи данных между сервером баз данных хоста или AS/400 и рабочей станцией через DB2 Connect.

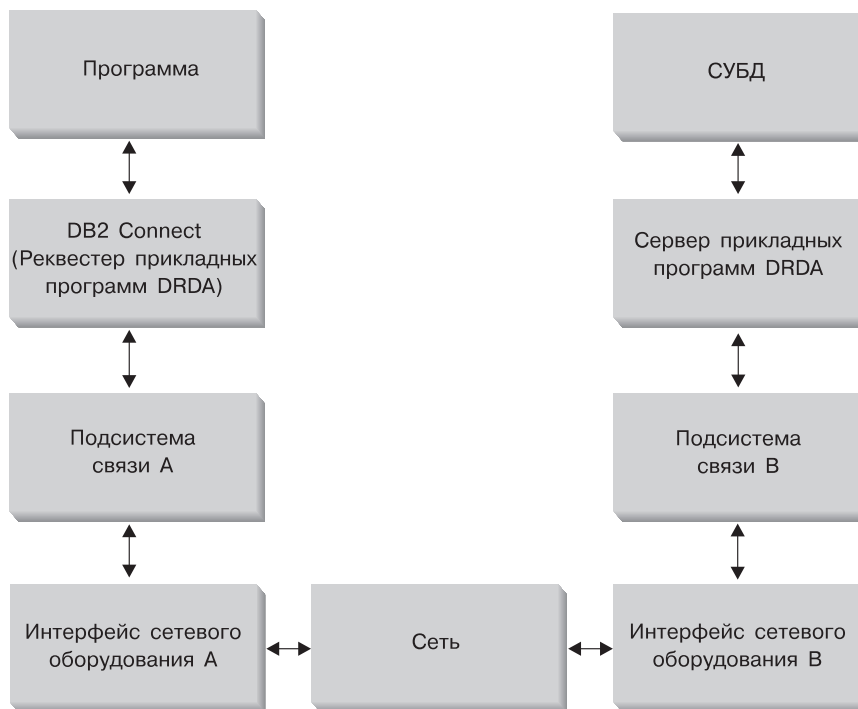


Рисунок 7. Потоки данных в DB2 Connect

- База данных хоста или AS/400 и часть В подсистемы связи обычно работают на одной системе, состоящей из одного или нескольких процессоров, памяти подсистемы ввода/вывода, запоминающего устройства прямого доступа и операционной системы. Эти компоненты могут одновременно использоваться другими программами, и конкуренция за ресурсы уменьшает производительность.
- Сеть состоит из сочетания кабелей, концентраторов, линий связи, переключателей и других контроллеров связи. Например, оборудование сетевого интерфейса В может представлять собой контроллер связи типа 3745 или 3172 или адаптер Token Ring для AS/400. Передача между аппаратными интерфейсами А и В может идти через несколько сред.
- Оборудование сетевого интерфейса А может представлять собой адаптер Token Ring, Ethernet**, другой сетевой адаптер или адаптер, поддерживающий протоколы SDLC или X.25. В качестве подсистемы связи А можно использовать, например, IBM Communications Server для OS/2, Microsoft SNA Server, IBM SNA Server для AIX или SNAplus2 для HP-UX.
- Продукт DB2 Connect и подсистема связи А обычно расположены на одной системе. В данной главе предполагается, что прикладная программа работает в той же системе.

Критические участки

Производительность транзакций зависит от самого медленного компонента в системе. Установив "узкие места", часто удается повысить производительность, изменив параметры настройки, выделив критическому компоненту больше ресурсов, обновив его или добавив новый компонент, чтобы снять часть нагрузки.

Существуют программы, определяющие, сколько времени тратится на запрос каждым компонентом. Эта информация покажет, какие компоненты надо настроить или обновить для повышения производительности. Например, если оказалось, что 60% времени обработки запроса расходуется на компьютере DB2 Connect, можно настроить DB2 Connect или (если есть удаленные клиенты) добавить в сеть еще один компьютер DB2 Connect.

Подробную информацию о средствах определения производительности смотрите в разделе "Средства определения производительности."

Тестирование производительности

Тестирование производительности – способ сравнить производительность в одной среде с производительностью в другой.

Начать тестирование можно с запуска эталонной программы в обычной среде. Когда источник низкой производительности установлен, можно разработать специальные тесты для конкретных функций.

Тестирование не должно быть сложным. Специальные тесты, не эмулируя всю программу, могут тем не менее давать ценную информацию. Начинайте с простых измерений и не усложняйте их без необходимости.

Признаки правильности тестирования:

- Все тесты повторяемы.
- Каждый прогон теста начинается в одном и том же состоянии системы.
- Устройства и программы, используемые при тестировании, соответствуют реальным условиям вашей работы.
- Активны только тестируемые функции или программы (если в сценарий не входит учет посторонних процессов в системе).

Примечание: Запущенные программы используют память, даже когда они свернуты и не активны. Это может вызвать подкачку и исказить результаты тестирования.

Средства определения производительности

В следующей таблице перечислены некоторые программы, которые помогают измерить производительность системы. Они сами используют системные ресурсы, поэтому, возможно, лучше не держать их постоянно активными.

<i>Таблица 7 (Стр. 1 из 2). Средства определения производительности</i>		
Система	Программа	Описание
Использование процессора и памяти		
AIX	vmstat, time, ps, tprof	Дают информацию о конкуренции за процессор и память на рабочей станции DB2 Connect и на удаленных клиентах.
HP-UX	vmstat, time, ps, monitor и glance, если это возможно	
OS/2	SPM/2, THESEUS/2, pstat	
Windows NT и Windows 2000	MS Performance Monitor	
Действия с базами данных		
Все	Монитор баз данных	Определяет, является ли база данных источником затруднений.
MVS или OS/390	DB2PM (IBM), OMEGAMON/DB2 (Candle), TMON (Landmark), INSIGHT (Goal Systems) и DB2AM (BMC)	
Windows NT и Windows 2000	MS Performance Monitor	
Сетевые процессы		
AIX	netrmon	Дает статистику низкого уровня, включая статистику TCP/IP и SNA, такую как число пакетов или кадров, получаемых за секунду.
DOS или OS/2	Token-Ring Network 16/4 Trace and Performance Program	Большинство мониторов сети предназначены для определенной платформы; эта программа предназначена только для сети Token Ring.

<i>Таблица 7 (Стр. 2 из 2). Средства определения производительности</i>		
Система	Программа	Описание
Сетевые контроллеры типа 3745	NetView Performance Monitor	Дает сведения об управлении связью и VTAM.
OS/2	DatagLANce	Средство трассировки, дающее сведения о производительности в виде графиков.
На основе UNIX	netstat	Обрабатывает трафик TCP/IP.

Оптимизация доступа ODBC

DB2 Universal Database позволяет оптимизировать связь через ODBC для повышения производительности. Эту оптимизацию связи можно использовать для Microsoft Access, Lotus Approach и Visual Basic. Чтобы сконфигурировать более быструю связь ODBC, используйте Ассистент конфигурирования клиента (CCA).

Чтобы активировать оптимизированную связь ODBC:

При определении нового соединения:

1. Запустите CCA DB2.
2. Выберите алиас базы данных, работу с которой нужно оптимизировать.
3. Нажмите кнопку **Свойства**.
4. Убедитесь, что включен переключатель **Зарегистрировать эту базу данных для ODBC**.
5. Выберите радиокнопку, описывающую требуемый тип регистрации базы данных.
6. Нажмите кнопку **Параметры**.
7. Нажмите кнопку **Оптимизировать** и выберите прикладную программу, для которой нужно оптимизировать параметры ODBC.
8. Нажмите кнопку **ОК** и завершите работу с CCA.

При изменении существующего нового соединения:

1. Запустите CCA DB2.
2. Выберите алиас базы данных, работу с которой нужно оптимизировать.
3. Нажмите кнопку **Свойства**.
4. Нажмите кнопку **Параметры**.
5. В окне Параметры CLI/ODBC нажмите кнопку **Оптимизировать** и выберите прикладные программы, для которых нужно оптимизировать ODBC.

6. Нажмите кнопку **ОК** и завершите работу с ССА.

Дополнительную информацию о ССА смотрите в книге *DB2 Connect Quick Beginnings*. Самую свежую информацию об оптимизации доступа ODBC для DB2 для OS/390 можно получить в Интернете по адресу:
<http://www.ibm.com/software/data/db2/os390/odbcatl.html>

Разработка программ

Создавая программу, можно повысить производительность несколькими способами, в том числе:

- Использовать составные операторы SQL и хранимые процедуры
- Группировать требования
- Использовать логику предикатов, чтобы не запрашивать ненужные данные
- Использовать блокирование данных
- По возможности использовать статические операторы SQL

Составные операторы SQL и хранимые процедуры

Для программ, посылающих и получающих много команд и ответов, затраты, связанные с передачей по сети, бывают достаточно большими. Составные операторы SQL и хранимые процедуры позволяют их снизить.

Если программа посылает несколько операторов SQL подряд, можно использовать составной оператор SQL. Если в группе операторов SQL должны применяться логические программные конструкции, можно использовать хранимые процедуры.

В составной оператор SQL могут входить все исполняемые операторы, кроме следующих:

```
CALL
FETCH
CLOSE
OPEN
Составные операторы SQL
Connect
Prepare
Release
Describe
Rollback
Disconnect
Set connection
execute immediate
```

Более подробную информацию смотрите в руководстве *SQL Reference*.

Использование составных операторов SQL в программе описано в разделе “Составной SQL NOT ATOMIC” на стр. 53. Использование составных

операторов SQL с утилитой импорта описано в разделе “Использование утилит импорта и экспорта” на стр. 110.

Хранимые процедуры помогают уменьшить сетевой трафик за счет размещения текста программы на сервере. В версиях DB2 до версии 5.0 хранимые процедуры возвращали только выходные параметры, так что программа должна была посылать отдельную команду для принятия, лишняя раз используя сеть. В версии DB2 5.0 и последующих принятие можно производить автоматически при выходе из процедуры. Кроме того, можно возвращать наборы результатов, что сводит к минимуму текст программы на клиенте.

Использование хранимых процедур описано в разделе “Хранимые процедуры” на стр. 51.

Группировка требований

Группировка связанных требований к базе данных (операторов SQL) в одно требование сокращает число передаваемых по сети требований и ответов. Например, если сгруппировать следующие операторы:

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1  
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=2
```

в

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1 OR ROW_ID=2
```

по сети будет передано меньше требований.

Можно также сократить число возвращаемых строк с помощью ключевых слов, таких как IN и BETWEEN. Кроме того, можно использовать ключевые слова WHERE, IN и BETWEEN с операторами UPDATE и DELETE.

Логика предикатов

Логика предикатов позволяет не требовать ненужных строк и столбцов. Это сводит к минимуму сетевой трафик и нагрузку на процессор при передаче данных.

Например, не используйте запрос:

```
SELECT * FROM TABLEA
```

если на самом деле вам требуется только первая строка TABLEA с ROW_ID=1 или только столбцы 1 и 2.

Блокирование данных

Блокирование данных удобно, если с сервера приходят большие объемы данных. Оно позволяет максимально использовать пропускную способность сети и уменьшает нагрузку на процессоры как сервера баз данных хоста или AS/400, так и рабочей станции DB2 Connect.

Прием и отправка любого сообщения, независимо от его объема, означает определенную нагрузку на сеть и процессор. Блокирование данных сокращает количество сообщений, сохраняя объем передаваемой информации.

При блокировании первая строка данных из запроса не будет передана программе до завершения первого блока. Это увеличивает время поиска первой строки, но уменьшает его для последующих строк.

Следует обратить внимание на использование памяти. Когда включено блокирование, потребности в памяти обычно увеличиваются. Подробный анализ блокирования при использовании соединений SNA смотрите в руководстве *DRDA Connectivity Guide*.

Внутри DB2 Connect можно управлять объемом данных, передаваемых в каждом блоке, как описывается в разделе “RQRIOBLK” на стр. 138.

Блокирование включается опцией BLOCKING в командах rper и bind. (Подробности смотрите в разделе “Команда BIND” на стр. 94.) Блокирование используется, если:

- Указатель является указателем только для чтения или
- Указатель неоднозначен, а в команде rper или bind задано блокирование.

Определение указателя только для чтения, изменяемого и неоднозначного указателя смотрите в руководстве *Application Development Guide*.

Примечание: При работе с динамическими операторами SQL указатель всегда является неоднозначным.

Операторы SQL с опцией BLOCKING

Операторы SELECT с возможностью изменения (использующие операторы UPDATE/DELETE WHERE CURRENT OF) не допускают блокирования, поэтому не используйте их без крайней необходимости.

Для оператора SELECT с возможностью изменения надо обеспечить, чтобы строка не изменилась от момента завершения SELECT до выполнения UPDATE/DELETE. Если такой уровень согласованности необязателен, можно вместо этого использовать DELETE или UPDATE с критериями поиска, основанными на данных, которые вернул оператор SELECT без возможности изменения.

Оператор SELECT без возможности изменения задается опцией FOR FETCH ONLY (кроме систем VM и VSE, где это не поддерживается).

Статические и динамические операторы SQL

По возможности используйте статические операторы SQL. Это позволит избежать подготовки разделов SQL во время выполнения и неоднозначных указателей. Если без динамических операторов не обойтись, можно уменьшить сетевой трафик и улучшить производительность следующим образом:

- Если используется оператор SELECT, который должен быть подготовлен, выполните `PREPARE ... INTO SQLDA`. SQLDA следует выделить полный объем, необходимый в ваших условиях. Если максимальное число столбцов равно x и не будет меняться, разместите SQLDA с x переменными SQLVAR. Если возможное число столбцов неизвестно (а памяти достаточно), используйте максимальное количество SQLVAR (256).

Если область SQLDA недостаточна для возвращаемых SQLDA, программе придется еще раз послать DESCRIBE с достаточными SQLDA, чтобы заново сохранить результат. Это увеличивает сетевой трафик.

Не используйте последовательности PREPARE и DESCRIBE. Оператор `PREPARE...INTO` дает лучшую производительность.

- Выполняйте статически связанные операторы SQL COMMIT или ROLLBACK вместо динамических COMMIT или ROLLBACK.
- Прочие операторы (кроме SELECT, COMMIT и ROLLBACK) выполняйте с помощью EXECUTE IMMEDIATE, а не с помощью последовательности PREPARE и EXECUTE.
- Программы ODBC используют динамический SQL. Для повышения производительности можно использовать возможность статического профилирования CLI/ODBC. Эта возможность позволяет перехватывать вызовы ODBC и преобразовывать их в статические операторы, хранимые в пакете баз данных. Реальная полученная производительность зависит от сложности вашей программы. Дополнительную информацию смотрите в разделе *CLI Guide and Reference*.

Другие особенности SQL

Процессор командной строки в целом работает медленнее динамических операторов SQL, поскольку он анализирует операторы, прежде чем передать их базе данных. Кроме того, он форматирует данные при получении, что может быть излишним для вашей программы.

Операторы SQL в интерпретируемом языке (например, REXX) выполняются существенно медленнее, чем те же операторы в компилируемом языке (например, C).

Операторы CONNECT бывают двух типов: тип 1 и тип 2. Оператор типа 2, создавая соединение с базой данных, переводит предыдущее соединение в неактивное состояние, но не удаляет его. Переключаясь позже на неактивное соединение, не надо снова загружать библиотеки и восстанавливать

внутренние структуры данных. Поэтому CONNECT типа 2 может повысить производительность программ, которые обращаются к нескольким базам данных. Подробную информацию об операторах CONNECT типа 2 смотрите в книгах *Administration Guide* и *SQL Reference*.

Настройка DB2 Connect

Для настройки DB2 Connect можно изменять различные параметры в файле конфигурации менеджера баз данных. Изменение этих параметров описано в руководстве *Administration Guide*.

RQRIOBLK

Параметр RQRIOBLK задает максимальный размер блоков ввода/вывода в сети. Большой размер блока может улучшить производительность для больших требований. Размер блока обычно не влияет на время ответа для небольших требований, для которых возвращается одна строка.

Большой размер блока обычно требует больше памяти на рабочей станции DB2 Connect. Это увеличивает размер рабочего набора и на слабых рабочих станциях может вызвать частую подкачку.

Используйте размер блока DRDA по умолчанию (32767), если это не ведет к подкачке при выполнении вашей программы. Иначе надо уменьшать размер блока, пока подкачка не прекратится. Подкачка вызывает существенное снижение производительности. Используйте программы для определения производительности (например, программу *vmstat* для систем на основе UNIX или *SPM/2* для OS/2), чтобы установить, идет ли подкачка на вашей системе. Другие средства описаны в разделе “Средства определения производительности” на стр. 131.

DIR_CACHE

Параметр DIR_CACHE определяет, сохраняется ли информация каталога. Если она сохраняется (DIR_CACHE=YES), файлы каталогов считываются и кэшируются в памяти, что позволяет не создавать внутренней структуры каталогов и не считывать файлы каталогов при каждом соединении.

Если эта информация не сохраняется (DIR_CACHE=NO), при каждом соединении с базой данных соответствующий каталог считывается с диска, после чего производится поиск. Когда требуемые записи найдены, вся память, отведенная для поиска в каталогах, освобождается.

Если сведения каталогов сохраняются, общий кэш каталогов создается при запуске **db2start** и освобождается при остановке DB2. Он используется всеми процессами сервера DB2 (агентами DB2). Кроме того, когда программа устанавливает первое соединение с базой данных, строится ее собственный кэш каталогов, который освобождается с завершением программы.

Каждый кэш содержит образ каталога системной базы данных, каталога служб соединений и каталога узла. При этом установить соединение становится легче, поскольку не требуются обращения к файлам каталогов и поиск происходит быстрее.

Когда каталог, хранящийся в кэше, исправляют, изменения не сразу переносятся в кэш. Если запись в каталоге не найдена в кэше, производится поиск в исходном каталоге.

Сохранение каталогов требует большей собственной памяти для работы программы. Без использования кэша эта память нужна только для поиска в каталоге. Кэши несколько увеличивают нужный DB2 объем совместно используемой памяти, поскольку туда переносится общая для агентов баз данных информация о каталогах. Объем памяти, занимаемой кэшем, зависит от числа записей в каждом каталоге.

Другие параметры DB2 Connect

Если на рабочей станции DB2 Connect нет локальной базы данных, параметры MAXDARI и NUMDB должны иметь минимальные значения. Это уменьшит потребляемые ресурсы.

AGENTPRI применяется только при наличии удаленных клиентов. Этот параметр отвечает за приоритет, назначаемый планировщиком операционной системы агентам экземпляра DB2 Connect. Экземпляр DB2 Connect, имеющий больший приоритет (меньший номер), получает больше процессорного времени. Поэтому для выполнения других действий на рабочей станции DB2 Connect остается меньше процессорного времени. Например, на одной рабочей станции можно запустить два экземпляра DB2 Connect, с высоким и с низким приоритетом (то есть с разными значениями AGENTPRI).

Каждое соединение компьютера клиента с сервером баз данных хоста или AS/400 через DB2 Connect требует работы агента на рабочей станции DB2 Connect. Задайте для MAXAGENTS значение не меньше наибольшего числа соединений с удаленными клиентами, обращающихся к серверу баз данных хоста или AS/400 через рабочую станцию DB2 Connect.

Если вы хотите использовать учетные строки, API sqlesact() дает лучшую производительность, чем использование переменной среды DB2ACCOUNT. Дополнительную информацию смотрите в разделе “Реализация учета оплаты в DB2 Universal Database for OS/390” на стр. 56.

Если вам не нужен пользовательский файл отображения SQLCODE, можно повысить производительность, используя файл отображений по умолчанию или отключив отображение SQLCODE. (Файл отображения по умолчанию включен в библиотеку DB2 Connect; пользовательский файл надо считывать с диска, что снижает производительность.) Подробную информацию об отображении SQLCODE смотрите в разделе Глава 11, “Отображение SQLCODE” на стр. 123.

Поддержка пула соединений

Часто серверы DB2 Connect Enterprise Edition обеспечивают соединения с базой данных для тысяч одновременных требований клиентов. Сервер баз данных может тратить очень много ресурсов на установление и поддержку соединений, что сильно ухудшает производительность сервера баз данных и сервера DB2 Connect. Это особенно существенно в среде Web, где для обращения к Web-странице может потребоваться создать новое соединение с сервером баз данных, выполнить запрос и завершить это соединение. В DB2 Connect Enterprise Edition для сокращения этих затрат ресурсов поддерживается *пул соединений*, в котором сохраняются уже открытые соединения с базой данных.

Как работает пул соединений

Пул соединений прозрачен для прикладных программ, соединяющихся с хостом через DB2 Connect. Когда прикладная программа запрашивает отключение от хоста, DB2 Connect завершает входящее соединение с этой прикладной программой, но сохраняет исходящее соединение с хостом в пуле открытых соединений. Когда новая прикладная программа запрашивает соединение, DB2 Connect использует уже открытое соединение из пула соединений. Использование таких постоянно открытых соединений уменьшает общее время соединения, а также затраты времени процессора хоста.

Чтобы использовать пул соединений, к DB2 для OS/390 Версии 6.1 нужно применить следующий APAR:

APAR PQ33473

Агенты DB2 Connect могут быть в одном из двух состояний: в состоянии бездействия или в активном состоянии. Агент активен, когда он выполняет работу для какой-либо прикладной программы. Когда эта работа закончена, агент переходит в состояние бездействия и ожидает дальнейших запросов на обработку от этой же или от другой прикладной программы. Все бездействующие агенты сохраняются в пуле бездействующих агентов. Размер этого пула можно задать при помощи параметра конфигурации NUM_POOLAGENTS. Этот параметр определяет максимальное число бездействующих агентов, поддерживаемых системой. Если для этого параметра задано нулевое значение, пул соединений не будет использоваться.

DB2 Connect не устанавливает соединения с базой данных, пока не получит первое требование от клиента. Однако, если нужно, пул бездействующих агентов можно формировать до получения требований от клиентов. Для этого используется параметр конфигурации NUM_INITAGENTS. Он задает число бездействующих агентов, которые должны создаваться при запуске DB2 Connect. Эти бездействующие агенты изначально не будут иметь соединений с сервером баз данных хоста.

Когда клиент запрашивает соединение с хостом, DB2 Connect пытается найти в пуле агент, имеющий соединение с сервером баз данных хоста. Если такой агент не найден, DB2 Connect пытается найти доступный агент в пуле бездействующих агентов. Если пул пуст, DB2 Connect создаст новый агент.

Используя параметр конфигурации MAX_COORDAGENTS, можно задать максимальное число одновременно активных агентов. При превышении этого числа агентов для новых соединений будет выдаваться код ошибки SQL1226. (Этот код означает, что превышено максимальное число одновременных исходящих соединений.)

Переменная реестра DB2 DB2CONNECT_IN_APP_PROCESS позволяет задать режим работы для прикладных программ, выполняемых на том же компьютере, что и DB2 Connect EE – для таких прикладных программ DB2 Connect может выполняться внутри процесса прикладной программы (поведение по умолчанию) или же эти прикладные программы могут использовать соединение с сервером DB2 Connect EE, а для соединения с хостом – агент DB2 Connect. Чтобы прикладные программы использовали пул соединений, для соединений с хостом должны использоваться агенты сервера DB2 Connect EE; в этом случае для переменной DB2CONNECT_IN_APP_PROCESS нужно задать значение NO.

Концентратор соединений DB2 Connect

Концентратор соединений DB2 Connect позволяет серверам DB2 Connect Enterprise Edition поддерживать тысячи одновременных коммерческих транзакций, значительно сокращая затраты ресурсов на серверах баз данных хоста S/390 или AS/400. Потоки данных для всех прикладных программ передаются через существенно меньшее число соединений с сервером баз данных хоста S/390 или AS/400. Хотя это может показаться похожим на работу описанного выше пула соединений, на самом деле это более сложный подход, позволяющий снизить потребление ресурсов для прикладных программ, требующих обработки очень большого числа диалоговых транзакций.

Пул соединений позволяет избежать затрат на установление соединения, если уже есть соединение, освобожденное завершившейся прикладной программой. Иначе говоря, одна прикладная программа должна закончить использование соединения, прежде чем другая программа сможет повторно использовать соединение из пула соединений.

Напротив, концентратор соединений позволяет DB2 Connect использовать соединение для прикладной программы, как только другая прикладная программа завершит транзакцию; при этом не требуется, чтобы эта другая прикладная программа завершила соединение. В итоге соединение с сервером баз данных и связанные с этим соединением ресурсы хоста и DB2 Connect используются прикладной программой только во время транзакции. Как только транзакция завершена, соединение и связанные с ним ресурсы могут

использоваться другой прикладной программой, требующей выполнения транзакции.

Как реализован концентратор соединений

В предыдущих версиях DB2 Connect каждая активная прикладная программа использовала отдельный блок Engine Dispatchable Unit (EDU), управляющий соединением с базой данных и всеми требованиями прикладной программы. Этот блок обычно называют *координирующим агентом*. Каждый координирующий агент отслеживает состояние и контекст прикладной программы и EDU. При увеличении числа соединений возрастает объем памяти, используемый всеми EDU, а переключение контекстов между агентами требует дополнительных затрат ресурсов.

В описанной выше архитектуре для каждого соединения используется свой EDU. Но концентратор соединений позволяет использовать один блок EDU для многих соединений. Таким образом, используемое число блоков EDU меньше числа соединений.

Концентратор соединений разделяет агент на два уровня – на *логический агент* и *рабочий агент*. Логический агент представляет какую-либо прикладную программу, но не указывает на конкретный блок EDU. Логический агент содержит все информационные и управляющие блоки, необходимые прикладной программе. Если с сервером соединены *n* прикладных программ, на сервере будет *n* логических агентов. Рабочие агенты – это физические блоки EDU, выполняющие требования прикладных программ, но не подключенные постоянно к какой-либо конкретной прикладной программе. Рабочие агенты связываются с логическими агентами только на время выполнения транзакции и после ее завершения возвращаются в пул доступных агентов.

Связи между рабочими агентами и логическими агентами назначаются *планировщиком логических объектов*. Существующие на некоторых платформах ограничения на число открытых файлов могут вызвать запуск нескольких экземпляров планировщика, если число логических агентов превышает максимально допустимое число открытых файлов.

Активирование концентратора соединений

Чтобы использовать концентратор соединений, к DB2 для OS/390 Версии 6.1 нужно применить следующий APAR:

APAR PQ33473

Параметр конфигурации менеджера баз данных MAX_LOGICAGENTS задает максимальное число логических агентов. Чтобы активировать использование концентратора, задайте для параметра MAX_LOGICAGENTS значение, большее его значения по умолчанию. Значение по умолчанию параметра MAX_LOGICAGENTS равно значению параметра MAX_COORDAGENTS.

Поскольку для каждой прикладной программы используется один логический агент, параметр `MAX_LOGICAGENTS` определяет число прикладных программ, которые могут соединиться с экземпляром базы данных, в то время как параметр `MAX_COORDAGENTS` определяет максимальное число одновременно активных исходящих соединений. Для параметра `MAX_LOGICAGENTS` можно задать значение в диапазоне от значения `MAX_COORDAGENTS` до 64000. Число логических агентов по умолчанию равно числу `MAX_COORDAGENTS`.

Некоторые из параметров конфигурации используются для конфигурирования агентов:

MAXAGENTS

Максимальное число рабочих агентов.

MAX_COORDAGENTS

Максимальное число активных координирующих агентов.

NUM_POOLAGENTS

Размер пула агентов. В этот пул входят неактивные агенты и бездействующие агенты.

NUM_INITAGENTS

Исходное число рабочих агентов в пуле. Это будут бездействующие агенты.

Поддержка транзакций XA

Архитектура концентратора соединений позволяет DB2 Connect поддерживать тесно связанные транзакции XA с DB2 для OS/390 и DB2 для AS/400. Концентратор связывает рабочий агент с конкретной транзакцией XA (одним XID) так же, как он это делает для любой другой транзакции. Однако, если эта транзакция XA заканчивается командой `xa_end()` (конец ветвления), рабочий агент не возвращается в общий пул свободных агентов. Вместо этого рабочий агент остается связанным с этой конкретной транзакцией XA. Когда другая прикладная программа присоединяется к этой транзакции XA, рабочий агент подключается к этой прикладной программе.

Все команды конца транзакции возвращают агент в нормальный пул. Например, `xa_prepare()` в режиме только для чтения, `xa_rollback()`, `xa_recover()`, `xa_forget()`, `xa_commit()` или любая ошибка XA, вызывающая откат, вернут агента в нормальный пул. Сам по себе вызов `xa_end()` приводит только к завершению ветвления транзакции, но его недостаточно для завершения связи агента с этим XID.

Примеры

1. Рассмотрим ситуацию, когда требуются 4000 или более одновременных соединений. Это характерно для сервера Web, использующего прикладные программы CGI, или корпоративной системы с большим числом

пользователей, работающих за настольными компьютерами. В таких случаях для обеспечения эффективной работы обычно требуется, чтобы DB2 Connect работал как автономный сервер, то есть чтобы база данных и система DB2 Connect находились на отдельных компьютерах.

Система сервера DB2 Connect может быть не в состоянии поддерживать 4000 одновременно открытых соединений с компьютером базы данных. В большинстве случаев число выполняющихся в данный момент транзакций будет значительно меньше числа одновременных соединений.

Администратор системы может повысить эффективность, задав следующие значения параметров конфигурации базы данных:

```
MAX_LOGICAGENTS = 4000
MAX_AGENTS      = 1000
MAX_COORDAGENTS = 1000
NUM_POOLAGENTS  = 1000
```

Концентратор будет поддерживать до 4000 одновременных сеансов, в то время как шлюз будет одновременно поддерживать только 1000 транзакций.

2. В этом примере рабочие агенты будут постоянно образовывать и прекращать связи с логическими агентами. Агенты, которые не активны (но не бездействуют), поддерживают соединение с базой данных, но не принимают участия в транзакциях, поэтому они могут использоваться любым логическим агентом (прикладной программой), запрашивающим соединения.

Для транзакций XA есть некоторые отличия. В описанной выше ситуации мы можем предположить, что вместе со шлюзом DB2 Connect и базой данных OS/390 или AS/400 используется монитор транзакций. Когда прикладная программа запрашивает соединение, концентратор использует для обслуживания этого запроса неактивный агент или создает новый рабочий агент. Теперь предположим, что прикладная программа запрашивает транзакцию XA. Для этой транзакции создается XID и с ним связывается рабочий агент.

Когда запрос прикладной программы выполнен, она выдает команду `xa_end()` и отключается от рабочего агента. Рабочий агент остается связанным с XID этой транзакции. Теперь он может обслуживать только запросы для транзакций с этим XID.

В этом момент другая прикладная программа может запросить транзакцию другого типа (не XA). Даже если нет других доступных рабочих агентов, данный агент, связанный с этим XID, не будет доступен для второй прикладной программы. Он считается активным. Для второй прикладной программы будет создан новый рабочий агент. Когда эта вторая прикладная программа закончит транзакцию, ее рабочий агент будет освобожден и помещен в пул доступных агентов.

Тем временем другие прикладные программы, запрашивающие транзакцию связанную с XID первого агента, могут подключаться к этому агенту и выполнять эту транзакцию XA. Все запросы прикладных программ для этой конкретной транзакции будут передаваться рабочему агенту этой транзакции, если он свободен.

Рабочий агент будет возвращен в общий пул, только когда прикладная программа выдаст команду конца транзакции (кроме `xa_end()`). Например, прикладная программа может завершить транзакцию при помощи `xa_commit()`, после чего рабочий агент более не будет связан с данным XID и вернется в пул доступных агентов. После этого любые прикладные программы могут использовать этот рабочий агент для других транзакций XA или для транзакций других типов (не XA).

Ограничения

Есть ряд важных ограничений на использование концентратора шлюза. Внимательно изучите следующую ниже информацию, прежде чем пытаться использовать концентратор соединений в вашей системе.

- Концентратор соединений может использоваться только клиентами DB2 Версии 7 или более новых версий.
- Концентратор поддерживает только хосты DB2 for OS/390 и DB2 for AS/400.
- Концентратор использует протокол TCP/IP для входящих соединений от локальных и удаленных клиентов. Только входящие соединения, использующие TCP/IP или локальное соединение (IPC), смогут использовать пул исходящих соединений. Концентратор может работать с соединениями через другие протоколы связи (например, SNA), но для таких соединений нельзя использовать возможности концентрации XA.
- Если на шлюзе работает концентратор, в клиентских программах не следует использовать статический оператор SET. Если используется этот оператор, DB2 не возвращает сообщения об ошибке, однако это может плохо повлиять на работу этой прикладной программы и всех других прикладных программ, использующих то же исходящее соединение.
- Поддерживается только немедленное выполнение операторов SET.
- Если в программе объявляются глобальные временные таблицы, их нужно явно закрыть в конце транзакции или ветвления. Ошибка при закрытии этих таблиц может вызвать ошибку при последующих транзакциях.
- Для поддержки тесно связанных транзакций XA все прикладные программы, принимающие участие в одной транзакции XA, должны для соединения с хостом использовать один и тот же шлюз.
- Выигрыш от использования концентратора получается только для прикладных программ, закрывающих указатели WITH HOLD в конце транзакции. Транзакции, в которых не закрываются указатели WITH

HOLD, будут выполняться, но для каждой из них будет использоваться отдельный рабочий агент и поэтому нельзя будет использовать все возможности концентратора.

- Все прикладные программы, принимающие участие в одной транзакции XA, должны иметь тот же CCSID и использовать для соединения один и тот же ID пользователя.
- Если исходящее соединение было установлено для поддержки двухфазного соединения, агент этого соединения может использоваться только для двухфазных соединений. Аналогично агенты, соединение которых было установлено для поддержки однофазных соединений, могут использоваться только для однофазных соединений.
- Аутентификация типа DCS_ENCRYPT не будет работать с концентратором в DB2 Версии 7.
- Концентратор поддерживает динамический SQL только для интерфейса командной строки. Запросы подготовки динамических операторов SQL из прикладных программ с встроенным SQL не будут выполнены. Нужно изменить прикладные программы, чтобы использовать в них статический SQL или вызывать динамические операторы SQL через интерфейс командной строки.

Настройка баз данных

Производительность системы зависит от производительности базы данных сервера хоста или AS/400.

Разные СУБД имеют свои особенности производительности. Например, оптимизаторы SQL разных систем могут по-разному работать с одной и той же программой. Чтобы узнать подробнее о производительности системы сервера баз данных хоста или AS/400, обратитесь к документации по ним.

Производительность DB2 Universal Database for AS/400 можно повысить с помощью опций связывания "чтение неприятого" (UR) или "без принятия" (NC), которые отменяют запись в журнал.

Примечание: UR позволяет только считывать не занесенные в журнал данные, но не изменять их, причем только если блокирование имеет значение ALL.

В зависимости от сервера прикладных программ и определяемой им степени детализации блокировок, уровень изоляции, используемый для запроса или программы, может существенно влиять на производительность.

База данных должна иметь подходящий уровень нормализации, эффективные индексы и достаточно выделенного места. На производительность также влияют используемые типы данных, что описано в следующих разделах.

Настройка DB2 for OS/390

Для поддержки TCP/IP требуется, как минимум, OS/390 V1R3. Настоятельно рекомендуется использовать OS/390 V2R5 или более новую.

За соединение распределенных программ с DB2 for OS/390 отвечает утилита DDF. DDF надо установить как сервер прикладных программ. Для этого можно либо занести имя LU удаленной системы в таблицу SYSIBM.LUNAMES, либо занести значения LUNAME, SYSMODENAME, USERSECURITY, ENCRYPTPSWDS, MODESELECT и USERNAMES в таблицу SYSIBM.SYSLUNAME. Затем измените загрузочный набор данных (BSDS) DDF, например:

```
DDF LOCATION=LOC1,LUNAME=LU1,PORT=8000,RESPORT=8001
```

Лучшую производительность дает рекомендуемый приоритет адресного пространства DDF (чуть меньший или равный DBM1 для режима COMPAT). Используйте кэширование RACF авторизации в VLF и, по возможности, кэширование авторизаций пакета версии 5. Для большинства операций достаточно значения CACHEPAC=32768.

DDF пытается установить соединение с VTAM, поэтому VTAM должен быть активен во время запуска DDF. Ниже приводится пример определения VTAM APPL:

```
SYD51TC* APPL AUTH=(ACQ), X
PARSESS=YES, X
HAVAIL=YES, X
EAS=1600, X
APPC=YES, X
DSESLIM=1024, X
DMINWNL=512, X
DMINWNR=512, X
AUTOSES=1, X
SECACPT=ALREADYV, X
SRBEXIT=YES, X
SYNCLVL=SYNCPT, X
MODETAB=DB2MODET, X
VPACING=63 X
```

В OS/390 можно оптимизировать обработку остановленных потоков. В Версии 3 допускается более 10000 одновременно соединенных клиентов, а в версиях 4 и 5 – до 25000. Впрочем, независимо от версии, одновременно активны могут быть не более 1999. Соединения со всеми клиентами рабочей станции могут сохраняться в неактивном состоянии; их потоки помещаются в неактивную цепочку при каждом принятии.

Параметры DSNZPARM CMTSTAT, CONDBAT и MAXDBAT влияют на обработку потоков. Наилучшую производительность дает значение CMTSTAT = INACTIVE и значение CONDBAT, равное максимальному числу соединений с DBAT, и значение MAXDBAT, равное максимальному возможному числу активных DBAT.

Подробное обсуждение соединений DB2 for OS/390 в сети DRDA, включая конфигурирование VTAM, смотрите в книге *Дополнение по возможностям соединений*.

Преобразование данных

При передаче данных из одной среды в другую иногда требуется их преобразование. Оно может влиять на производительность.

Рассмотрите следующие платформы:

- Intel (OS/2, Windows NT или Windows 2000)
- IEEE (системы на основе UNIX)
- System/370 и System/390 (MVS, OS/390, VM и VSE)
- OS/400

и следующие типы числовых данных:

- Упакованный десятичный
- Зонный десятичный
- Целое
- С плавающей точкой

В Табл. 8 показано, когда происходит преобразование.

Таблица 8 (Стр. 1 из 2). Преобразование данных

	Intel	IEEE	S/370 и S/390	OS/400
Упакованные десятичные данные				
Intel	Нет	Нет	Нет	Нет
IEEE	Нет	Нет	Нет	Нет
S/370/390	Нет	Нет	Нет	Нет
OS/400	Нет	Нет	Нет	Нет
Зонные десятичные данные				
Intel	Нет	Нет	Да	Да
IEEE	Нет	Нет	Да	Да
S/370/390	Да	Да	Нет	Нет
OS/400	Да	Да	Нет	Нет
Целые данные				
Intel	Нет	Да	Да	Да
IEEE	Да	Нет	Нет	Нет
S/370/390	Да	Нет	Нет	Нет
OS/400	Да	Нет	Нет	Нет
Данные с плавающей точкой				

Таблица 8 (Стр. 2 из 2). Преобразование данных

	Intel	IEEE	S/370 и S/390	OS/400
Intel	Нет	Да	Да	Да
IEEE	Да	Нет	Да	Нет
S/370/390	Да	Да	Нет	Да
OS/400	Да	Нет	Да	Нет

Затраты на преобразование однобайтных символьных данных обычно меньше, чем для числовых (где преобразование необходимо).

Затраты на преобразование даты DATE/TIME/TIMESTAMP почти такие же, как для однобайтного символа CHAR. Дороже всего обходится преобразование типа FLOATING с плавающей точкой. Учитывая это, можно повысить производительность программы для DB2 Connect при ее разработке.

Если в таблице базы данных есть столбец, определенный как 'FOR BIT DATA', данные, которыми обмениваются программа и база данных, не требуют преобразования. Это можно использовать при архивации данных на сервере баз данных хоста или AS/400.

Типы символьных данных

Символьные данные имеют тип CHAR или VARCHAR. Их использование зависит от типичной длины данных в поле:

- Если длина данных значительно меняется, лучше использовать VARCHAR, поскольку CHAR заполняет свободную часть поля пробелами, которые затем пересылаются по сети наравне с другими символами.
- Если длина данных примерно постоянная, лучше использовать CHAR, поскольку каждое поле VARCHAR содержит несколько байтов сведений о длине, которые необходимо пересылать.

Настройка сети

Общую производительность в среде распределенных баз данных лучше всего повысить, устранив задержки в сети. Администраторы сети часто считают сеть эффективной, если для передачи собирается максимально возможный объем данных. Такой подход не оправдан при работе с распределенными базами данных и подобными программами, поскольку он заранее предполагает задержки. Конечный пользователь видит только эти задержки и не обращает внимание на эффективность сети.

Большинство сетевых устройств имеют параметры задержки, и большинство их значений по умолчанию неудобны для распределенных баз данных. Чтобы повысить производительность, следует найти параметры задержки и по возможности задать для них нулевые значения. Кроме того, следует убедиться, что буферы устройств достаточны, чтобы избежать повторной

передачи из-за потери данных. Например, обычная длина очереди передачи или приема в системе UNIX по умолчанию равна 32. Лучшие результаты дает длина очереди, равная 150. Соответствующий параметр в настройках DLC (длина очереди приема) также должен иметь значение 150.

Значение параметра IOBUF обычно слишком мало. Обычно задают 500, но опыт показывает, что при работе с большими объемами данных, особенно для канальных соединений, таких как ESCON или 3172, оптимальным является значение 3992.

Для соединений SNA значение параметра Mode Profile любой программы рабочей станции должно равняться 63. В целом значения синхронизации связи для приема в сети должны быть как можно большими, поэтому параметры VPACING и PACING в операторе DB2 APPL и PU/LU для рабочей станции в переключении главного режима должны также иметь значение 63. Это позволит пропорционально увеличить объем потока сообщений, после которого отправитель должен ждать ответа.

В локальной сети или LLC размер окон приема и передачи может существенно влиять на производительность. Размер окна передачи должен быть не меньше семи, а наилучший размер окна приема в большинстве конфигураций – не больше четырех.

Если вы используете Ethernet, задайте размер сегмента TCP 1500 байт. В сети Token Ring или FDDI это значение должно быть 4400 байт, а для адаптера ESCON с TCP/IP размер сегмента должен всегда равняться 4096.

Наконец, в сетях TCP/IP размер буферов приема и передачи TCP должен быть не менее 32768. Как правило, оптимальное значение – 65536.

Примечание: Установить соединение с шлюза на сервер (исходящее соединение) сложнее, чем соединение с клиента на шлюз (входящее соединение). В среде, где тысячи клиентов постоянно устанавливают и разрывают соединения с сервером через шлюз, значительная часть времени тратится на установление исходящих соединений. DB2 Connect позволяет создавать пул соединений с TCP/IP. Когда клиент разрывает соединение с сервером, шлюз удаляет только входящее соединение с этим клиентом, а исходящее, с сервером, помещает в пул. Когда новый клиент требует установить соединение, шлюз использует существующее соединение из пула, что экономит время и снижает нагрузку на процессор на сервере.

Подробную информацию о создании пула соединений в DB2 смотрите в руководстве *Administration Guide*.

В следующей таблице кратко изложены методы настройки производительности сети.

На что обратить внимание	Пример	Значение	Примечание
Принудительные задержки	Параметры задержки у сетевых устройств	Задать равными 0.	Значения по умолчанию обычно ненулевые.
Буферы	Параметр IOBUF	Увеличить до 3992.	Особенно эффективно для ESCON и других канальных адаптеров.
	RUSIZE	Оптимальный размер – 4096.	Иногда наилучшую производительность дает одинаковый размер RUSIZE и RQRIOBK.
	Синхронизация связи	VPACING, PACING и Mode Profiles должны иметь значение 63.	По возможности используйте адаптивную синхронизацию.
Настройки адаптера	Длина очереди приема/передачи	Рекомендуется 150.	Длина по умолчанию обычно 32.
	Управление окнами DLC на SNA	Задайте большой размер окна передачи (>7), маленький размер окна приема (например, 1), проверьте и подберите наилучшее сочетание.	Логические устройства создают дополнительные задержки. Топология сети должна быть максимально простой.
Настройки TCP	Размер сегментов	1500 для Ethernet, 4400 для Token Ring и FDDI.	Для адаптеров ESCON в сети TCP/IP следует задавать значение 4096.
	Размер областей приема/передачи	Обе области должны иметь размер 64 Кбайта.	Значение по умолчанию для Windows всего 8192. Это значение можно задать через реестр Windows.

Сетевые устройства

Следует учитывать следующие особенности аппаратуры:

- Скорость сети или среды передачи.

Высокопроизводительная среда передачи повышает производительность. Ниже приводятся типичные скорости передачи необработанных данных:

"Канал–канал" (оптоволоконный кабель)

4,0 Мбайт/с

Локальная сеть 16 Мбит/с

2,0 Мбайт/с

"Канал–канал" (обычный)

1,0 Мбайт/с

Локальная сеть 4 Мбит/с

0,5 Мбайт/с

Высокоскоростной канал T1 (1,544 Мбит/с)

0,193 Мбайт/с

Быстрая телефонная линия 56 Кбит/с

0,007 Мбайт/с

Модем 19,6 Кбит/с

0,002 Мбайт/с

Модем 9600 бит/с

0,001 Мбайт/с

Скорость передачи данных ограничена скоростью самой медленной среды передачи на пути к хосту или серверу баз данных AS/400.

- Сетевой адаптер или контроллер связи

Следует обратить внимание на то, как сетевой адаптер и контроллер связи используют память. Кроме того, надо с помощью специалиста по сетям убедиться, что контроллер способен справиться с дополнительным трафиком, создаваемым DB2 Connect.

- Топология сети

Если данные передаются с одной локальной сети на другую и с одной сети SNA на другую, обратите внимание на время передачи. На прохождение мостов, маршрутизаторов и шлюзов уходит дополнительное время. Например, чем меньше количество используемых мостов, тем меньше ретрансляций нужно для каждого требования.

Важно также физическое расстояние между узлами. Даже при спутниковой передаче сообщения скорость передачи ограничена скоростью света ($3 * 10^8$ м/с) и расстоянием между отправителем и получателем.

- Сетевой трафик

Если диапазон сети используется полностью, снижается и время ответа, и скорость передачи данных для отдельной программы.

Если данные скапливаются на определенном участке сети, может возникнуть перегрузка; таким участком может быть, например, старый NCP с маленьким буфером.

- Надежность сети

Если в сети часто возникают ошибки, ее пропускная способность падает и общая производительность снижается из-за повторной передачи данных.

Конкуренция за системные ресурсы

Производительность снижается, когда несколько заданий одновременно претендуют на системные ресурсы. Изучите следующие вопросы:

- Не перегружен ли процессор? Возможно, следует обновить систему, уменьшить объем работы и настроить систему, чтобы снизить нагрузку.
- Не перегружена ли память? Возможно, следует нарастить память, уменьшить объем работы и настроить систему, чтобы уменьшить рабочую область памяти.
- Не слишком ли занят адаптер/контроллер связи? Возможно, следует обновить сеть или использовать парные платы Token Ring.
- Не перегружена ли одна из подсистем? Не проходит ли через нее маршрут передачи данных?
- Не выполняются ли в системе посторонние процессы и задачи? Общее правило состоит в том, чтобы конфигурировать и запускать только постоянно используемые службы, чтобы не тратить лишних ресурсов.
- Не использует ли малое число процессов или заданий большую часть ресурсов? Нельзя ли их остановить? Нельзя ли снизить их приоритет? Можно ли снизить использование ими ресурсов?

Поиск причин низкой производительности

Если пользователи DB2 Connect при больших запросах долго ожидают ответа от серверов хоста или AS/400, причину низкой производительности может искать в следующих областях:

1. Для запросов, для которых с сервера хоста или AS/400 возвращаются большие блоки данных (обычно от 32 Кбайт и выше), установите для параметра конфигурации менеджера баз данных RQRI0BLK значение 32767. Это можно сделать, введя в командной строке команду:

```
db2 update database manager configuration using RQRI0BLK 32767
```
2. Если в соединениях с сервером хоста или AS/400 используется VTAM, найдите в конфигурации "switched major node" (переключаемого главного узла) значение параметра PACING. На рабочей станции DB2 Connect проверьте определение режима IBMRDB при установке связи "LU 6.2 Mode Profile" (профиль режима LU 6.2). Убедитесь, что в этом определении значение параметра "Receive pacing window" (окно синхронизации приема) не больше, чем значение PACING, заданное на VTAM. Можно задать для параметров "Receive pacing window" (окно синхронизации приема) на рабочей станции DB2 Connect и "PACING" на VTAM одинаковые значения, равные 8.

3. Убедитесь, что максимальный размер RU, заданный в определении режима IBMRDB, имеет подходящее значение. При использовании Token Ring рекомендуется задавать не меньше 4 Кбайт. При использовании Ethernet ограничивающим фактором может являться максимальный размер кадра Ethernet, равный 1536 байтам.
4. Обратитесь к администратору вашей среды VTAM и убедитесь, что VTAM использует "adaptive pacing" (адаптивную синхронизацию) для сеансов LU–LU с вашей рабочей станцией DB2 Connect.

Дополнительные советы и замечания по настройке производительности SNA

В этом разделе приводятся дополнительные советы и замечания по настройке производительности SNA для использования с DB2 Connect.

Общая информация о производительности для DB2 Connect

При работе DB2 Connect в основном использует процессор и выполняет мало операций ввода–вывода. Как правило, чем выше скорость процессора, тем быстрее работает DB2 Connect. DB2 Connect полностью использует возможности симметричных многопроцессорных конфигураций.

Быстрый сервер DB2 Connect Enterprise Edition может обработать одну пару запрос/ответ SQL менее чем за пять миллисекунд, если не считать времени, необходимого для клиента, для передачи по сети и для обработки на сервере хоста или AS/400. Простой оператор SQL или запрос с несколькими строками данных может быть обработан от начала до конца менее чем за 0,1 сек. (от клиента к серверу хоста или AS/400 и обратно).

Если в запросе более четырех – пяти операторов SQL, использование хранимых процедур обеспечит высокую производительность OLTP и поможет избежать возможного роста числа конфликтов блокировок из–за задержек операторов SQL в сети.

Обычные причины проблем с производительностью – используемый тип подключения к хосту, характеристики сетевой маршрутизации, а также настройки и структура прикладных программ. Некоторую общую информацию о производительности DB2 Connect можно найти в разделе “Другие источники информации о производительности DB2 Connect” на стр. 156.

Выбор и настройка сетевого подключения

Хорошую производительность при использовании DB2 Connect обеспечивают различные сетевые подключения, в том числе:

1. Плата подключения канала
2. IBM 3172 Model 3, или новые модели, или эквивалентные

3. IBM 2216
4. Open System Adaptor Card (OSA–2, но не OSA–1)
5. IBM 3745 с Network Control Program (NCP)
6. IBM 3174 Terminal Controllers или эквивалентные

Последний вариант не рекомендуется, как сказано ниже.

Наилучший способ соединения с хостом – это использование платы подключения канала ESCON для AIX, Windows NT или Windows 2000. У IBM 3172 Model 3 и IBM 2216 тоже неплохие показатели, но эти платы несколько уступают по пропускной способности ESCON.

При использовании AIX с платами ESCON надо применить исправления PTF, относящиеся к MPC (Multi Path Channel). Без этих исправлений эффективность драйвера AIX SNA ESCON может быть ниже. Более детальную информацию смотрите в разделе “Поддержка Multi Path Channel для SNA с использованием ESCON” на стр. 156. Дополнительную информацию можно также найти на странице <http://www.networking.ibm.com.cms/cmsnew01.html>.

Параметры Communications Server, NCP и VTAM, настраиваемые для повышения эффективности DB2 Connect, описаны в разделе “Как настраивать соединения DB2 Connect через NCP” на стр. 156. Все рекомендации, кроме относящихся к NCP, применимы ко всем типам DB2 Connect и подключений клиент–сервер.

Из-за малой производительности в кадрах в секунду плата OSA–2 в System/390 при большом количестве маленьких транзакций может уступать по пропускной способности плате 3272 Model 3. Подробную информацию о последних усовершенствованиях смотрите в разделе “Сведения об усовершенствованиях OSA–2” на стр. 160.

Плата 3145 с NCP обычно настроена специально под существующий сетевой трафик. Из-за этого при работе с программами баз данных структуры клиент–сервер ее производительность может быть ниже. Основная часть проблем с производительностью DB2 Connect возникает из-за временных задержек между NCP и VTAM и/или между NCP. В разделе “Как настраивать соединения DB2 Connect через NCP” на стр. 156 приводится контрольный список для настройки.

Обычно рекомендуется избегать использования 3174 Terminal Controller, так как размер RU у них – 256 байт – слишком мал. Для поддержки независимых LU в соединениях APPC с базами данных необходим 3174 с микрокодом уровня C. Для некоторых эквивалентных 3174 устройств независимых производителей также могут иметь место подобные ограничения.

Другие источники информации о производительности DB2 Connect

- Посмотрите сайт технической библиотеки DB2 по адресу <http://www.ibm.com/software/data/db2/library>. Чтобы получить свежую информацию о DB2 Connect из WWW, поищите в библиотеке DB2 Universal Database технические замечания ("Technotes") по ключевым словам "DB2CONNECT" и "Performance".
- Много сообщений о производительности DB2 в среде клиент–сервер, в том числе и о DB2 Connect, есть на сайте <http://www.ibm.com/software/data/performance>.

Поддержка Multi Path Channel для SNA с использованием ESCON

Поддержка Multi Path Channel (MPC) для SNA с использованием ESCON позволяет системе, на которой работает IBM eNetwork Communications Server, использовать адаптер ESCON для создания станции связи MPC с хостом. MPC обычно работает быстрее, чем CDLC, потому что:

1. MPC использует отдельные подканалы для чтения и записи.
2. MPC не ограничен размером буфера ввода–вывода. Кадры имеют размер 4 Кбайта и могут быть сблокированы вместе.

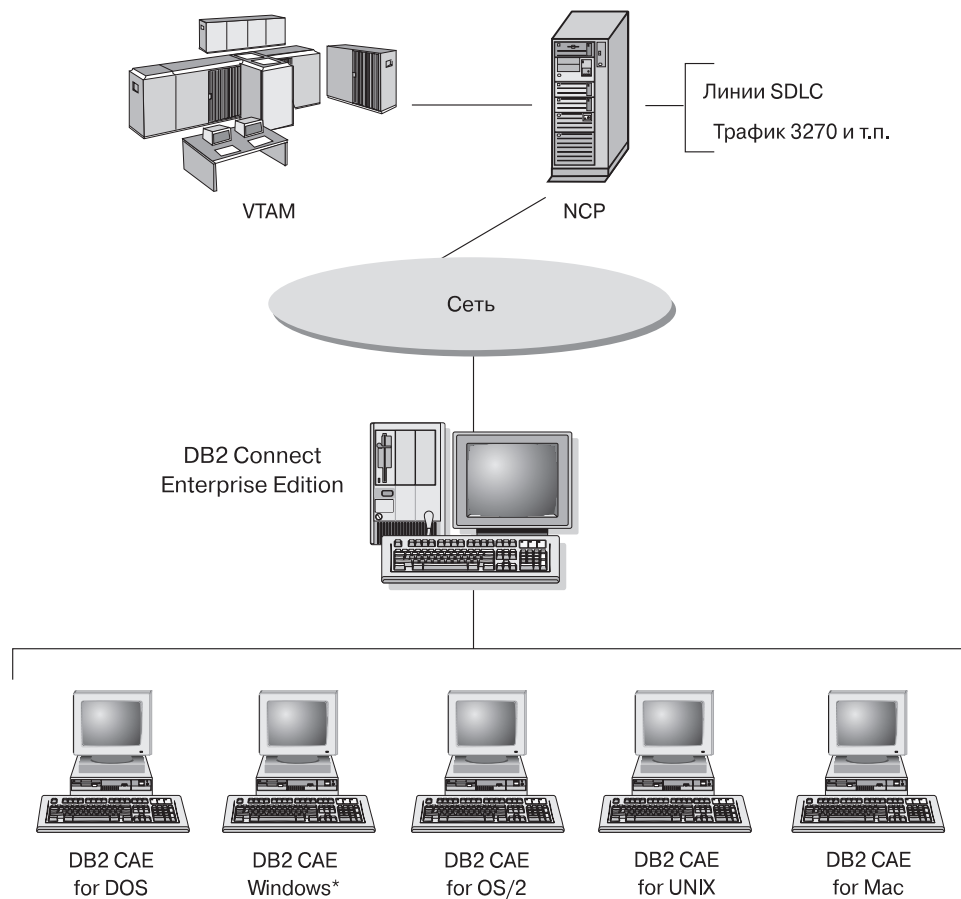
Испытания показали увеличение производительности связи MPC втрое по сравнению с ESCON Channel Data Link Control (CDLC) с размером буфера ввода–вывода менее 1 Кбайта. Для AIX SNA MPC требуется ESCON и MVS VTAM V4R4 или более поздний и уровень кода 4024 для Communications Server for AIX (5765–652). На системах Windows NT надо использовать IBM eNetwork Communications Server for Windows NT Версии 6.

Ниже приводится список исправлений PTF Communications Server for AIX, требуемых для MPC:

номер APAR	номер PTF	имя LPP
IX67032	U449693	sna.books.chdoc
IX67032	U449693	sna.books.esdoc
IX67032	U449300	sna.rte
IX67032	U450027	sna.msg.en_US.rte
IX65820	U447759	sna.dlcchannel
IX67618	U449691	mpc.rte
IX65813	U447758	devices.mca.8fc3.rte

Как настраивать соединения DB2 Connect через NCP

Типичная конфигурация сети может быть такой:



*16- и 32-битные операционные системы Windows.

Рисунок 8. Сетевой сценарий SNA для шлюза DB2 Connect Enterprise Edition

Основное внимание в этом сценарии уделяется пропускной способности и времени ответа между сервером баз данных хоста или AS/400 и шлюзом DB2 Connect Enterprise Edition, а также параметрам, которые могут влиять на них.

Критерии настройки

Изменения предлагается делать в таком порядке:

- 1 - Параметр DELAY в макрокоманде PCCU*
- 2 - Настройка DLC/LLC*
- 3 - Размер PIU*
- 4 - Изменение окна синхронизации*
- 5 - Параметр DELAY в макрокоманде LINE*
- 6 - Изменение MAXBFRU
- 7 - Размер кадров сети

* Возможно значительное увеличение пропускной способности

Размер PIU (RU + 29 байтов): На хосте и на сервере DB2 Connect надо установить максимальный размер RU. Это означает, что размер RU должен быть достаточным для двусторонней работы API (что дает возможность как получать, так и отправлять данные для транзакции, когда это возможно), чтобы стек программы VTAM реже приходилось обращаться. Если сегментация RU нежелательна, максимальный размер RU может ограничивать максимальный размер кадров в сети.

Рекомендуем установить размер блока DB2 Connect (RQRIOBLK), RU и значения окна синхронизации так, чтобы $RU * \text{окно синхронизации} \geq RQRIOBLK$. Например, размер RQRIOBLK по умолчанию – 32 Кбайта – подходит для большинства случаев, и для него можно установить $RU = 4$ Кбайта, а окно синхронизации приема – 8 ($4 * 8 = 32$).

- Размеры RU и окна синхронизации устанавливаются в таблице режима, которая определена и на рабочей станции DB2 Connect, и в VTAM. В обоих местах эта таблица должна определяться одинаково.
- Значение RQRIOBLK устанавливается командой DB2 UPDATE DBM CFG.
- Размер кадров в сети (для I-кадров) устанавливается в конфигурации DLC на рабочей станции DB2 Connect и в NCP.

Размеры окон синхронизации: Следует задать максимальные размеры окон синхронизации сеанса и VR: выбирайте предельное значение, которое не приводит к задержкам в сети или к задержкам VR. Для среды тестирования задайте размер окна 0 или максимальное значение X'3F'.

Значения задержек (DELAY): Задержками управляет параметр DELAY. Значение параметра DELAY в макрокоманде PCCU управляет задержкой исходящих данных (исходящих по отношению к хосту). Значение параметра DELAY в операторе определения LINE для NCP управляет задержкой входящих данных (входящих по отношению к хосту).

Значение DELAY определяет, как долго PIU задерживается в очереди (NCP или VTAM) перед передачей. Назначение этого ожидания – увеличить вероятность поступления других PIU во время задержки, что позволит передать все PIU за один вызов программы канала. Чтобы уменьшить время ответа, надо задать значение DELAY равным 0. Изменение значения выходных задержек на 0 не должно оказать заметного влияния на работу хоста, за исключением

увеличения производительности исходящего трафика. Можно также достичь некоторого увеличения производительности входящего трафика.

Однако для изменения параметра DELAY для NCP на 0 надо проверить некоторые условия. Это значение можно установить равным 0, только если NCP не перегружена и входящий трафик не содержит значительной доли кадров малого размера. Задание значений DELAY, равных 0, может существенно улучшить время ответа, особенно при небольшой нагрузке или в тестовой среде.

```
VTAMB7 PCCU CUADDR=CAF,
            AUTODMP=NO,
            AUTOIPL=NO,
            AUTOSYN=YES,
            BACKUP=YES,
            DELAY=0,
            VFYLM=YES,
            CHANCON=UNCOND,
            MAXDATA=32768,
            DUMPDS=NCPDUMP,
            OWNER=HOSTB7,
            SUBAREA=17
```

```
LNCTLS      GROUP LNCTL=CA,CA=TYPE6,DELAY=0.0,TIMEOUT=500.0
CA0         LINE   ADDRESS=00
PUCHAN0     PU     PUTYPE=5,TGN=1
CA1         LINE   ADDRESS=01
PUCHAN1     PU     PUTYPE=5,TGN=1
```

Обсуждение параметров DELAY приводится в руководстве *VTAM Network Implementation Guide*.

MAXBFRU: Для параметра MAXBFRU надо задать значение в два или в три раза больше, чем максимальный размер PIU.

Настройка слоя DLC/LLC: Размеры окон LLC2 (счетчики окон передачи и приема DLC) для NCP и для шлюза DB2 Connect Enterprise Edition должны быть одинаковы. Это дает значительный эффект, в особенности если сервер — DB2 Connect for AIX. Рекомендуется задавать счетчик окон передачи большим, чем счетчик окон приема.

Вообще говоря, для любого соединения SNA в сети Token Ring таймеры/окна LLC2 надо оптимизировать. В некоторых случаях это может улучшить пропускную способность и время ответа в 6 раз.

Размеры кадров сети: Для сети Token Ring следует задать максимально возможный размер кадра.

Сведения об усовершенствованиях OSA–2

Следующая информация взята из документа IBM WSC Flash номер 9718.

ЗАГОЛОВОК: WSC FLASH 9718: ДОСТУПНЫЕ УСОВЕРШЕНСТВОВАНИЯ OSA–2
ИДЕНТИФИКАТОР ДОКУМЕНТА G023691 ОБЩЕДОСТУПНЫЙ

Усовершенствования Open Systems Adapter 2 (OSA–2) для Systems Network Architecture (SNA) стали доступными ранее, чем это было объявлено. Это следующие усовершенствования:

- o Усовершенствования SNA/APPN для OS/390, MVS/ESA, VM/ESA и VSE/ESA
 - Улучшения доступности: балансировка нагрузки, дублирование и перегрузка
 - Улучшения связи: поддержка большего числа физических устройств (PU) (от 255 PU на порт до 2047 PU на порт).
- o Поддержка для сетей ACF/VTAM for VSE/ESA

ПРИМЕЧАНИЕ: Эти усовершенствования не относятся к OSA–1.

БАЛАНСИРОВКА НАГРУЗКИ, ДУБЛИРОВАНИЕ И ПЕРЕГРУЗКА

БАЛАНСИРОВКА НАГРУЗКИ: Теперь можно задать единый адрес MAC (Medium Access Control) для подключенных физических устройств (PU) OSA–2 SNA/APPN, даже если они подключены через разные физические порты. Эта поддержка обеспечивается только для сред ретрансляции с маршрутизацией от источника (Token Ring и FDDI). При этом отслеживается число сеансов, установленных через порт, и пользовательские сеансы распределяются равномерно по одинаково сконфигурированным портам.

НАДЕЖНОСТЬ: Теперь можно сконфигурировать вторичный путь между рабочей станцией сети и системой хоста. Если первичный путь становится недоступен, сетевой трафик направляется по вторичному пути. Это повышает доступность системы и упрощает управление сетью.

ПЕРЕГРУЗКА: Пользовательские сеансы направляются через первичный порт OSA–2, пока не будет достигнуто предельное число сеансов. Дополнительные пользовательские сеансы направляются на следующий порт OSA–2. Так как все пользовательские рабочие станции конфигурируются одинаково, управление сетью упрощается, а ее масштабируемость повышается. Упрощается добавление новых пользователей.

Поддержку балансировки нагрузки, дублирования и перегрузки обеспечивают следующие исправления PTF для OSA/SF:

- | | | |
|----------------|-----------------------------------|----------|
| o OS/390 и MVS | - OW20205/UW34618 | 03/31/97 |
| o VM/ESA | - OW23952/UW37028 | 03/31/97 |
| o VSE/ESA | - Обеспечивается в VSE/ESA V2.2.1 | 04/29/97 |

ПОДДЕРЖКА БОЛЬШЕГО ЧИСЛА ФИЗИЧЕСКИХ УСТРОЙСТВ (PU) ЧЕРЕЗ OSA/SF:

Изменение архитектуры позволяет теперь поддерживать до 2047 PU на физический порт, определенный для OSA-2 Ethernet, Token-Ring и FDDI, вместо прежнего ограничения 255 PU на порт. Это усовершенствование доступно как для уже установленных, так и для новых систем. Реальные возможности связи могут меняться в зависимости от рабочей нагрузки.

Поддержку большего числа физических устройств (PU) обеспечивают следующие исправления PTF для OSA/SF:

- o OS/390 и MVS - OW23429/UW37210 03/31/97
- o VM/ESA - OW24952/UW37028 03/31/97
- o VSE/ESA - PQ03091/UQ04224 04/29/97

Поддержку большего числа физических устройств (PU) обеспечивают следующие исправления PTF для ACF/VTAM:

- o ACF/VTAM for OS/390 and MVS
 - VTAM 4.1 OW14043/UW24904
 - VTAM 4.2 OW14043/UW24905
 - VTAM 4.3 OW14043/UW24906
- o ACF/VTAM VM/ESA
 - VM60877/UV59834
- o ACF/VTAM VSE/ESA
 - DY44347/UD50254

VSE/ESA - ПОДДЕРЖКА SNA

Поддержка OSA-2 и OSA/SF обеспечивается в VSE/ESA, Версия 2 Выпуск 2.1. Эта поддержка VSE/ESA соответствует объявленной в Заявлении Генеральной Дирекции в Объявлениях об Оборудовании 196-194 и 196-193 от 10 сентября 1996 года.

Возможность OSA-2 поддерживает ACF/VTAM для программ хоста VSE/ESA с прямым доступом к сетям Ethernet, Token-Ring и FDDI и к Forum-совместимым сетям эмуляции Asynchronous Transfer Mode (ATM).

OSA/SF доступна:

- o Как неисключительный элемент OS/390 Выпуска 1 или более нового (5645-001)
- o Как отдельный программный продукт S/390 Open Systems Adapter Support Facility, Версия 1 Выпуск 2 для MVS/ESA 4.3 или более новой (5655-104)
- o Как возможность VM/ESA, Версия 2 Выпуск 2.0 (5654-030)
- o Как компонент VSE Central Functions 6.1.1 в VSE/ESA Версия 2 Выпуск 2.1 (5690-VSE).

ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ

Объявления 297-043, 297-040

Другие источники информации

В этом разделе перечислены дополнительные источники информации.

Другие публикации

Дополнительную информацию о производительности смотрите в публикациях:

- *DB2 Connect for OS/2 to DB2 Performance Benchmark*
- *SNA Server for AIX and SNA Server Gateway for AIX Performance Guide*

Источники в WWW

Подробную информацию о настройке производительности DB2 Connect, а также исследования и примеры, можно найти в WWW. Откройте в браузере Web URL:

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

Дополнительные советы и подсказки пользователям SNA

Смотрите раздел “Дополнительные советы и замечания по настройке производительности SNA” на стр. 154.

Глава 13. Диагностика ошибок

Среда DB2 Connect включает много программных и аппаратных средств и средств связи. При диагностике ошибок разумно исключать гипотезы, уточняя доступные данные, пока не будет получено заключение (то есть найден источник ошибки).

Следующие темы помогут вам при диагностике ошибок:

- “Другие источники информации”
- “Сбор информации” на стр. 164
- “Ошибка при начальном соединении” на стр. 164
- “Ошибки, возникающие после начального соединения” на стр. 165
- “Средства диагностики” на стр. 167
- “Утилита трассировки (ddcstrc)” на стр. 167
- “Часто встречающиеся ошибки при работе с DB2 Connect” на стр. 176

Собрав информацию и выбрав тему, переходите к соответствующему разделу.

Другие источники информации

В этом разделе перечислены дополнительные источники информации.

Руководство Troubleshooting Guide

Подробная информация о диагностике ошибок DB2 Connect и DB2 Universal Database приводится в книге *Troubleshooting Guide*.

Источники в WWW

Последнюю информацию и советы по диагностике ошибок DB2 Connect можно найти в технической библиотеке по службам и продуктам DB2 в WWW:

1. Вызовите страницу Web: <http://www.ibm.com/software/data/db2/library/>.
2. Выберите ссылку DB2 Universal Database.
3. Ищите “Technotes” с ключевыми словами “DDCS” или “Connect”.

Документация по смысловым кодам APPC, CPI-C и SNA

Прежние издания по смысловым кодам APPC, CPI-C и SNA поставляются теперь в новом формате – в виде файла PostScript и книги в формате HTML (только на английском языке).

PDF–версию этой книги можно найти на компакт–диске публикаций по DB2. В формате HTML эта книга всегда устанавливается вместе с DB2 Connect; она интегрирована с информационным центром DB2.

Сбор информации

Диагностика ошибок включает их локализацию и исследование возможных причин. Прежде всего следует собрать информацию и определить, что известно, каких данных не хватает и какие гипотезы можно исключить. В качестве первого шага ответьте на следующие вопросы:

- Было ли успешным начальное соединение?
- Исправно ли оборудование?
- Не нарушена ли связь?
- Не вносили ли в сеть изменения, после которых стали неверны старые записи в каталогах?
- Была ли запущена база данных?
- Где нарушена связь – между клиентом и рабочей станцией DB2 Connect, рабочей станцией DB2 Connect и сервером баз данных хоста или AS/400, для всех клиентов или для одного?
- Что можно определить по содержанию сообщения и по возвращенным в нем элементам?
- Имеет ли смысл применить диагностические программы?
- Успешно ли выполняется та же задача на других компьютерах?
- Если это удаленная задача, можно ли выполнить ее локально?

Ошибка при начальном соединении

Ответьте на следующие вопросы и убедитесь, что установка была проведена правильно.

1. *Успешно ли завершился процесс установки?*
 - Все ли требуемые программы были в наличии?
 - Достаточно ли было памяти и места на диске?
 - Была ли установлена поддержка удаленных клиентов?
 - Завершилась ли установка программы связи без сообщений об ошибках?
2. *Для систем на базе UNIX: был ли создан экземпляр продукта?*
 - В качестве пользователя root создали ли вы пользователя – владельца экземпляра и группу sysadm?
3. *Если требовалась информация о лицензии, была ли она обработана успешно?*

- Для систем на базе UNIX: внесли ли вы в файл блокировки узла пароль, предоставленный IBM?
4. *Правильно ли были сконфигурированы сервер баз данных хоста или AS/400 и программы связи на рабочей станции?*
- Обратите внимание на три конфигурации:
 - a. Конфигурация сервера баз данных хоста или AS/400 идентифицирует реквестер прикладных программ для сервера. На СУБД сервера хоста или AS/400 в системном каталоге должны быть записи, задающие для реквестера положение, сетевой протокол и защиту.
 - b. Конфигурация рабочей станции DB2 Connect определяет клиенты для сервера и сервер хоста или AS/400 для клиента.
 - c. В конфигурации клиентской рабочей станции должно быть указано имя рабочей станции и протокол связи.
 - Исследуя ошибку начального соединения, для соединений SNA надо убедиться, что все имена LU (логических устройств) и PU (физических устройств) заданы полностью и правильно, а для соединений TCP/IP — что указаны правильный номер порта и имя хоста.
 - И у администратора базы данных сервера хоста или AS/400, и у администратора сети есть утилиты диагностики.
5. *Достаточно ли у вас прав для использования базы данных сервера хоста или AS/400?*
- Обратите внимание на права доступа для пользователя, правила табличных спецификаторов и права для ожидаемых результатов.
6. *Что будет, если посылать операторы SQL на сервер баз данных хоста или AS/400 с помощью процессора командной строки?*
- Выполнена ли процедура связывания процессора командной строки с сервером баз данных хоста или AS/400?

Ошибки, возникающие после начального соединения

Локализацию ошибки удобно начать с поиска ответов на следующие вопросы:

1. *Проходит ли работа в каких-нибудь особых или необычных условиях?*
 - Используется ли новая программа?
 - Используются ли новые процедуры?
 - Не влияют ли на систему какие-нибудь недавние изменения? Например, не изменялись ли программы со времени последней успешной работы данной программы или сценария?
 - Для прикладных программ: какие интерфейсы прикладного программирования (API) используются в программе?

- Запускались ли ранее на системе данного пользователя прикладные программы, создававшиеся при помощи того же программного обеспечения и использующие те же API?
 - Не было ли недавно установлено PTF? Если ошибка возникла при попытке пользователя воспользоваться функцией, которую не использовали (или не загружали) на данной операционной системе со времени ее установки, определите последний уровень PTF IBM и загрузите этот уровень *после* установки функции.
2. *Возникала ли данная ошибка раньше?*
 - Сохранились ли описания, как эту ошибку устраняли раньше?
 - Кто участвовал в этом, и можно ли с ними посоветоваться относительно возможных мер?
 3. *Использовали ли вы команды программ связи, которые возвращают информацию о сети?*
 - Есть ли в вашем программном обеспечении SNA программа проверки?
 - Если вы работаете с TCP/IP, полезную информацию могут дать команды и демоны TCP/IP.
 4. *Нет ли полезной информации в SQLCA (области связи SQL)?*
 - При диагностике ошибок следует изучить содержимое полей SQLCODE и SQLSTATE.
 - SQLSTATE позволяют создателям программ проверять классы ошибок, общих для семейства продуктов баз данных DB2. В сети распределенных реляционных баз данных это поле может дать общие сведения. Дополнительную информацию смотрите в руководстве *Справочник по сообщениям*.
 5. *Была ли на сервере выполнена команда DB2START?* Кроме того, проверьте правильность значения переменной среды DB2COMM для клиентов, пытающихся получить удаленный доступ к серверу.
 6. *Смогли ли установить соединение с сервером другие компьютеры, выполняющие ту же задачу?* Возможно, было достигнуто максимальное число клиентов, пытающихся установить соединение с этим сервером. Получает ли клиент возможность установить соединение с сервером после того, как другой клиент прервал свое соединение?
 7. *Правильно ли задан адрес компьютера?* Убедитесь, что он уникален в сети.
 8. *При удаленном соединении: получил ли клиент необходимые права?* Даже при успешном соединении с экземпляром могут быть не даны права на уровне баз данных или таблиц.
 9. *Связывались ли с удаленной базой данных другие компьютеры?* В распределенных средах маршрутизаторы или мосты между сетями могут блокировать связь между клиентом и сервером. Например, при работе с APPC убедитесь, что можно установить сеанс. При работе с TCP/IP попробуйте сделать PING для удаленного хоста.

Средства диагностики

При ошибках можно использовать следующие средства:

- Служебный журнал первого отказа, где диагностическая информация собирается и хранится в доступном для чтения формате. Дополнительную информацию смотрите в книге *Troubleshooting Guide*. Сообщения в этом журнале описаны в книге *Справочник по сообщениям*.
- `db2diag.log`

Для систем UNIX путь к этому файлу –
`/u/db2/sqllib/db2dump/db2diag.log`, где `db2` – имя экземпляра.

Для систем Intel путь к этому файлу – `x:\sqllib\db2\db2diag.log`, где `x:` – логический диск, а `db2` – имя экземпляра.

- `db2alert.log` (Находится там же, где `db2diag.log`.)
- Утилита трассировки, описанная в разделе “Утилита трассировки (`ddcstrc`).”
- Для систем на базе UNIX – команда `ps`, которая выводит в стандартный выходной поток информацию о состоянии процесса по всем активным процессам.
- В системах на базе UNIX при серьезных ошибках в текущем каталоге создается файл ядра. Он содержит образ памяти прерванного процесса и позволяет определить, какая функция вызвала ошибку.
- В системах Windows NT и Windows 2000 используйте Окно просмотра событий.

Подробную информацию о диагностике ошибок для соединений TCP/IP (и о других темах) смотрите в руководстве *Troubleshooting Guide* или проведите поиск “Technotes” в технической библиотеке по службам и продуктам DB2 (смотрите раздел “Источники в WWW” на стр. 163).

Утилита трассировки (`ddcstrc`)

Утилита `ddcstrc` предназначена для записи данных, передаваемых между рабочей станцией DB2 Connect (которая обслуживает клиента баз данных) и системой управления сервером баз данных хоста или AS/400.

Администратору баз данных (или разработчику программ) полезно понимать, как работает этот поток данных, потому что это может помочь в определении причины конкретной проблемы. Например, вы вводите команду базы данных `CONNECT TO` для сервера баз данных хоста или AS/400, но при ее выполнении происходит ошибка и вы получаете код возврата. Если вы знаете, какая информация передается системе управления сервером баз данных хоста или AS/400, можно определить причину сбоя, даже если код возврата содержит

только самую общую информацию. Многие проблемы вызываются просто пользовательскими ошибками.

В выходной информации ddcstrc перечисляются потоки данных, которыми обменивались рабочая станция DB2 Connect и система управления сервером баз данных хоста или AS/400. Данные, посланные на сервер баз данных хоста или AS/400, помечаются SEND BUFFER, а данные, полученные от сервера баз данных хоста или AS/400, помечаются RECEIVE BUFFER.

Если буфер приема содержит информацию SQLCA, далее следует форматированная интерпретация этих данных с меткой SQLCA. Поле SQLCODE в SQLCA содержит *неотображенное* значение, возвращаемое сервером баз данных хоста или AS/400. (Дополнительная информация об отображении приведена в разделе Глава 11, “Отображение SQLCODE” на стр. 123.) Буферы отправки и приема в файле идут в порядке от самых ранних к самым поздним. Для каждого буфера указываются:

- ID процесса
- Метка SEND BUFFER, RECEIVE BUFFER или SQLCA. Первая команда или объект DDM в буфере имеет метку DSS TYPE.

Остальные данные в буферах отправки и приема организуются в пять столбцов:

- Счетчик числа байтов
- В столбцах 2 и 3 в формате ASCII или EBCDIC представлен поток данных DRDA, которыми обменялись две системы
- Представление столбцов 2 и 3 в формате ASCII
- Представление столбцов 2 и 3 в формате EBCDIC

Дополнительную информацию о DDM смотрите в публикациях:

- *DB2 for OS/390 Reference for Remote DRDA Requesters and Servers*
- *Distributed Relational Database Reference*
- *Distributed Data Management Architecture Level 3: Reference*

Синтаксис команды трассировки

Команда вводится в командной строке операционной системы и имеет следующий синтаксис:

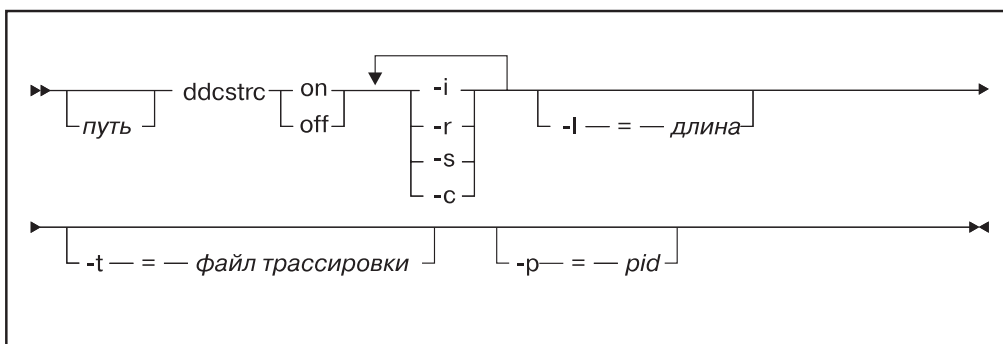


Рисунок 9. Синтаксис команды *ddcstrc*

Примечание: Синтаксис этой команды в разных операционных системах может немного отличаться. Например, в операционной системе OS/2 вместо "-" можно использовать "/".

Параметры трассировки

- on** Включает трассировку DB2 Connect потоков DRDA при обмене данными с сервером баз данных хоста или AS/400.
- off** Выключает трассировку DB2 Connect потоков DRDA при обмене данными с сервером баз данных хоста или AS/400.
- i** В трассировочную информацию будут включены временные отметки.
- r** Трассирует потоки данных DRDA, получаемые от сервера баз данных хоста или AS/400.
- s** Трассирует потоки данных DRDA, посылаемые на сервер баз данных хоста или AS/400.
- c** Трассирует SQLCA, получаемые от сервера баз данных хоста или AS/400.

Параметры по умолчанию: **-t**, **-s** и **-c**.

-l=длина

Задаёт размер буфера для хранения трассировочной информации. По умолчанию принимается 1 Мбайт, минимальное значение – 64 Кбайта.

-t=файл_трассировки

Задаёт назначение трассировки; *файл_трассировки* может быть именем файла или стандартного устройства. Если в имени файла путь указан не полностью, вместо отсутствующей части используется текущий путь. Имя файла по умолчанию – *ddcstrc.dmp*.

-p=id_процесса

Трассируются события только данного процесса. Если **-p** не указан, все процессы экземпляра программы пользователя записываются в выходной файл.

Примечание: Для удаленного клиента *id_процесса* можно найти в поле ID агента, который возвращает системный монитор баз данных.

Дополнительную информацию смотрите в разделе Глава 8, “Монитор системы баз данных” на стр. 97.

Вывод трассировки

Утилита `ddcstrc` записывает в *файл_трассировки* следующую информацию:

- `-t`
 - Тип объекта/ответа DRDA
 - Буфер приема
- `-s`
 - Тип требования DRDA
 - Буфер отправки
- `-c`
 - Подпрограммы
- Информация об ошибках CPl-C
 - Код возврата функции приема
 - Уровень серьезности ошибки
 - Используемый протокол
 - Используемый API
 - Функция
 - Код возврата CPl-C
 - Номер ошибки
 - Внутренний код возврата
- Информация об ошибках SNA
 - Код возврата функции приема
 - Уровень серьезности ошибки
 - Используемый протокол
 - Функция
 - Имя LU партнера
 - Номер ошибки
- Информация об ошибках TCP/IP
 - Код возврата функции приема
 - Уровень серьезности ошибки
 - Используемый протокол
 - Используемый API

- Функция
- Номер ошибки

Примечания:

1. Нулевое значение кода выхода означает, что команда завершена успешно, ненулевое значение свидетельствует о неудаче.
2. Возвращаемые поля могут быть разными в зависимости от используемого API. SNA API используется только для соединений 2PC SPM.
3. Возвращаемые поля могут быть разными в зависимости от платформы, на которой выполняется DB2 Connect, даже для одного и того же API.
4. Если ddcstrc направляет вывод в уже существующий файл, старое содержимое файла будет стерто, если разрешения для файла позволяют это сделать.

Анализ выходного файла трассировки

Ниже приведен пример вывода, где показаны некоторые потоки данных DRDA между рабочими станциями DB2 Connect и сервером баз данных хоста или AS/400. При этом пользователь ввел в командной строке команду базы данных CONNECT TO.

На рис. 10 на стр. 172 используется DB2 Connect Enterprise Edition, Версия 7 и DB2 Universal Database for OS/390, Версия 5.1 с соединением APPC.

На рис. 11 на стр. 173 используется DB2 Connect Enterprise Edition, Версия 7 и DB2 Universal Database for OS/390, Версия 5.1 с соединением TCP/IP.

```

1      DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
      pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 177

      SEND BUFFER:  EXCSAT RQSDSS          (ASCII)          (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 006AD04100010064 10410020115E8482 .j.A...d.A. .^.. .|}.....;db
0010 F282974040404040 4040404040404040 ...@@@@@@@@@@@@ 2bp
0020 4040F0F0F0F1F7F3 C5C3000C116DA685 @@.....m.. 000173EC..._we
0030 81A2859340400013 115AC4C2F240C396 ....@...Z...@.. ase1 ...]DB2 Co
0040 95958583A340F54B F200141404140300 .....@.K..... nnect 5.2.....
0050 0414440003240700 05240F0003000D11 ..D..$...$.....
0060 47D8C4C2F261F6F0 F0F00085D0010002 G....a..... .QDB2/6000.e)...
0070 007F200100162110 E2C1D56DC6D9C1D5 .. ...!....m.... .".....SAN_FRAN
0080 C3C9E2C3D6404040 40400006210F2407 .....@@@@...!.$: CISCO .....
0090 000D002FD8E3C4E2 D8D3C1E2C3000C11 .../..... ....QTDSQLASC...
00A0 2EE2D8D3F0F5F0F2 F0003C210437E2D8 .....

```

```

3      DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
      pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 177

      SEND BUFFER:  RDBCMM RQSDSS          (ASCII)          (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 000AD00100010004 200E          ..... . ..}.....

```

```

4      DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
      pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 178

      RECEIVE BUFFER:  ENDUOWRM RPYDSS      (ASCII)          (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 002BD05200010025 220C000611490004 .+.R...%"....I.. ..}.....
0010 00162110E2C1D56D C6D9C1D5C3C9E2C3 ..!....m..... ...SAN_FRANCISC
0020 D640404040400005 211501000BD00300 .@@@@...!..... 0 .....}..
0030 0100052408FF          ...$. . . . .

```

```

5      DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
      pid 95212; tid 537115484; node 0; cpid 0; sec 0; nsec 0; tpoint 179

```

Подпрограммы

```

SQLCAID: SQLCA
SQLCABC: 136
SQLCODE: 0
SQLERRML: 0
SQLERRMC:
SQLERRP: DSN
SQLERRD[0->5]: 00000000, 00000000, 00000000, 00000000, 00000000, 00000000
SQLWARN(0->A): , , , , , , , , , ,
SQLSTATE: 00000

```

Рисунок 10. Пример вывода трассировки (соединение типа APPC)

```

1      DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
      pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 177
      SEND BUFFER: EXCSAT RQSDSS      (ASCII)      (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 006ED04100010068 10410020115E8482 .n.A...h.A. .^.. .>}.....;db
0010 F282974040404040 4040404040404040 ...@@@@@@@@@@@@ 2bp
0020 4040F0F0F0F1F3F9 F9C5000C116DA685 @@.....m.. 0001399E..._we
0030 81A2859340400013 115AC4C2F240C396 ....@...Z...@.. ase1 ...]DB2 Co
0040 95958583A340F54B F200181404140300 .....@.K..... nnect 5.2.....
0050 0514740005240700 05240F0003144000 ..t..$...$....@. ....
0060 05000D1147D8C4C2 F261F6F0F0F00010 ...G...a..... ....QDB2/6000..
0070 D0410002000A106D 000611A20003003C .A.....m.....< }....._...s....
0080 D04100030036106E 000611A200030016 .A...6.n..... }.....>...s....
0090 2110E2C1D56DC6D9 C1D5C3C9E2C3D640 !...m.....@ ..SAN_FRANCISCO
00A0 40404040000C11A1 9781A2A2A6969984 @@@..... ..password
00B0 000A11A0A4A28599 8984009CD0010004 ..... ..userid..}...
00C0 0096200100162110 E2C1D56DC6D9C1D5 .. ...!...m.... .o.....SAN_FRAN
00D0 C3C9E2C3D6404040 40400006210F2407 .....@@@@...!$. CISCO .....
00E0 000D002FD8E3C4E2 D8D3C1E2C3000C11 .../..... ....QTDSQLASC...
00F0 2EE2D8D3F0F5F0F2 F0003C210437E2D8 .....

3      DB2 fnc_data      gateway_drda_ar      sqljcsend (1.35.10.80)
      pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 177

      SEND BUFFER: RDBCMR RQSDSS      (ASCII)      (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 000AD00100010004 200E ..... . ..}.....

4      DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
      pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 178

      RECEIVE BUFFER: ENDUOWRM RPYDSS (ASCII)      (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 002BD05200010025 220C000611490004 .+.R...%"....I.. ..}.....
0010 00162110E2C1D56D C6D9C1D5C3C9E2C3 ..!...m..... ....SAN_FRANCISCO
0020 D640404040400005 211501000BD00300 .@@@@...!..... 0 .....}...
0030 0100052408FF .....$.. .....

5      DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
      pid 80286; tid 537125164; node 0; cpid 0; sec 0; nsec 0; tpoint 179
Подпрограммы
SQLCAID: SQLCA
SQLCABC: 136
SQLCODE: 0
SQLERRML: 0
SQLERRMC:
SQLERRP: DSN
SQLERRD[0->5]: 00000000, 00000000, 00000000, 00000000, 00000000, 00000000
SQLWARN(0->A): , , , , , , , , ,
SQLSTATE: 00000

```

Рисунок 11. Пример вывода трассировки (соединение типа TCP/IP)

В данные трассировки попала следующая информация:

- ID процесса (PID) программы клиента
- RDB_NAME, занесенное в каталог DCS
- CCSID DB2 Connect
- CCSID сервера баз данных хоста или AS/400
- Система управления сервером баз данных хоста или AS/400, с которой связывается система DB2 Connect

Первый буфер содержит команды Exchange Server Attributes (EXCSAT) и Access RDB (ACCRDB), отправленные системе управления сервером баз данных хоста или AS/400. Эти данные отправляются, когда вводится команда базы данных CONNECT T0.

Следующий буфер содержит ответ, который DB2 Connect получает от системы управления сервером баз данных хоста или AS/400. Он содержит ответные данные Exchange Server Attributes (EXCSATRD) и ответное сообщение Access RDB (ACCRDBRM).

Анализ EXCSAT и ACCRDB

Команда EXCSAT содержит имя рабочей станции, клиент которой задан объектом Server Name (SRVNAM) с кодом X'116D' согласно спецификации DDM. команда EXCSAT находится в первом буфере. В команде EXCSAT величина X'116DA68581A28593' (в кодировке CCSID 500) в символьном виде, если отбросить X'116D', дает *weasel*.

Команда EXCSAT также содержит объект EXTNAM (External Name), который часто записывается в диагностическую информацию системы управления сервером баз данных хоста или AS/400. Он состоит из 20-байтного ID программы, за которым следует 8-байтный ID процесса (или 4-байтный ID процесса и 4-байтный ID потока). Он обозначен кодом X'115E' и в нашем примере имеет значение db2bp_32, дополненное пробелами, за которым следует 0000BE5C. Для UNIX-клиента баз данных это значение соответствует значению в команде **ps**, которая записывает в стандартный вывод информацию о состоянии активных процессов.

Команда ACCRDB содержит RDB_NAME в объекте RDBNAM, код которого – X'2110'. команда ACCRDB следует за командой EXCSAT в первом буфере. Величина X'2110E2C1D56DC6D9C1D5C3C9E2C3D6' в команде ACCRDB в символьном виде, если отбросить X'2110' – это SAN_FRANCISCO. Это соответствует полю имени базы данных назначения в каталоге DCS.

Код учетной строки – X'2104' (смотрите раздел “Реализация учета оплаты в DB2 Universal Database for OS/390” на стр. 56).

Кодовый набор для рабочей станции DB2 Connect задается с помощью объекта CCSID в CCSIDSBC (CCSID для однобайтных символов) с кодом

X'119C' в команде ACCRDB. В нашем примере CCSIDSBC – это X'0352', то есть 850.

Если присутствуют дополнительные объекты CCSIDDBC (CCSID для двухбайтных символов) и CCSIDMBC (CCSID для символов смешанной длины), с кодами соответственно X'119D' и X'119E', это значит, что рабочая станция DB2 Connect сконфигурирована для поддержки кодовой страницы DBCS. В выходном файле примера нет этих двух дополнительных кодов, то есть рабочая станция не сконфигурирована для использования DBCS.

Примечание: Потоки TCP/IP содержат две новые команды: ACCSEC используется для получения доступа к менеджеру защиты и механизмам поддержки защиты обмена, а команда SECCHK, содержащая элементы аутентификации, используется для аутентификации конечного пользователя соединения. Информация ACCSEC и SECCHK появляется в трассировке только для соединений TCP/IP и расположена между EXCSAT и ACCRDB.

Анализ EXCSATRD и ACCRDBRM

Величины CCSID возвращаются также с сервера баз данных хоста или AS/400 в ответном сообщении Access RDB (ACCRDBRM) во втором буфере. Этот буфер содержит EXCSATRD и следующий за ним ACCRDBRM. Выходной файл примера содержит CCSID 500 (X'01F4', CCSID набора однобайтных символов) для системы сервера баз данных хоста или AS/400

Если DB2 Connect не распознает кодовую страницу, пришедшую с сервера баз данных хоста или AS/400, пользователю вместе с кодовыми страницами источника и назначения будет возвращен SQLCODE –332. Если сервер баз данных хоста или AS/400 не распознает кодовый набор, посланный с DB2 Connect, он возвратит VALNSPRM (значение параметра не поддерживается) с кодом DDM X'1252', который для пользователя будет преобразован в SQLCODE –30073.

ACCRDBRM также содержит параметр PRDID (идентификатор продукта, с кодом X'112E'). Значение равно X'C4E2D5F0F5F0F1F0'. Эта шестнадцатеричная строка в EBCDIC соответствует DSN05010. Согласно стандартам, DSN – это DB2 for MVS/ESA или DB2 Universal Database for OS/390. Указывается также номер версии (5.1). ARI – это DB2 for VSE & VM, SQL – DB2 Common Server, а QSQ – DB2 Universal Database for AS/400.

Анализ последующих буферов

Для получения дополнительной информации можно проанализировать последующие буферы. Третий буфер содержит информацию о принятии. Команда **commit** указывает системе управления сервером баз данных хоста

или AS/400 выполнить принятие для текущей единицы работы. Четвертый буфер принимается от системы управления сервером баз данных хоста или AS/400 в результате выполнения принятия или отката. Он содержит ответное сообщение Конец единицы работы (ENDUOWRM), указывающее на завершение текущей единицы работы. В нашем примере он содержит пустое значение SQLCA с кодом DDM X'2408' и следующим за ним X'FF'. Пустое значение SQLCA (X'2408FF') означает успешное завершение операции (SQLCODE 0). Если буфер приема содержит SQLCA (непустое значение SQLCA), то ddcstrc после этого буфера даст форматированную интерпретацию значения SQLCA.

На рис. 12 показан пример буфера приема со значением SQLCA ошибки, а также форматированной интерпретацией SQLCA. Это значение SQLCA — результат попытки удаления строк в несуществующей таблице.

```

1      DB2 fnc_data      gateway_drda_ar      sqljcrecv (1.35.10.81)
      pid 48732; tid 1; node 0; cpid 0; sec 0; nsec 0; tpoint 178

      RECEIVE BUFFER:  SQLCARD  OBJDSS      (ASCII)      (EBCDIC)
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF 0123456789ABCDEF
0000 0065D0030001005F 240800FFFFFFFF34F4 .e....._$......4. ..}.....^.....4
0010 F2F7F0F4C4E2D5E7 D6E3D34000E2C1D5 .....@..... 2704DSNXOTL .SAN
0020 6DC6D9C1D5C3C9E2 C3D64040404040FF m.....@@@. _FRANCISCO .
0030 FFFE0C0000000000 000000FFFFFFFF00 .....
0040 0000000000000040 4040404040404040 .....@@@@@. ....
0050 4040000000FC4C4 C3E2E4E2F14BD4E8 @@.....K.. ....DDCSUS1.MY
0060 E3C1C2D3C5 ..... TABLE

2      DB2 fnc_data      gateway_drda_ar      sqljmsca (1.35.10.108)
      pid 48732; tid 1; node 0; cpid 0; sec 0; nsec 0; tpoint 179
Подпрограммы

      SQLCAID:  SQLCA
      SQLCABC:  136
      SQLCODE:  -204
      SQLERRML: 15
      SQLERRMC: DDCSUS1.MYTABLE
      SQLERRP:  DSNXOTL
      SQLERRD[0->5]: FFFFE0C, 00000000, 00000000, FFFFFFFF, 00000000, 00000000
      SQLWARN(0->A): , , , , , , , , ,
      SQLSTATE: 42704

```

Рисунок 12. Пример буфера приема

Часто встречающиеся ошибки при работе с DB2 Connect

В этом разделе перечислены характерные признаки ошибок соединения при использовании DB2 Connect. Для каждой ошибки приводится:

- Номер сообщения и код возврата (или код возврата для данного протокола), связанный с этим сообщением. Для каждой комбинации сообщения и кода возврата есть отдельный заголовок; они расположены в порядке нумерации сообщений и сгруппированы по кодам возврата.
- Признак дается обычно в форме перечня сообщений примера.
- Предлагаемое решение приводится с указанием вероятной причины ошибки. В некоторых случаях предложенное решение не является единственным.

Примечания:

1. Свежую информацию по рекомендуемым уровням исправления программ смотрите в руководстве Quick Beginnings для вашего программного продукта и в последнем выпуске Release Notes.
2. Для комбинаций сообщения и кода возврата, свойственных разным типам связи APPC, может также указываться смысловой код SNA. На сегодня любую информацию о смысловом коде SNA, связанную с каким-то определенным сообщением, надо получать от подсистемы SNA.

Иногда смысловые коды SNA можно найти в системных журналах. Эта возможность зависит от используемой подсистемы SNA, и в ряде случаев, чтобы получить информацию о смысловом коде, приходится повторно генерировать ошибку при включенной трассировке SNA.
3. Термин шлюз относится к DB2 Connect Enterprise Edition.

SQL0965 или SQL0969

Признак

Сообщения SQL0965 и SQL0969 могут посылаться с множеством различных кодов возврата от DB2 Universal Database for AS/400, DB2 Universal Database for OS/390, DB2 for MVS/ESA и DB2 for VM & VSE.

Если вы получили одно из этих сообщений, надо найти исходный код SQL в документации по программному продукту сервера баз данных, который послал сообщение.

Решение

Код SQL, полученный от базы данных хоста, не поддается преобразованию. Исправьте ошибку, исходя из ее кода, и повторите невыполненную команду.

SQL1338 при выполнении оператора CONNECT

Признак / Причина

Символическое имя назначения не определено или определено неправильно.

Это может произойти, например, при использовании узла APPC, если символическое имя назначения, указанное в каталоге узла DB2, не соответствует записи CPI-C в конфигурации локальной подсистемы связи APPC.

Другая возможная причина – на вашем компьютере установлено несколько стеков SNA. Возможно, надо будет исправить PATH и LIBPATH, чтобы требуемый стек стоял первым.

Решения

1. Убедитесь, что имя профиля информации о стороне взаимодействия CPIC, указанное в записи каталога узла DB2, соответствует конфигурации SNA (регистр символов учитывается).
2. Возможно, надо исправить PATH и LIBPATH, чтобы требуемый стек SNA стоял первым.

SQL1403N при выполнении оператора CONNECT

Признак

SQL1403N Неправильные имя пользователя и/или пароль.

Решение

1. Неудачная аутентификация пользователя на рабочей станции DB2 Connect. Посмотрите, предполагалась ли аутентификация пользователя на рабочей станции DB2 Connect.

Если это так, проверьте, что в операторе CONNECT пароль указан верно.

В противном случае в записи в системном каталоге баз данных, очевидно, было ошибочно указано AUTHENTICATION SERVER (это значение используется по умолчанию, если параметр AUTHENTICATION не задан явно). В этом случае следует повторно занести запись в каталог с указанием AUTHENTICATION DCS или CLIENT.

2. Отсутствует пароль для отсылки базе данных сервера назначения. Если в записи в системном каталоге баз данных указано AUTHENTICATION DCS, клиент DB2 должен посылать пароль базе данных сервера назначения. На некоторых платформах, например, на AIX, пароль можно использовать, только если он указан в операторе CONNECT.

SQL5043N

Признак

Не были успешно запущены системы поддержки одного или нескольких протоколов связи. Однако ядро менеджера баз данных запущено успешно.

Возможно, на шлюзе DB2 Connect не запущен протокол TCP/IP. До этого соединение с клиентом могло работать нормально.

Если `diaglevel = 4`, журнал `db2diag.log` может содержать подобную запись, например:

```
1997-05-30-14.09.55.321092 Instance:svtdbm5 Node:000
PID:10296(db2tcpdm) Appid:none
common_communication sqlcctcpconnmgr_child Probe:46
DIA3205E Адрес гнезда "30090", сконфигурированный в файле служб
TCP/IP и необходимый для поддержки сервера TCP/IP,
используется другим процессом.
```

Решение

Это предупреждение сигнализирует о том, что DB2 Connect, действующая как шлюз для удаленных клиентов, испытывает затруднения при обработке одного или нескольких клиентских протоколов связи. Это могут быть протоколы TCP/IP, APPC и другие, причем обычно данное сообщение указывает, что один из протоколов связи, определенных для DB2 Connect, сконфигурирован неверно.

Часто это происходит из-за того, что переменная профиля DB2COMM не определена или определена неправильно. В общем случае ошибка возникает из-за несоответствия между переменной DB2COMM и именами, определенными в конфигурации менеджера баз данных (например, `svcsname`, `pnname`, или `trname`).

Сообщение об ошибке SQL5043 может быть получено при работе первоначального успешного соединения, в то время как ни одна из конфигураций не изменилась. Это может произойти при использовании протокола TCP/IP, когда удаленная система по какой-то причине ненормально завершает соединение. В этом случае соединение может продолжать существовать на клиенте, и есть вероятность его восстановления без дальнейшего вмешательства в ход сеанса с помощью приведенных ниже команд.

Скорее всего, у одного из клиентов, соединенных со шлюзом, все еще есть хэндл на порту TCP/IP. На каждом клиентском компьютере, который соединен со шлюзом, введите команды:

1. `db2 terminate`
2. `db2stop`

SQL30020

Признак

SQL30020N Выполнение завершилось неудачно из-за ошибки распределенного протокола, которая повлияет на выполнение последующих команд и операторов SQL.

Решения

При возникновении этой ошибки необходимо обратиться к службе поддержки.

Посмотрите в каталоге db2dump файл дампа ffdc (pid.000). Затем отформатируйте этот файл дампа командой db2fdump и проверьте, есть ли в полученном файле слово "ERROR". Возможно, в нем будет список MVSD. В этом случае посмотрите дальнейшую информацию на консоли MVS и отыщите код аварийного завершения в руководстве DB2 for MVS Messages and Codes.

SQL30060

Признак

SQL30060N "<ID_авторизации>" не имеет полномочий для выполнения операции "<операция>".

Решение

При соединении с DB2 for MVS или DB2 for OS/390 таблицы базы данных связей (Communications Database, CDB) не были правильно изменены.

Посмотрите книги:

- DB2 Connect. Быстрый старт, или
- Дополнение по возможностям соединений DB2

SQL30061

Признак

Произошло соединение с неверным хостом или сервером баз данных AS/400 – не удастся найти базу данных назначения.

Решение

Возможно, в записи каталога DCS указано неверное имя сервера баз данных. В этом случае прикладной программе возвращается код SQLCODE –30061.

Проверьте записи узла DB2 базы данных и каталога DCS. Поле имени базы данных назначения в записи каталога DCS должно соответствовать имени

базы данных на этой платформе. Например, для базы данных DB2 Universal Database for OS/390 используемое имя должно совпадать с именем в поле загрузочного набора данных (Boot Strap Data Set, BSDS) "LOCATION=имя_положения", которое выдается также в сообщении DSNL004I (LOCATION=положение) при запуске DDF. Смотрите также разделы "Понятие базы данных" на стр. 4 и Глава 6, "Изменение каталогов базы данных" на стр. 75.

В руководстве Быстрый старт по DB2 Connect приводятся также примеры исправления каталогов DB2. Посмотрите разделы про изменение каталогов DB2 в главах, где описывается конфигурация SNA, или главу "Конфигурирование баз данных хоста и AS/400 для DB2 Connect" и раздел "Конфигурирование соединения TCP/IP".

Правильные команды для узла APPC или APPN:

```
db2 catalog appc node <имя_узла> remote <имя_симв_назначения> security program
db2 catalog dcs database <локальное_имя> as <реальное_имя_базы>
db2 catalog database <локальное_имя> as <алиас> at node <имя_узла>
authentication dcs
```

Правильные команды для узла TCP/IP:

```
db2 catalog tcPIP node <имя_узла> remote <имя_или_адрес_хоста>
server <номер_порта_или_имя_службы>
db2 catalog dcs database <локальное_имя> as <реальное_имя_базы>
db2 catalog database <локальное_имя> as <алиас> at node <имя_узла>
authentication dcs
```

После этого, чтобы связаться с базой данных, используется команда:

```
db2 connect to <алиас> user <имя_пользователя> using <пароль>
```

SQL30073 с кодом возврата 119C при выполнении оператора CONNECT

Признак

Получено сообщение SQL30073 с кодом возврата 119C. Это происходит, когда база данных сервера назначения не поддерживает кодовую страницу, используемую клиентом DB2 (соединяющимся через DB2 Connect). Кодовая страница определяется конфигурацией операционной среды, в которой работает клиент DB2.

Дополнительную информацию смотрите в руководстве *Administration Guide*.

Решение

Эту ошибку часто можно устранить, внося исправление на системе базы данных сервера назначения. Свяжитесь с вашей обслуживающей организацией, чтобы получить и применить исправления, рекомендованные в этом случае.

В качестве временной меры пользователь может переопределить используемую по умолчанию кодовую страницу, задав переменную среды DB2CODEPAGE. Посмотрите используемую национальную версию или задайте DB2CODEPAGE=850.

На платформах UNIX пользователь может переключиться на другую кодовую страницу, задав новое значение переменной среды LANG.

SQL30081N с кодом возврата 1

Признак

Признак этой ошибки – получение следующего сообщения со смысловым кодом SNA:

```
db2 connect to <имя_базы_данных> user <ID_пользователя>
Введите пароль для <ID_пользователя>:
SQL30081N Ошибка связи.
Используемый протокол связи: "TCP/IP".
Используемый протокол связи: "APPC".
Место, где обнаружена ошибка: "".
Функция связи, обнаружившая ошибку: "smallc".
Коды ошибки протокола связи: "1", "*",
"0x10030021".
SQLSTATE=08001
```

Решения

В данном примере смысловой код – 10030021.

Ниже приводятся часто встречающиеся с этим сообщением смысловые коды и предлагаемые в каждом случае решения:

1.

SQL30081N с кодом возврата 1 и смысловым кодом SNA 0877002C

Указано неверное сетевое имя.

2.

SQL30081N с кодом возврата 1 и смысловым кодом SNA ffff0003

Указан неверный адрес MAC или связь SNA неактивна.

3.

SQL30081N с кодом возврата 1 и смысловым кодом SNA 10030021

Не совпадает тип LU.

4.

SQL30081N с кодом возврата 1 и смысловым кодом SNA 084B6031

Задано нулевое значение MAXDBAT в DSNZPARM (в DB2 for MVS или DB2 for OS/390)

Другие советы:

1. При создании профиля локального LU определите это LU как LU по умолчанию. Например, в панели списка SNA Feature в CM/2 нужно:
 - Включить переключатель 'Использовать это локальное LU как алиас локального LU по умолчанию', или
 - Задать в профиле или в переменной среды APPCLLU в системе шлюза DB2 Connect Enterprise Edition имя локального LU. В системах OS/2, например, для этого надо отредактировать файл CONFIG.SYS, а в системах Windows NT это делается через панель управления.
2. Проверьте, запущена ли SNA на шлюзе DB2 Connect.
3. Если используется DB2 for MVS или DB2 for OS/390, проверьте, запущено ли адресное пространство DDF и работает ли DB2.

SQL30081N с кодом возврата 2

Признак

Получено сообщение SQL30081N с кодом возврата 2 и смысловым кодом SNA 08120022.

Решение

Для параметра NUMILU в NCP (со стороны хоста), возможно, задано значение по умолчанию (0). Проверьте, так ли это. При необходимости измените определение NCP и повторите попытку, после того как изменение вступит в силу.

SQL30081N с кодом возврата 9

Признак

Признак – получение следующего сообщения (смысловый код SNA в этом случае не требуется):

```
db2 connect to <база_данных>  
user <ID_пользователя>
```

```
SQL30081N Ошибка связи.  
Используемый протокол связи: "TCP/IP".  
Используемый протокол связи: "APPC".  
Место, где обнаружена ошибка: "".  
Функция связи, обнаружившая ошибку: "cmsend".  
Коды ошибки протокола связи: "9", "*",  
"0x10086021".  
SQLSTATE=08001
```

Решение

Причина ошибки заключается в том, что имя программы транзакции (TPNAME) неверно определено в системе DB2 Connect. Например, конфигурация SNA изменена, но еще не проверена на шлюзе DB2 Connect. Подробности смотрите в руководствах *DB2 Connect Enterprise Edition for OS/2 and Windows Quick Beginnings* или *DB2 Connect Personal Edition. Быстрый старт*.

SQL30081N с кодом возврата 10

Признак

Признак – получение следующего сообщения (смысловый код SNA не требуется):

```
SQL30081N Ошибка связи.  
Используемый протокол связи: "TCP/IP".  
Используемый протокол связи: "APPC".  
Место, где обнаружена ошибка: "".  
Функция связи, обнаружившая ошибку: "cmrcv".  
Коды ошибки протокола связи: "10", "*", "*".  
SQLSTATE=08001
```

Решение

Убедитесь, что DB2 установлена правильно.

Если используется шлюз DB2 Connect для OS/2, при неправильном задании имени программы транзакции можно получить следующее сообщение:

```
Коды ошибки протокола связи: "10", "*", "0x084C0000".  
SQLSTATE=08001
```

В SM/2 в этом случае имя надо определить так:

```
Transaction program name      = 'tpname'    (user defined)  
OS/2 program path and file name = notused
```

и (на следующем экране конфигурирования SM/2)

```
Presentation type - background  
Operation type - Queued, operator preloaded
```

SQL30081N с кодом возврата 20

Признак

```
SQL30081N Ошибка связи.  
Используемый протокол связи: "TCP/IP".  
Используемый протокол связи: "APPC".  
Место, где обнаружена ошибка: "".  
Функция связи, обнаружившая ошибку: "xcstp".  
Коды ошибки протокола связи: "20", "*", "*".  
SQLSTATE=08001
```

Решение

Убедитесь, что в системе DB2 Connect запущена подсистема SNA.

SQL30081N с кодом возврата 27

Признак

Получено сообщение SQL30081N с кодом возврата 27 и смысловым кодом SNA 800Axxxx.

Решение

Слишком длинный блок информации о пути (PIU) VTAM.

SQL30081N с кодом возврата 79

Признак

SQL30081N Ошибка связи.
Используемый протокол связи: "TCP/IP".
Используемый интерфейс API связи: "SOCKETS".
Положение, где обнаружена ошибка: "".
Функция связи, обнаружившая ошибку: "connect".
Коды ошибки протокола связи: "79", "*", "*".
SQLSTATE=08001

Решения

Эта ошибка происходит, если удаленному клиенту не удастся соединиться со шлюзом DB2 Connect. Она может также произойти при соединении с хостом со шлюза DB2 Connect.

1. Возможно, на шлюзе DB2 Connect неверно задана переменная профиля DB2COMM. Проверьте, так ли это. Например, если DB2 Extended Enterprise Edition работает на AIX, в профиле sqllib/db2profile должна быть команда `db2set db2comm=tcip`.
2. Возможно, имя службы TCP/IP и номер порта на клиенте DB2 не соответствуют заданным на шлюзе DB2 Connect. Проверьте записи в файлах `services TCP/IP` на обоих компьютерах.
3. Проверьте, запущена ли DB2 на шлюзе DB2 Connect. Установите для `diaglevel` в конфигурации менеджера баз данных значение 4 с помощью команды:

```
db2 update dbm cfg using diaglevel 4
```

После остановки и перезапуска DB2 просмотрите файл `db2diag.log` и проверьте, была ли запущена связь DB2 TCP/IP. На выходе должно появиться сообщение следующего типа:

```
1998-02-03-12.41.04.861119 Instance:svtdbm2 Node:00
PID:86496(db2sysc) Appid:none
common_communication sqlcctcp_start_listen Probe:80
DIA3000I Поддержка протокола "TCP/IP" запущена успешно.
```

SQL30081N с кодом ошибки протокола связи 10032

Признак

```
SQL30081N Ошибка связи.
Используемый протокол связи: "TCP/IP".
Используемый интерфейс API связи: "SOCKETS".
Положение, где обнаружена ошибка: "".
Функция связи, обнаружившая ошибку: "send".
Коды ошибки протокола связи: "10032", "*", "*".
SQLSTATE=08001
```

Решение

Это сообщение об ошибке может быть получено при попытке разъединения с компьютером, связь TCP/IP с которым уже была разорвана. Исправьте ошибку с помощью подсистемы TCP/IP.

На большинстве компьютеров для исправления этой ошибки достаточно перезапустить протокол TCP/IP. В некоторых случаях может потребоваться перезапуск всего компьютера.

Часть 3. Приложения

Приложение А. Функции, введенные в предыдущих выпусках

В этой главе представлен краткий перечень улучшений в каждой версии и выпуске, начиная с самой последней версии и выпуска.

DB2 Connect, Версия 6 Выпуск 1

DB2 Connect, Версия 6.1 содержит следующие усовершенствования:

- Использование протокола связи TCP/IP
- Двухфазное принятие
- Многострочные хранимые процедуры
- Защита DCE
- Поддержка DCE Cell Directory и хост-системы
- Расширенные сообщения об ошибках защиты
- Улучшенное использование System/390 SYSPLEX
- Оптимизированный доступ к каталогам для прикладных программ ODBC и JDBC
- Новые опции BIND
- Поддержка сервера Microsoft Transaction Server
- Упрощенное управление паролями
- Расширение информации о клиенте
- Поддержка языков с двумя направлениями письма
- Слежение за программами DB2 Connect
- Усовершенствования двухфазного принятия
- Упрощенная конфигурация менеджера точек синхронизации DB2
- Поддержка новых объектов и типов данных
- DB2 Connect для Personal Communications

DB2 Connect, Версия 5 Выпуск 2

DB2 Connect Версии 5.2 содержит следующие улучшения в функциях DRDA хоста или AS/400:

- Поддержка DCE Cell Directory
- Улучшенная диагностика ошибок защиты
- Улучшенное использование System/390 SYSPLEX

- Оптимизированный доступ к каталогам для прикладных программ ODBC и JDBC
- Поддержка сервера Microsoft Transaction Server
- Новые опции BIND (DYNAMICRULES)
- Улучшения в API для задания информации о клиенте
- Поддержка SQLDescribeParam для DB2 Connect
- Поддержка языков с обратным направлением письма
- Улучшения в системном мониторе
- Улучшения в поддержке двухфазного принятия
- Упрощенная конфигурация менеджера точек синхронизации DB2
- Поддержка операционной системы SCO**
- Поддержка типов данных большое целое, большой объект, ID строки и особых пользовательских типов.

DB2 Connect, Версия 5.0

- Новые упрощенные пакеты:
 - Единый пакет DB2 Connect Personal Edition, содержащий версии этого продукта для OS/2, Windows 3.1, Windows 95 и Windows NT. В этом пакете содержится все необходимое для работы, а также копия Lotus Approach.
 - Единый пакет DB2 Connect Enterprise Edition, содержащий версии для OS/2, Windows NT и все версии UNIX.
- Возможности:
 - Новый драйвер ODBC уровня 3 со многими улучшениями
 - Обновленный драйвер JDBC для лучшей поддержки Java
 - Поддержка хранимых процедур, возвращающих наборы результатов со многими строками и несколько наборов результатов (требуется DB2 Universal Database for OS/390 Версии 5.1 или более поздней)
 - Встроенная поддержка репликации
 - Общая опция связывания: можно задать любую опцию связывания, поддерживаемую базой данных хоста
 - Использование SYSPLEX (только для DB2 Connect Personal Edition; требуется DB2 Universal Database for OS/390 Версии 5.1 или более поздней)
- Удобство использования:
 - Новый метод установки
 - Существенно упрощенное конфигурирование соединений баз данных TCP/IP (требуется DB2 Universal Database for OS/390 Версии 5.1 или более поздней или DB2 Universal Database for AS/400 Версии 4.2)

- Встроенная поддержка SNA с удобным заданием конфигурации (только для DB2 Connect Personal Edition)
- Новая удобная утилита конфигурирования соединений хоста с графическим интерфейсом
- Гораздо более простая обработка для соединений настольных систем клиентов с серверами DB2 Connect Enterprise Edition. Клиенты могут искать серверы DB2 Connect и все базы данных, определенные на каждом сервере
- Улучшенная трассировка ODBC, содержащая подробную информацию для анализа производительности
- Центр управления и другие инструменты с графическим интерфейсом, упрощающие отдельные задачи администратора баз данных
- Защита:
 - Защита DCE (требуется DB2 Universal Database for OS/390 Версии 5.1 или более поздней)
 - Возможность выполнять прикладные программы ODBC, не предоставляя каждому пользователю полномочия на работу с базовыми таблицами. Пользователи могут теперь при связывании драйвера ODBC позволить прикладным программам выполняться с полномочиями пользователя, связавшего этот драйвер ODBC.
- Производительность:
 - Более быстрый доступ к каталогу DB2 для прикладных программ ODBC
 - Уменьшение сетевого трафика:
 - Более раннее закрытие указателей
 - Отложенная подготовка
 - Уменьшение объема составных операторов SQL
 - Некоторые другие улучшения сетевого потока
 - Поддержка хранения данных на хосте в формате ASCII (требуется DB2 Universal Database for OS/390 Версии 5.1 или более поздней)
- Связь:
 - Поддержка соединений DRDA в сети TCP/IP с другими серверами прикладных программ DRDA IBM (введена поддержка протокола TCP/IP)
 - Поддержка SNA в сети TCP/IP через встроенный MPTN (требуется наличие AnyNet на хосте)
 - Поддержка дополнительных вариантов связи SNA:
 - IBM Communication Server for Windows NT
 - IBM Personal Communications
- Другие преимущества:

- Возможность инициировать транзакции с двухфазным принятием в сети TCP/IP (требуется DB2 Universal Database for OS/390 Версии 5.1 или более поздней)
- Возможность для прикладных программ рабочей станции участвовать в транзакциях с двухфазным принятием без необходимости использовать шлюз (только для TCP/IP, требуется DB2 Universal Database for OS/390 Версии 5.1 или более поздней)
- Возможность использовать добавленные в DB2 Universal Database for OS/390 средства обеспечения надежности согласования транзакций (требуется DB2 Universal Database for OS/390 Версии 5.1 или более поздней и TCP/IP)
- Другие многочисленные улучшения и исправления, влияющие на все аспекты производительности, надежности и простоты использования системы

DDCS, Версия 2 Выпуск 4

В однопользовательской версии Distributed Database Connection Services (DDCS) for Windows Версии 2.4 введены:

- Средство Data Source Setup, помогающее быстро и легко определить соединения с серверами хоста или AS/400
- Wall Data Rumba, для поддержки связи при таких соединениях
- Утилита DB2 Password Expiration Maintenance (DB2PEM), позволяющая изменить пароль DB2 for MVS/ESA, не регистрируясь в TSO
- Изменения, улучшающие производительность и гибкость DB2 Connect:
 - Отложенная подготовка, которая улучшает производительность прикладных программ ODBC и других прикладных программ, использующих динамические операторы SQL, объединяя требование PREPARE с последующим требованием, вместо того чтобы посылать их по отдельности
 - Асинхронный ODBC, увеличивающий возможности прикладных программ ODBC. В более ранних версиях при работе таких программ в некоторых ситуациях в сети могли возникать задержки при обработке длинных запросов.
 - Поддержка многопоточных прикладных программ в системах AIX и OS/2, позволяющая другим (не ODBC) прикладным программам поддерживать несколько соединений с базами данных с их собственными контекстами

DDCS, Версия 2 Выпуск 3

Новые возможности в DDCS Версии 2 Выпуска 3.1:

- Двухфазное принятие для соединений DRDA с использованием менеджера точек синхронизации LU6.2 (SPM) в системах OS/2 и AIX.

Новые возможности в DDCS Версии 2 Выпуска 3.0:

- Возможность улучшить производительность прикладной программы клиента, выполняя хранимые процедуры на серверах DB2 for MVS/ESA Версии 4.1 и DB2 Universal Database for AS/400 Версии 3.1. Смотрите раздел “Хранимые процедуры” на стр. 51.
- Возможность работать с несколькими базами данных в одной транзакции.
- Возможность улучшить производительность, объединяя операторы SQL. Смотрите разделы “Составной SQL NOT ATOMIC” на стр. 53 и “Использование утилит импорта и экспорта” на стр. 110.
- Возможность вести учет расходов, используя учетные строки. Смотрите раздел “Реализация учета оплаты в DB2 Universal Database for OS/390” на стр. 56.
- Возможность использовать много новых опций связывания при связывании прикладных программ с сервером прикладных программ DRDA. Смотрите раздел “Команда BIND” на стр. 94.
- Возможность при использовании каталога DCE объединить в централизованном хранилище информацию каталога, необходимую всем клиентам. Смотрите раздел Приложение D, “Использование служб каталога DCE” на стр. 201.
- Большая гибкость обработки SQLCODE. Смотрите раздел Глава 11, “Отображение SQLCODE” на стр. 123.
- Информация диагностики, сохраняемая в удобном формате и накапливаемая в одном месте (в журнале First Failure Service Log). Дополнительную информацию смотрите в книге *Troubleshooting Guide*.
- Для упрощения работы переменная среды DDCSSETP была заменена опциями BIND и PREPARE, например SQLERROR CONTINUE.
- Реализованы также другие улучшения производительности.

Приложение В. Листок настройки каталогов

Используйте этот листок для настройки ваших каталогов. Синтаксис команд смотрите в разделе “Изменение каталогов” на стр. 86 или в книге *Command Reference*.

Таблица 9. Параметры каталога узла

Параметр	Пример	Собственное значение
Имя узла	DB2NODE или MVSIPNOD	
Символическое имя назначения (узел APPC)	DB2CPIC	
Имя удаленного хоста (узел TCP/IP)	MVSHOST	
Сервер (имя службы или номер порта TCP/IP)	db2inst1c (или 446)	
Тип защиты	PROGRAM для узлов APPC; NONE для узлов TCP/IP	
Примечания: 1. Номер порта TCP/IP по умолчанию для DRDA – 446. 2. Если вы не уверены, что сервер баз данных хоста или AS/400 поддерживает SECURITY SOCKS, не задавайте защиту для узла TCP/IP.		

Таблица 10. Параметры каталога DCS

Параметр	Пример	Собственное значение
Имя базы данных	DB2DB	
Имя базы данных назначения	NEW_YORK3	
Реквестер прикладных программ		
Строка параметров	" ,,,,,,LOCALDATE=\\"ГГММДД\\""	

Таблица 11. Параметры системного каталога баз данных

Параметр	Пример	Собственное значение
Имя базы данных	DB2DB	
Алиас базы данных	NYC3	
Имя узла	DB2NODE	
Аутентификация	DCS	

Приложение С. Вопросы поддержки национальных языков

Особенности поддержки национальных языков DB2 Connect:

- Сообщения DB2 Connect переведены на несколько национальных языков. Информация о доступности переведенных сообщений приводится в книге *Quick Beginnings* для вашей платформы.
- DB2 Connect поддерживает многие языки и кодовые страницы. Список кодовых страниц приводится в руководстве *Administration Guide*.
- При передаче данных между DB2 Connect и сервером баз данных хоста или AS/400 они обычно преобразуются из кодовой страницы рабочей станции в CCSID хоста и наоборот.

В руководствах *DB2 Connect Quick Beginnings* приводится дополнительная информация по использованию DB2 Connect, в том числе:

- Форматы времени и даты
- Языки, поддерживаемые DB2 Connect Enterprise Edition и DB2 Connect Personal Edition
- Настройка рабочей станции DB2 Connect для конкретной национальной языковой среды
- Настройка значения CCSID хоста

Преобразование символьных данных

При передаче символьных данных между двумя компьютерами данные должны быть преобразованы в формат, с которым может работать принимающий компьютер.

Например, при передаче данных между рабочей станцией DB2 Connect и сервером баз данных хоста или AS/400 они обычно преобразуются из кодовой страницы рабочей станции в CCSID хоста и наоборот. Если на компьютерах используются разные кодовые страницы или CCSID, коды символов преобразуются из одной кодовой страницы в другую (или из одного CCSID в другой). Это преобразование выполняется на принимающем компьютере.

Символьные данные, *посылаемые* базе данных, состоят из операторов SQL и входных данных. Символьные данные, *принимаемые* от базы данных — это выходные данные. Выходные данные, которые воспринимаются как битовые (например, данные из столбца, объявленного с условием FOR BIT DATA), не преобразуются. Все остальные входные и выходные символьные данные преобразуются, если у компьютеров разные кодовые страницы или CCSID.

Например, когда DB2 Connect обращается к данным DB2 Universal Database for OS/390, происходит следующее:

1. DB2 Connect посылает оператор SQL и входные данные системе OS/390.
2. DB2 Universal Database for OS/390 преобразует данные в кодовый набор EBCDIC и обрабатывает их.
3. DB2 Universal Database for OS/390 посылает результат обратно, на рабочую станцию DB2 Connect.
4. DB2 Connect преобразует результат в кодовую страницу ASCII или ISO и возвращает пользователю.

В приведенной ниже таблице показаны преобразования, поддерживаемые между кодовыми страницами (на рабочих станциях) и CCSID (на хосте). Более подробные сведения о поддерживаемых преобразованиях кодовых страниц смотрите в руководстве *Administration Guide*.

Таблица 12 (Стр. 1 из 2). Преобразование кодовой страницы рабочей станции в CCSID хоста

CCSID хоста	Кодовая страница	Страны
037, 273, 277, 278, 280, 284, 285, 297, 500, 871, 1140–1149	437, 819, 850, 858, 860, 863, 1004, 1051, 1252, 1275	Австралия, Австрия, Албания, Бельгия, Бразилия, Великобритания, Германия, Дания, Ирландия, Исландия, Испания, Италия, Канада, Латинская Америка, Нидерланды, Новая Зеландия, Норвегия, Португалия, США, Финляндия, Франция, Швеция, Швейцария, Южная Африка
423, 875	737, 813, 869, 1253, 1280	Греция
870	852, 912, 1250, 1282	Хорватия, Чешская республика, Венгрия, Польша, Румыния, Сербия/Черногория (латиница), Словакия, Словения
1025	855, 866, 915, 1251, 1283	Болгария, Македония, Сербия/Черногория (кириллица), Россия
297	857, 920, 1254, 1281	Турция
424	862, 916, 1255	Израиль (смотрите ниже примечание 3)

Таблица 12 (Стр. 2 из 2). Преобразование кодовой страницы рабочей станции в CCSID хоста

CCSID хоста	Кодовая страница	Страны
420	864, 1046, 1089, 1256	Арабские страны (смотрите ниже примечание 3)
838	874	Таиланд
930, 939, 5026, 5035	932, 942, 943, 954, 5039	Япония
937	938, 948, 950, 964	Тайвань
933, 1364	949, 970, 1363	Корея
935, 1388	1381, 1383, 1386	КНР
1112, 1122	921, 922	Латвия, Литва, Эстония
1025	915, 1131, 1251, 1283	Беларусь
1123	1124, 1125, 1251	Украина

Примечания:

1. Кодовая страница 1004 поддерживается как страница 1252.
2. Вообще говоря, данные могут быть преобразованы из кодовой страницы в CCSID и обратно в ту же кодовую страницу без изменений. Исключения из этого правила следующие:
 - В случае кодовых страниц с набором двухбайтных символов (DBCS) могут быть потеряны некоторые данные, содержащие символы, определенные пользователем.
 - В случае однобайтных кодовых страниц, определенных внутри кодовых страниц со смешанным набором символов, а также некоторых новых однобайтных кодовых страниц символы, которые не являются общими для обеих страниц, могут быть отображены на символы подстановки и затем потеряны при обратном преобразовании к исходной кодовой странице.
3. Для языков с двумя направлениями письма фирма IBM определила несколько специальных "двунаправленных CCSID"; DB2 Connect поддерживает эти CCSID.

Если атрибуты двунаправленного письма на сервере и на клиенте отличаются, для преобразования можно использовать эти специальные CCSID.

Дополнительную информацию о них смотрите в руководстве *Administration Guide*. Их использование для соединений с хостом DRDA описано в Замечаниях по выпуску DB2 Connect.

Приложение D. Использование служб каталога DCE

При использовании служб каталога DCE Cell (CDS) информацию сервера можно хранить в CDS, а не на каждом клиенте. CDS поддерживается для всех клиентов DB2 Universal Database и для DB2 Connect Enterprise Edition на всех платформах.

Примечание: Если поддержку служб каталога DCE Cell предполагается использовать в DB2 Connect для соединений с DB2 for MVS/ESA через SNA, нужно применить исправление PTF UN73393 для DB2 for MVS/ESA, которое позволяет использовать DB2DRDA как имя удаленной программы транзакций (RTPN – remote transaction program name).

Чтобы использовать каталог DCE, нужно создать следующие объекты:

- *Объект базы данных*, содержащий информацию о базе данных
- *Объект локатора базы данных*, содержащий информацию о соединении между удаленными клиентами и рабочей станцией DB2 Connect
- *Объект маршрутизации информации*, который связывает объекты базы данных с объектами локатора базы данных

Для каждого сервера баз данных хоста или AS/400, к которому требуется доступ, перед созданием этих объектов надо:

- Убедиться, что в файл атрибутов cds на рабочей станции, с которой создаются объекты, были добавлены приведенные ниже атрибуты DCE.

В системе AIX

Имя файла */etc/dce/cds_attributes*.

В системе OS/2

Имя файла – *x:\opt\dcelocal\etc\cds_attr*, где *x*: – буква диска.

В 32-битных системах Windows

Имя файла – *x:\root\dcelocal\etc\cds_attributes*, где *x*: – буква диска, а *root* – каталог, куда вы установили DCE.

1.3.18.0.2.4.30	DB_Comment	char
1.3.18.0.2.4.31	DB_Communication_Protocol	char
1.3.18.0.2.4.32	DB_Database_Protocol	char
1.3.18.0.2.4.33	DB_Database_Locator_Name	char
1.3.18.0.2.4.34	DB_Native_Database_Name	char
1.3.18.0.2.4.35	DB_Object_Type	char
1.3.18.0.2.4.36	DB_Product_Name	char
1.3.18.0.2.4.37	DB_Product_Release	char
1.3.18.0.2.4.38	DB_Target_Database_Info	char
1.3.18.0.2.4.39	DB_Authentication	char
1.3.18.0.2.4.63	DB_Principal	char

- Убедитесь, что права доступа, с которыми вы регистрировались в DCE, достаточны для создания этих объектов. Для входа в систему UNIX или Windows 2000 можно использовать следующую команду DCE:

```
dce_login id-принципала пароль
```

Для входа в систему OS/2 можно использовать следующую команду DCE:

```
dcelogin id-принципала пароль
```

Примечание: Перед тем как соединиться с базами данных, используя эти объекты, нужно также сконфигурировать связь на сервере баз данных хоста или AS/400 и на рабочих станциях. Информацию о том, как это сделать, смотрите в соответствующей книге *Quick Beginnings* (Быстрый старт).

Создание объекта базы данных

Объект базы данных задает хост или сервер баз данных AS/400 для DB2 Connect; он должен быть определен всегда. Для каждого сервера баз данных хоста или AS/400, к которому требуется доступ, используйте для создания объекта базы данных команду DCE **cdsdp create object**. Например:

```
cdsdp create object глобальное_имя_базы_данных
```

Добавьте к этому объекту следующие атрибуты:

DB_Object_Type

D – база данных

DB_Product_Name

Продукт реляционных баз данных. Например, DB2_for_MVS или DB2_for_OS390.

DB_Native_Database_Name

Имя базы данных в системе сервера баз данных хоста или AS/400:

MVS или OS/390

Значение LOCATION

VSE или VM

Имя базы данных

OS/400

Имя реляционной базы данных

DB_Database_Protocol

DRDA

DB_Authentication

SERVER, CLIENT или DCE, как описано в разделе “Защита с использованием служб каталога DCE” на стр. 208.

DB_Principal

Если используется метод аутентификации DCE, в этом атрибуте задайте принципал DCE.

DB_Communication_Protocol

Для протокола связи между сервером DB2 Connect и сервером баз данных хоста или AS/400 задается следующая информация:

- Для протокола связи APPC:
 1. Протокол связи (APPC)
 2. Сетевой ID сервера баз данных хоста или AS/400
 3. Имя LU для сервера баз данных хоста или AS/400
 4. Имя программы транзакций для соединений с сервером баз данных хоста или AS/400. Для DB2 for MVS/ESA задайте DB2DRDA. Для остальных операционных систем задайте допустимое значение (не в шестнадцатеричном виде).
 5. Имя режима
 6. Тип защиты, как описано в разделе “Защита с использованием служб каталога DCE” на стр. 208. Например:

APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;PROGRAM

- Для протокола связи TCP/IP:
 1. Протокол связи (TCP/IP)
 2. Имя хоста TCP/IP назначения (для сервера баз данных хоста или AS/400)
 3. Номер порта TCP/IP
 4. Тип соединения (использовать SOCKS или нет – NONE). Это необязательная информация. Если она не задана, используется NONE. Например, для протокола связи TCP/IP можно указать следующее значение этого атрибута:

tcip;jaguar;19713;NONE

Чтобы создать объект базы данных с системной защитой, в файл можно записать следующие строки:

```

create object /.../cdscell1/subsys/database/DBMVS01
add object /.../cdscell1/subsys/database/DBMVS01 DB_Object_Type=D
add object /.../cdscell1/subsys/database/DBMVS01 DB_Product_Name=DB2_for_MVS
add object /.../cdscell1/subsys/database/DBMVS01 DB_Database_Protocol=DRDA
add object /.../cdscell1/subsys/database/DBMVS01 DB_Native_Database_Name=\
    NEW_YORK
add object /.../cdscell1/subsys/database/DBMVS01 DB_Authentication=SERVER
add object /.../cdscell1/subsys/database/DBMVS01 DB_Communication_Protocol=\
    APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;PROGRAM

```

Затем введите следующую команду:

```
cdscp < имя_файла
```

Примечание: Когда запись оператора в файле надо продолжить на следующей строке, вводите обратную косую черту (\).

Чтобы создать объект базы данных с защитой DCE, в файл можно записать следующие строки:

```

create object /.../cdscell1/subsys/database/DBMVS02
add object /.../cdscell1/subsys/database/DBMVS02 DB_Object_Type=D
add object /.../cdscell1/subsys/database/DBMVS02 DB_Product_Name=DB2_for_MVS
add object /.../cdscell1/subsys/database/DBMVS02 DB_Database_Protocol=DRDA
add object /.../cdscell1/subsys/database/DBMVS02 DB_Native_Database_Name=\
    NEW_YORK
add object /.../cdscell1/subsys/database/DBMVS02 DB_Authentication=DCE
add object /.../cdscell1/subsys/database/DBMVS02 DB_Principal=\
    /.../cdscell1/principal_name
add object /.../cdscell1/subsys/database/DBMVS02 DB_Communication_Protocol=\
    APPC;SPIFNET;NYM2DB2;DB2DRDA;IBMRDB;NONE

```

Затем введите следующую команду:

```
cdscp < имя_файла
```

Создание объекта локатора базы данных

Объект локатора базы данных используется, чтобы задать сервер DB2 Connect Enterprise Edition для его клиентов. Чтобы создать объект локатора базы данных для рабочей станции DB2 Connect, используйте команду DCE **cdscp create object**. Например:

```
cdscp create object глобальное_имя_объекта
```

Добавьте к этому объекту следующие атрибуты:

DB_Object_Type

L – объект локатора

DB_Communication_Protocol

Для каждого протокола связи между рабочей станцией DB2 Connect и удаленными клиентами нужно задать приведенную ниже информацию.

Ниже приводится сводка протоколов для платформ:

- OS/2: APPC, IPX, NETBIOS и TCP/IP
- 32-битные операционные системы Windows: APPC, IPX, NETBIOS, NPIPE и TCP/IP
- UNIX: APPC и TCP/IP

Для APPC:

1. Протокол связи (APPC)
2. ID сети рабочей станции DB2 Connect
3. Имя LU для рабочей станции DB2 Connect
4. Имя программы транзакций для соединений от удаленных клиентов
5. Имя режима
6. Тип защиты, как описано в разделе “Защита с использованием служб каталога DCE” на стр. 208

Для TCP/IP:

1. Протокол связи (TCP/IP)
2. Имя хоста рабочей станции DB2 Connect
3. Порт соединения, используемый рабочей станцией DB2 Connect для приема соединений от удаленных клиентов
4. Тип соединения (использовать SOCKS или нет – NONE). Это необязательная информация. Если она не задана, используется NONE.

Для IPX/SPX:

1. Протокол связи (IPXSPX)
2. Имя файл-сервера. Для прямой адресации задайте *
3. Имя объекта. Для прямой адресации используйте межсетевой адрес.

Для именованных конвейеров:

1. Протокол связи (NPIPE)
2. Имя компьютера рабочей станции DB2 Connect.
3. Имя экземпляра.

Для NETBIOS:

1. Протокол связи (NETBIOS)
2. NNAME для сервера или шлюза DB2 Connect Enterprise Edition

Например, в файл можно записать следующие строки:

```

create object /.../cdscell1/subsys/database/DBAIX01
add object /.../cdscell1/subsys/database/DBAIX01 DB_Object_Type= L
add object /.../cdscell1/subsys/database/DBAIX01 DB_Communication_Protocol=\
TCP/IP;AIX001;3700
add object /.../cdscell1/subsys/database/DBAIX01 DB_Communication_Protocol=\
APPC;SPIFNET;NYX1GW01;NYSERVER;IBMRDB;NONE

```

Затем введите следующую команду:

```
cdscp < имя_файла
```

Для 32-битных сред Windows аналогичным способом задается именованный конвейер. Например:

```
add object /.../cdscell1/subsys/database/DBAIX01 DB_Communication_Protocol=\
NPIPE;имя_компьютера;имя_экземпляра
```

Для OS/2 протокол можно задать в атрибуте DB_Communication_Protocol. Например:

- IPXSPX;файл-сервер;имя_объекта
- NETBIOS;nname

Создание объекта маршрутизации информации

Объект маршрутизации информации должен быть задан в DCE, он передается клиенту DB2. Для создания объекта маршрутизации информации используйте команду DCE **cdscp create object**. Например:

```
cdscp create object глобальное_имя_объекта
```

Добавьте атрибут **DB_Object_Type** с параметром R.

Для каждого объекта базы данных добавьте один атрибут **DB_Target_Database_Info**. Каждый атрибут **DB_Target_Database_Info** содержит следующие параметры:

База данных

Полное имя объекта базы данных, включая путь. Чтобы задать все прочие (не описанные явно) базы данных, укажите *OTHERDBS.

Протокол исходящих соединений

Протокол базы данных для соединений сервера баз данных хоста или AS/400 (DRDA)

Протокол входящих соединений

Протокол базы данных для соединений удаленного клиента (DB2RA)

Проводить аутентификацию на шлюзе

0 (Нет) или 1 (Да), как описано в разделе “Защита с использованием служб каталога DCE” на стр. 208

Строка параметров для шлюза

Строка, содержащая параметры, которые будут использоваться на шлюзе. Ее содержание зависит от конкретного шлюза. Строки шлюза DB2 Connect описаны в разделе “Каталог DCS” на стр. 77.

Локаатор базы данных

Имя объекта локаатора базы данных, представляющего рабочую станцию DB2 Connect

Например, в файл можно записать следующие строки:

```
create object /.../cdscell11/subsys/database/ROUTE1
add object /.../cdscell11/subsys/database/ROUTE1 DB_Object_Type=R
add object /.../cdscell11/subsys/database/ROUTE1 DB_Target_Database_Info=\
/.../cdscell11/subsys/database/DBMVS01;DRDA;DB2RA;0;;\
/.../cdscell11/subsys/database/DBAIX01
add object /.../cdscell11/subsys/database/ROUTE1 DB_Target_Database_Info=\
*OTHERDBS;DRDA;DB2RA;0;;\
/.../cdscell11/subsys/database/DBAIX02
```

Затем введите следующую команду:

```
cdcsp < имя_файла
```

Задание параметров конфигурации

Измените конфигурацию менеджера баз данных:

```
DB2 UPDATE DATABASE MANAGER CONFIGURATION USING
  [DIR_PATH_NAME путь]
  DIR_OBJ_NAME объект_локатора
  DIR_TYPE DCE
  [ROUTE_OBJ_NAME объект_маршрутизации]
  [DFT_CLIENT_COMM протокол]
  [DFT_CLIENT_ADPT 0-15]
```

где:

- *путь* – путь по умолчанию, используемый в полном имени базы данных назначения (по умолчанию /./subsys/database/)
- *объект_локатора* – идентифицирует клиент в пространстве имен DCE
- DIR_TYPE DCE задает каталоги DCE, использующиеся программой клиента
- *объект_маршрутизации* – имя объекта маршрутизации информации (например, ROUTE1)
- *протокол* – протокол связи между клиентом и рабочей станцией DB2 Connect (APPC или TCPIP – для UNIX; APPC, IPXSPX, NETBIOS или TCP/IP – для OS/2; APPC, TCPIP, IPXSPX, NETBIOS и NPIPE – для 32-битных операционных систем Windows)

- Для NETBIOS по умолчанию номер адаптера клиента от 0 до 15. Если используется протокол NETBIOS, и номер адаптера клиента – не 0, задайте этот номер адаптера клиента.

Примечание: Следующие переменные среды могут соответственно переопределять переменные, перечисленные выше.

- DB2DIRPATHNAME переопределяет DIR_PATH_NAME
- DB2ROUTE переопределяет ROUTE_OBJ_NAME
- DB2CLIENTCOMM переопределяет DFT_CLIENT_COMM
- DB2CLIENTADPT переопределяет DFT_CLIENT_ADPT

Каталогизация базы данных

Если база данных находится не в каталоге по умолчанию или вы хотите использовать алиас, который отличается от имени базы данных, глобальную базу данных можно каталогизировать. Для этого можно ввести в командной строке команду CATALOG GLOBAL DATABASE:

```
db2 CATALOG GLOBAL DATABASE глобальное_имя_базы_данных
AS алиас USING DIRECTORY DCE
```

Этот алиас будет использоваться любой программой, обращающейся к базе данных.

Например:

```
db2 CATALOG GLOBAL DATABASE /.../cdscell2/subsys/database/dbmvs12 AS NYC3
USING DIRECTORY DCE
```

Защита с использованием служб каталога DCE

Как администратор DB2 Connect вы можете задать, где будут проверяться имена пользователей и пароли. В каталогах DCE это можно сделать, задав:

- Тип защиты протокола связи в объекте локатора базы данных, представляющем рабочую станцию DB2 Connect. Укажите тип защиты NONE.
- Тип аутентификации объекта базы данных
- Тип защиты протокола связи в объекте базы данных
- Параметр `authenticate at gateway` в объекте маршрутизации информации

Допустимые комбинации этих значений и места проверки для каждой из этих комбинаций указаны в Табл. 13 на стр. 209 и Табл. 14 на стр. 210. DB2 Connect со службами каталога DCE поддерживает только комбинации, показанные в этих таблицах.

Таблица 13. Допустимые сценарии защиты DCE при использовании соединений APPC

Случай	Объект базы данных сервера		Объект маршрутизации	Проверка
	Аутентификация	Защита	Аутентификация на шлюзе DB2 Connect (1=да, 0=нет)	
1	CLIENT	SAME	0	Удаленный клиент (или рабочая станция DB2 Connect)
2	CLIENT	SAME	1	Рабочая станция DB2 Connect
3	SERVER	PROGRAM	0	Сервер баз данных хоста или AS/400
4	SERVER	PROGRAM	1	Рабочая станция DB2 Connect и сервер баз данных хоста или AS/400
5	DCE (конечное оборудование для передачи данных DCE)	NONE	нет	На сервере защиты DCE

Примечание: Если удаленный клиент соединяется с рабочей станцией шлюза DB2 Connect Enterprise Edition через APPC, в объекте локатора DCE шлюза задайте тип NONE.

Таблица 14. Допустимые сценарии защиты DCE при использовании соединений TCP/IP

Случай	Объект базы данных сервера	Объект маршрутизации	Проверка
	Аутентификация	Аутентификация на шлюзе DB2 Connect Enterprise Edition (1=да, 0=нет)	
1	CLIENT	0	Удаленный клиент (или рабочая станция DB2 Connect)
2	CLIENT	1	Рабочая станция DB2 Connect
3	SERVER	0	Сервер баз данных хоста или AS/400
4	НЕПРИМЕНИМО	НЕПРИМЕНИМО	Нет
5	DCE	НЕПРИМЕНИМО	На сервере защиты DCE

Ниже каждая комбинация описывается более подробно:

- В первом варианте имя пользователя и пароль проверяются только на удаленном клиенте. (Для локального клиента имя пользователя и пароль проверяются только на рабочей станции DB2 Connect.)

Ожидается, что аутентификация пользователя происходит там, где он первоначально регистрируется. По сети посылается ID пользователя, но не пароль. Этот тип защиты используется, только если на всех рабочих станциях клиентов есть адекватные средства защиты.

- Во втором случае имя пользователя и пароль проверяются только на рабочей станции DB2 Connect. Пароль посылается по сети с удаленного клиента на сервер DB2 Connect, но не на сервер баз данных хоста или AS/400.
- В третьем случае имя пользователя и пароль проверяются только на сервере баз данных хоста или AS/400. Пароль передается по сети от удаленного клиента на сервер DB2 Connect и от рабочей станции DB2 Connect на сервер баз данных хоста или AS/400.
- В четвертом случае имя пользователя и пароль проверяются и на рабочей станции DB2 Connect, и на сервере баз данных хоста или AS/400. Пароль передается по сети от удаленного клиента на сервер DB2 Connect и от сервера DB2 Connect на сервер баз данных AS/400.

Поскольку проверка производится в двух местах, на сервере DB2 Connect и на сервере баз данных хоста или AS/400 должны быть одинаковые наборы имен пользователей и паролей.

- В пятом случае разрешение DCE получается от сервера защиты DCE.

Примечания:

1. В системах AIX все пользователи, использующие тип защиты SAME, должны принадлежать группе AIX **system**.
2. В системах AIX с удаленными клиентами экземпляр программного продукта DB2 Connect, запускаемый на сервере DB2 Connect должен принадлежать группе AIX **system**.
3. Доступ к серверу баз данных хоста или AS/400 контролируется их собственными механизмами или подсистемами защиты, например, VTAM и RACF (Resource Access Control Facility – утилита управления доступом к ресурсам). Доступ к защищенным объектам баз данных управляется операторами SQL **GRANT** и **REVOKE**.

Приложение Е. Связывание утилит для клиентов ранних версий

Если у вас есть удаленные клиенты предыдущих версий, возможно, вам понадобится связать утилиты этих клиентов с сервером баз данных хоста или AS/400:

- Если старый клиент использовался с предыдущей версией DB2 Connect на том же сервере баз данных хоста или AS/400, никаких дополнительных действий не требуется.
- Если старый клиент использовался без DB2 Connect (например, если были соединены несколько компьютеров с OS/2, но соединений с сервером баз данных хоста или AS/400 не было), выполните следующие действия:

1. Если у вас есть клиенты DB2 for OS/2 Версии 1.0 или 1.2, создайте файл списка связывания, содержащий следующие строки:

```
sqlabind.bnd+
sqlueiwi.bnd+
sqluigsi.bnd+
sqluiici.bnd+
sqluiict.bnd+
sqluexpm.bnd+
sqluimpm.bnd+
sqlurexp.bnd+
sqlarxcs.bnd+
sqlarxrr.bnd+
sqlarxur.bnd
```

и скопируйте каждый из этих файлов связывания с одного из ваших клиентов на рабочую станцию DB2 Connect.

2. Если у вас есть Client Application Enabler Версии 1.0 или 1.2, создайте файл списка связывания со следующими строками:

```
db2ajgrt.bnd+
db2clics.bnd+
db2clpcs.bnd+
db2clprp.bnd+
db2clpur.bnd+
db2ueiwi.bnd+
db2uigsi.bnd+
db2uiici.bnd+
db2uiict.bnd+
db2uexpm.bnd+
```

```
db2uimpn.bnd+
db2urexp.bnd
```

и скопируйте каждый из этих файлов связывания с одного из ваших клиентов на рабочую станцию DB2 Connect.

3. На сервере DB2 Connect свяжите каждый файл списка связей с каждым сервером баз данных хоста или AS/400. Введите команды, аналогичные приведенным ниже:

```
db2 connect to алиас_базы_данных user ID_пользователя using пароль
db2 bind путь@файл_связывания.lst blocking all
      sqlerror continue messages файл_связывания.msg grant public
db2 connect reset
```

где *алиас_базы_данных*, *ID_пользователя* и *пароль* относятся к серверу баз данных хоста или AS/400, *файл_связывания* – имя файла списка связывания, а *путь* – каталог, в котором он находится.

При помощи опции **grant** команды **bind** можно дать права EXECUTE всем пользователям (PUBLIC) или задать идентификатор пользователя или ID группы. Если вы не воспользуетесь опцией **grant** команды **bind**, надо выполнить GRANT EXECUTE (RUN) индивидуально для каждого пакета.

Чтобы получить имена пакетов для файлов связывания, введите следующую команду:

```
ddcspkgn @файл_связывания.lst
```

Приложение F. Настройка производительности прикладных программ CLI/ODBC с помощью параметра CLISCHEMA

Этот раздел содержит новую информацию по настройке производительности прикладных программ ODBC/CLI с помощью параметра инициализации CLISCHEMA. Общие сведения о настройке производительности сетей или баз данных здесь не приводятся (смотрите Глава 12, “Производительность” на стр. 129). В этом приложении обсуждаются следующие вопросы:

- “Среда назначения”
- “CLI/ODBC”
- “Параметр инициализации DB2 CLISCHEMA” на стр. 216
- “Рекомендуемый подход” на стр. 219
- “Дополнительные советы и указания” на стр. 219
- “Оптимизатор каталога db2ocat” на стр. 220
- “Дополнительные источники информации” на стр. 220

Среда назначения

Приведенная информация предназначена прежде всего для пользователей DB2 Universal Database for OS/390; средой применения рекомендаций может быть:

- Прикладная программа CLI/ODBC, выполняемая на клиенте DB2 Universal Database
- DB2 Connect версии 5 или более поздней (Personal Edition или Enterprise Edition)
- DB2 Universal Database for OS/390 версии 5.1 или более поздней (или DB2 for MVS/ESA, если не указано иное)

CLI/ODBC

CLI/ODBC – это интерфейс прикладного программирования SQL, который можно вызывать из ваших прикладных программ баз данных. Он передает операторы динамического SQL как вызовы функций базы данных. В отличие от встроенного SQL он не требует использования переменных хоста или прекомпилятора.

При использовании CLI/ODBC прикладная программа должна прежде всего вызвать SQL для некоторых таблиц системного каталога в базе данных назначения, чтобы получить информацию о содержимом других баз данных.

Прикладные программы CLI/ODBC всегда обращаются так к таблицам системного каталога. Для сбора информации о базе данных, с которой устанавливается соединение, можно использовать десять вызовов API. Это следующие вызовы API:

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns

Дополнительные сведения об этих вызовах API и таблицах смотрите в книге *CLI Guide and Reference*.

По умолчанию при соединении с базой данных прикладная программа CLI/ODBC посылает таблицам системного каталога запрос на информацию обо *всех* таблицах в этой базе данных. В большой системе это может сильно увеличить сетевой трафик и значительно задержать запуск прикладной программы.

Параметр инициализации DB2 CLISCHEMA

В DB2 Universal Database есть несколько параметров инициализации CLI/ODBC, которые позволяют ограничить объем данных, возвращаемых начальными вызовами API на этапе сбора информации, после того как с базой данных в первый раз установлено соединение. Эти параметры можно задать:

1. Редактированием вручную файла db2cli.ini
2. Изменением параметров ODBC/CLI для базы данных с помощью ассистента конфигурирования клиента (на платформах, где он доступен)
3. Изменением конфигурации CLI базы данных с помощью интерфейса командной строки DBA

Это следующие параметры:

- DBNAME
- TABLETYPE
- SCHEMALIST
- SYSSHEMA
- CLISCHEMA

Информацию об этих параметрах, за исключением CLISCHEMA, можно найти в диалоговой справке CLI/ODBC и в книге *CLI Guide and Reference*. Все дальнейшее обсуждение относится только к использованию CLISCHEMA.

В книгу *CLI Guide and Reference* надо добавить следующее описание параметра CLISCHEMA:

db2cli.ini Синтаксис параметра: CLISCHEMA = схема_cli

Значение по умолчанию: Альтернативная схема не задается.

Закладка параметров DB2 CLI/ODBC: Отсутствует

Замечания по использованию

Опция CLISCHEMA указывает альтернативную схему, таблицы и набор индексов для поиска вместо схем SYSIBM (или SYSTEM, QSYS2) при поступлении вызовов DB2 CLI и функций каталога ODBC для получения информации каталога.

Например, если задано CLISCHEMA='SERGE', внутренние вызовы API CLI/ODBC, которые обычно обращаются к системным таблицам, будут вместо этого обращаться к следующим пользовательским таблицам:

- SERGE.TABLES
- SERGE.COLUMNNS
- SERGE.SPECIALCOLUMNNS
- SERGE.TSTATISTICS
- SERGE.PRIMARYKEYS
- SERGE.FOREIGNKEYS
- SERGE.TABLEPRIVILEGES
- SERGE.COLUMNTABLES
- SERGE.PROCEDURES
- SERGE.PROCEDURESCOLUMNNS

Эти пользовательские таблицы должны быть созданы администратором базы данных до использования CLISCHEMA.

Примечание: Поддержку CLISCHEMA обеспечивает DataPropagator, поэтому администратор базы данных может выполнить указанную задачу тремя способами:

1. При помощи db2cli.exe на клиенте
2. Автоматически на сервере с помощью DataPropagator
3. Вручную на сервере

Ниже объясняется, как сделать это на клиенте.

Утилиты db2cli и bldschem

Утилиту для создания пользовательских таблиц, которые требует CLISCHEMA, можно вызвать недокументированной командой bldschem интерфейса командной строки; интерфейс командной строки запускается вызовом /samples/cli/db2cli.exe. Документация по db2cli.exe (команда bldschem в ней не описана!) содержится в файле /samples/cli/INTCLI.DOC.

Например, чтобы построить набор пользовательских таблиц, необходимый для работы с CLISHEMA='SERGE' для таблицы STAFF, принадлежащей владельцу (создателю) схемы USERID в базе данных SAMPLE, следует после команды db2start и регистрации базы данных на ODBC/CLI выполнить команду:

```
db2cli < addstaff.txt
```

где "addstaff.txt" содержит следующий сценарий:

```
opt callerror on
opt echo on
quickc 1 1 sample userid password
#
# Повторить следующую строку для каждой добавляемой таблицы.
#
bldschem 1 SERGE USERID STAFF
#
# Выход
#
killenv 1
```

Этот сценарий создает набор перечисленных выше таблиц SERGE.* с индексами, заполненных данными таблиц системного каталога для таблицы USERID.STAFF. Например, в таблице SERGE.TABLES появится новая строка для каждой соответствующей записи. Дополнительные вызовы bldschem приводят к добавлению записей к существующим таблицам SERGE.* с заменой прежних строк новыми.

Синтаксис команды bldschem:

```
bldschem <номер_хэндла> <значение_CLISHEMA> <владелец_схемы> <имя_таблицы>
```

Где:

- <номер_хэндла> должен быть 1
- <значение_CLISHEMA> должно совпадать с именем схемы, заданным ключевым словом CLISHEMA
- <владелец_схемы> - создатель таблицы
- <имя_таблицы> может быть именем пользовательской таблицы, производной таблицы, алиасом, синонимом или именем системной таблицы (Разрешены символы подстановки.)

Если вы далее выполните приведенные команды примера в db2cli.exe, к пользовательским таблицам SERGE.*, созданным в предыдущем примере, будут добавлены строки с данными из таблиц системного каталога для каждой таблицы, владельцы схем которой – FRED и BERT.

```
bldschem 1 SERGE FRED %
bldschem 1 SERGE BERT %
```

Когда для параметра CLISCHEMA CLI/ODBC затем задается значение SERGE, прикладные программы ODBC/CLI при работе с базой данных SAMPLE будут вместо таблиц системного каталога обращаться к набору таблиц SERGE.*.

Рекомендуемый подход

В большинстве сред, работающих в промышленном режиме, поиск по умолчанию таблиц системного каталога может возвращать очень много данных, поэтому каждый раз, когда прикладная программа CLI/ODBC открывает базу данных, может происходить значительная задержка. Даже в типичной тестовой базе данных такая задержка вполне может достигать 25 секунд.

Вначале необходимо измерить эту задержку до того, как задано одно из упомянутых выше ключевых слов CLI, не забыв сделать поправку на время соединения. Особенно важно учесть длинные задержки, которые могут возникать, когда клиент DB2 впервые устанавливает соединение с новой базой данных – автоматическое связывание часто может занимать несколько минут.

Ваши дальнейшие действия зависят от структуры ваших данных и вашей организации. В некоторых случаях можно использовать комбинации ключевых слов DBNAME, SCHEMALIST и TABLETYPE, чтобы ограничить поиск для определенной прикладной программы или группы программ. Например, если клиенты DBA обычно обращаются к таблицам под определенными DBNAME и схемой, эти таблицы легко можно указать.

CLISCHEMA обеспечивает наивысшую производительность для большинства пользователей. По этой причине можно в целом рекомендовать использовать CLISCHEMA в средах промышленного режима, поскольку значительно легче создавать и модифицировать таблицы пользователя CLISCHEMA через интерфейс командной строки (db2cli.exe) и его команду bldschem. Смотрите также раздел “Оптимизатор каталога db2ocat” на стр. 220.

Дополнительные советы и указания

Ключевое слово CLISCHEMA следует добавить в файл db2cli.ini, включив его в раздел для имени DSN или в раздел common. Начало раздела обозначается текстом в квадратных скобках. Раздел COMMON начинается заголовком "COMMON" в квадратных скобках. Обратите внимание на то, что ключевые слова и имена разделов регистронезависимы.

При установлении соединения каждое из возможных ключевых слов ищется сначала в разделе под именем DSN, а затем, если оно не найдено, в разделе COMMON. Это выполняется как для ключевых слов DSN, так и для глобальных (клиентских) ключевых слов.

Кроме того, можно использовать ключевое слово DBALIAS для создания различных DSN (источников данных ODBC), которые ссылаются на ту же самую базу данных. (Длина имени DSN – до 255 символов, оно отображается на имя базы данных из 8 символов.)

В приведенном ниже примере каждый раз, когда пользователь соединяется с TESTDB или любым DSN, не упомянутым в этом файле, будет использоваться clischema=ODBCCAT. При соединении с TestDBcat2 используется clischema=odbccat2, но соединение с базой данных testdb сохраняется.

Файл примера db2cli.ini:

```
[TESTDB]

[COMMON]
clischema=odbccat

[TestDBcat1]
DBALIAS=testdb
clischema=odbccat1

[TestDBcat2]
DBALIAS=testdb
clischema=odbccat2
```

Оптимизатор каталога db2ocat

В 32-битных операционных системах Windows используется новое средство db2ocat, которое помогает оптимизировать поиск в системном каталоге для прикладных программ ODBC и JDBC.

Оптимизатор каталога db2ocat с графическим интерфейсом можно получить в виде архива db2ocat.zip с ftp-сайта:

```
ftp://ftp.software.ibm.com/ps/products/db2/tools
```

Дополнительные источники информации

Вам могут пригодиться также следующие дополнительные источники информации:

- Статья по автоматическому распространению каталогов с помощью DataPropagator:
<http://www.ibm.com/software/data/db2/os390/odbcatl1g.html>
- Статья, описывающая ручной вариант работы без использования DataPropagator:
<http://www.ibm.com/software/data/db2/os390/odbcmanu.html>

Приложение G. Дополнительные источники информации

Другие публикации на темы, связанные с данной

Номер формы	Название книги
SG24-2006	<i>Migrating to DB2 Universal Database Version 5</i>
SG24-2213	<i>DB2 for OS/390 Version 5 Performance Topics</i>
SG24-4893	<i>DB2 Meets NT</i>
SG24-4894	<i>The Universal Connectivity Guide to DB2</i>
SG24-4693	<i>Getting Started with DB2 Stored Procedures</i>
SG24-2212	<i>DRDA Support for TCP/IP in DB2 Universal Database for OS/390 V5.1 and DB2 Universal Database V5.0</i>
SC33-0814	<i>CICS for AIX Application Programming Guide</i>
SC33-0931	<i>CICS for AIX Customization and Operation Guide</i>
GC09-2952	<i>DB2 Connect Enterprise Edition for UNIX Quick Beginnings</i>
GC09-2953	<i>DB2 Connect Enterprise Edition for OS/2 and Windows Quick Beginnings</i>
GH43-0127	<i>DB2 Connect Personal Edition. Быстрый старт</i>
GG24-4155	<i>Distributed Relational Database Architecture: Using DDCS for AIX DRDA support with DB2 for MVS/ESA and DB2 Universal Database for AS/400</i>
GG24-4311	<i>Distributed Relational Database Architecture Cross Platform Connectivity and Application</i>
SC23-2443	<i>Encina for AIX Product Family Overview</i>

Приложение Н. 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.

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 СОБЛЮДЕНИЯ ЧЬИХ-ЛИБО АВТОРСКИХ ПРАВ, 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. 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 со стороны 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 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. 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 примеры 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.DataOS/2
AS/400	OS/390
BookManager	OS/400
CICS	PowerPC
C Set++	QBIC
C/370	QMF
DATABASE 2	RACF
DataHub	RISC System/6000
DataJoiner	RS/6000
DataPropagator	S/370
DataRefresher	SP
DB2	SQL/DS
DB2 Connect	SQL/400
DB2 Extenders	System/370
Сервер OLAP DB2	System/390
DB2 Universal Database	SystemView
Distributed Relational	SystemView
Database Architecture	VM/ESA
DRDA	VSE/ESA
eNetwork	VTAM
Extended Services	WebExplorer
FFST	WIN-OS/2
First Failure Support Technology	

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

Спецсимволы

- , (запятая) в строке параметров 78
- „ (запятая запятая) в строке параметров 78
- " (двойные кавычки) в CLP для AIX 110
- * (звездочка) в CLP для AIX 110
- \ (обратная наклонная черта) в OS/2 169
- && в файле отображения SQLCODE 124

A

- ACCSEC 175
- API
 - изменение каталогов базы данных 86
- API sqlesact 57
- APPC
 - символическое имя назначения 195
- ARI (DB2 for VSE & VM) 44
- AS/400
 - DRDA 9
- ASCII
 - порядок сортировки 48
 - смешанные данные 43
- AUTHENTICATION, параметр 114
- AUTHENTICATION=CLIENT 121

B

- bldschem 217
 - синтаксис 218

C

- CCSID 199
- CDRA (Character Data Representation Architecture – Архитектура представления символьных данных) 10
- CGI, программирование
 - ограничения 26
 - преимущества 26
- CICS 42
- CLI
 - утилиты 217
- CREATE IN COLLECTION NULLID 90
- CURRENTPACKAGESET 121

D

- DataPropagator 217
- DB_Communication_Protocol, параметр 204
- DB_Native_Database_Name, параметр 202
- DB_Object_Type, параметр 202
- DB_Product_Name, параметр 202
- DB2 Connect Enterprise Edition в качестве сервера связи 24
- DB2 Connect, Версия 5.0
 - усовершенствования 190
- DB2 Connect, Версия 5.2
 - усовершенствования 189
- DB2 Connect, Версия 6.1
 - усовершенствования 189
- DB2 for MVS/ESA или DB2 Universal Database for OS/390
 - DOMAIN 76
 - RESPORT 77
 - TCPPORT 77
 - набор данных начальной загрузки 76, 77

- DB2 Universal Database for OS/390
 - DYNAMICRULES(BIND) 121
 - TCP/IP already verified 121
- db2ocat
 - оптимизатор каталога 220
- DCE (конечное оборудование для передачи данных DCE)
 - защита 113
 - информация по установке 113
 - службы каталогов 201
 - тип аутентификации 114
 - требования к программному обеспечению 113
- DCL (язык управления данными) 44
- DCS
 - тип аутентификации 114
- DDCS, Версия 2.3
 - усовершенствования 193
- DDCS, Версия 2.4
 - усовершенствования 192
- DDL (язык определения данных) 42
- DECLARE STATEMENT
 - поддержка 55
- DELETE
 - поддержка 43
- Distributed Data Management (DDM) 10
- Distributed Relational Database Architecture (DRDA)
 - архитектуры,
 - используемые 10
 - основные понятия 9
 - поток данных 10
 - публикации 19
 - реквестер прикладных программ 10
 - сервер прикладных программ 10

DML (язык манипулирования данными) 43
DSN (DB2 Universal Database for OS/390) 44

E

EBCDIC
порядок сортировки 48
смешанные данные 43

F

FD:OCA (Formatted Data Object Content Architecture – архитектура содержимого объекта форматированных данных) 10
FOR FETCH ONLY с оператором SELECT 136

G

GRANT, оператор
защита 117, 211

I

IBM SQL 6
IBM WebSphere
обзор 30
ID авторизации (монитор) 102
ID кодовой страницы (монитор) 104
ID программы хоста (монитор) 102
ID программы–клиента (монитор) 103
ID продукта клиента (монитор) 104
ID продукта хоста (монитор) 104
ISO/ANS SQL92 47

J

Java
выполнение программ 68
сервер прикладных программ
использование DB2
Connect 28
JDBC
выполнение программ 68
оптимизатор каталога 220
производительность
прикладных программ 215

M

Microsoft ODBC Driver
Manager 63
Microsoft Windows, прикладные программы 24
MSA (SNA Management Services Architecture – архитектура служб управления SNA) 11
MVS
DRDA 9

N

Net.Data
обзор 29
свойства 30
NNAME клиента (монитор) 103
NULLID для OS/400 90

O

ODBC
выполнение программ 62
интерфейс 24
производительность
прикладных программ 215
регистрация для менеджера драйверов 64
ODBC/CLI
оптимизатор каталога 220
odbcad32.exe 63
OS/390
DRDA 9

OS/400
DRDA 9

P

Password Expiration Management (PEM) 82
PIU 158

Q

QSQ (DB2 Universal Database for AS/400) 44

R

RACF 211
resource access control facility (RACF) 117, 211
ROLLBACK WORK RELEASE
не поддерживается 56

S

SECCHK 175
SET CURRENT
PACKAGESET 121
SOCKS
обязательные переменные
среды 76
SQL
динамический 137
статический 137
SQL/DS
DRDA 9
SQL92 47
SQLCODE
автономная 47
отображение 123
поле в SQLCA 168
различия между
платформами 49
SQLDA, оптимальный
размер 137
SQLSTATE
автономная 47
в поле SQLERRMC
SQLCA 54

SQLSTATE (продолжение)

- коды классов 124
- различия 49

T

TCP/IP

- ACCSEC 175
 - DOMAIN 76
 - RESPORT 77
 - SECCHK 175
 - TCPPORT 77
 - защита уже проверена 121
 - имя службы 76
 - имя службы или номер порта 195
 - имя удаленного хоста 76, 195
 - имя хоста 195
 - конфигурирование связей хоста 22
 - порт ресинхронизации 77
- Tuxedo
- с DB2 Connect 39

V

VALIDATE RUN

- поддержка DB2 Connect 45
- virtual telecommunications access method (VTAM) 117, 211

VM

- DRDA 9

VSE

- DRDA 9
- VSE/ESA VTAM 211

W

WebSphere

- Advanced Edition 31
 - Enterprise Edition 32
 - Standard Edition 31
 - обзор 30
 - свойства 30
- Windows, прикладные программы 24

A

- авторереферентные таблицы 48
- алиас DB клиента (монитор) 103
- алиас базы данных 85
- аппаратные средства
 - производительность сети 151
- атомарные составные операторы SQL
 - не поддерживаются в DB2 Connect 134
- атрибут DB_Target_Database_Info 206
- атрибут ID собрания DB2 Universal Database for AS/400 46
 - пакеты 46
- атрибуты пакетов
 - владелец 46
 - создатель 46
 - спецификатор 46
- атрибуты пользователей
 - пакеты 46
- атрибуты создателей
 - пакеты 46
- атрибуты спецификаторов
 - пакеты 46
 - различные платформы 46
- аутентификация 86, 195
 - нарушение 113

Б

- база данных
 - алиас 195
 - группировка требований 135
 - имя 195
 - настройка 146
 - объект 201
 - объект локатора 201
 - средства определения производительности 132
- блокирование 45
 - данные 135

- блокировка на уровне страницы 48
- блокировка на уровне строки 48
- блокировки
 - превышение срока ожидания 48
 - уровень страницы 48
 - уровень строки 48
- буфер отправки (трассировка) 168
- буфер приема (трассировка) 168

В

- величина VALNSPRM 175
- внешний ключ 48
- время ответа 129
- встроенный SQL 18

Г

- группировка требований
 - база данных 135

Д

- данные, блокирование 135
- данные, преобразование 148
- двойной амперсанд (&&) в файле отображения SQLCODE 124
- двухфазное принятие 13
 - порт ресинхронизации, используемый соединениями TCP/IP 77
- диагностика ошибок 163
- динамические операторы SQL 6, 137
 - поддержка DB2 Connect 41
- динамические операторы расширенного SQL
 - не поддерживается 56
- динамические указатели 46
- динамический SQL
 - CURRENTPACKAGESET 121

длинные поля 43

Е

единица работы
распределенная 13
удаленная 11

З

задание
DB2 Connect 5
запуск прикладных программ
клиент баз данных 61
запятая в строке параметров 78
запятая запятая в строке
параметров 78
защита 208
APPC 115
DCE (конечное оборудование
для передачи данных
DCE) 113
GRANT 211
GRANT, оператор 117
NONE 116
PROGRAM 116
REVOKE 211
SAME 115
оператор REVOKE 117
особенности 113
тип 115, 195
типы 76, 208
зонный десятичный тип
данных 148

И

идентификатор набора кодовых
символов (CCSID) 197
изменение каталогов базы
данных 75
импорт
ограничения 111
имя AR 78
имя базы данных 77, 85
имя базы данных
назначения 77, 195

имя базы данных назначения
AS 77
имя базы данных хоста
(монитор) 104
имя пользователя 208
имя программы (монитор) 102
имя реквестера прикладных
программ 78
имя узла 76, 85, 195
инструменты
занятость процессора 132
использование памяти 132
производительность 131
интерактивный режим ввода
(CLP) 110
интерфейс XA
определение 39
информация по
программированию 41
использование DB2 Connect
Tuxedo 39
с менеджером транзакций
стандарта XA 39
с мониторами
транзакций 36
сценарии 21
источник данных 12
исходящий последовательный
номер (монитор) 104

К

каскад 48
каталог DCS 75, 77
задание строки
параметров 84
имя AR 78
имя базы данных 77
имя базы данных
назначения 77
имя базы данных назначения
AS 77
имя реквестера прикладных
программ 78
параметр BIDI 82
параметр CHGPWD_SDN 82
параметр LOCALDATE 80

каталог DCS (*продолжение*)
параметр SYSPLEX 80
каталог
SYSIBM.SYSPROCEDURES
(OS/390) 51
каталог баз данных
DCS 75
системный каталог баз
данных 75
узел 75
каталог узла 75, 76
каталоги базы данных
обновление 75
ключевые слова
CLISchema 216, 217, 219
DBALIAS 220
ключи
внешний 48
первичный 48
код возврата SQL1338 76
код страны
в поле SQLERRMC
SQLCA 44
кодовая страница 197
в поле SQLERRMC
SQLCA 44
исключительные ситуации
преобразования 199
кодový набор
в поле SQLERRMC
SQLCA 44
команда ACCRDB 174, 175
команда ACCRDBRM 174, 175
команда BIND
синтаксис 94
команда commit 176
команда EXCSAT 174
команда EXCSATRD 174, 175
команда EXCSQLSTT 55
команда FORCE 45, 102
команда LIST DCS
APPLICATIONS 102
команда quit (CLP) 110
команда REBIND PACKAGE
(CLP) 94
команда ROLLBACK
статически связанная 137

- команда terminate (CLP) 110
 команда атрибутов сервера
 обмена 174
См. также команда EXCSAT
 команда доступа RDB 174
См. также команда ACCRDB
 команды
 ACCRDB 174, 175
 ACCRDBRM 174, 175
 BIND 94
 EXCSAT 174
 EXCSATRD 174, 175
 EXCSQLSTT 55
 FORCE 45
 quit 110
 REBIND PACKAGE 94
 terminate 110
 принятие 176
 конкуренция за системные
 ресурсы 153
 конфигурирование
 DB2 Connect 5
 драйвер ODBC 65, 67
 концентратор XA
 примеры 143
 концентратор соединений
 затраты на соединение 142
 обзор 141
 ограничения 145
 параметры
 конфигурации 143
 поддержка транзакций
 XA 143
 примеры 143
 критические участки
 транзакции 131
- Л**
 логика предикатов 135
- М**
 маршрутизация требований 11
 мастер по многоузловому
 изменению 18
- мастера
 многоузловое изменение 18
 менеджер транзакций стандарта
 XA
 менеджер ресурсов стандарта
 XA 39
 определение 39
 многоузловое изменение
 поддержка 54
 многоузловые изменения 13
 тестирование 19
 Центр управления 18
 модель распределенной
 обработки транзакций (DTP)
 X/Open
 обзор 39
 монитор системы баз
 данных 97
 мониторинг
 соединений на шлюзе DB2
 Connect 97
 мониторы транзакций
 примеры 38
- Н**
 набор данных начальной
 загрузки в DB2 for MVS/ESA
 или DB2 Universal Database for
 OS/390
 параметры BSDS 76, 77
 настройка
 база данных 146
 производительность
 прикладных программ 215
 производительность
 сети 149
 НЕ АТОМАРНЫЕ составные
 операторы SQL 134
 неоднозначные указатели 45
 неявное соединение 44
 номер раздела 55
- О**
 обработка транзакций
 характеристики 36
- общие операторы SQL 6
 объект EXTNAM 174
 объект RDBNAM 174
 объект SRVNAM 174
 объект маршрутизации
 информации 201
 ограничения
 импорт и экспорт 111
 концентратор
 соединений 145
 однозначные указатели 45
 оператор ACQUIRE 55
 оператор CALL USING
 DESCRIPTOR (OS/400) 51
 оператор COMMIT
 статически связанный 137
 оператор COMMIT WORK
 RELEASE
 не поддерживается 55
 оператор CREATE STORGROUP
 поддержка 42
 оператор CREATE
 TABLESPACE
 поддержка 42
 оператор DECLARE 55
 оператор DECLARE
 PROCEDURE (OS/400) 51
 оператор DESCRIBE 55, 137
 поддержка 55
 оператор EXECUTE
 IMMEDIATE 137
 оператор INSERT
 поддержка 43
 оператор LABEL ON 55
 оператор PREPARE 137
 поддержка 56
 оператор PUT
 не поддерживается 56
 оператор REVOKE
 защита 117, 211
 оператор 44
 оператор ROLLBACK 45
 оператор SELECT 136, 137
 поддержка 43
 оператор SET CURRENT
 поддержка 56

оператор UPDATE
поддержка 43

операторы
ACQUIRE 55
COMMIT 137
COMMIT WORK
RELEASE 55
CREATE STORGROUP 42
CREATE TABLESPACE 42
DECLARE 55
DELETE 43
DESCRIBE 55, 137
EXECUTE IMMEDIATE 137
FOR FETCH ONLY 136
GRANT 44
INSERT 43
LABEL ON 55
PREPARE 56, 137
ROLLBACK 45, 137
SELECT 43, 136, 137
SET CURRENT 56
UPDATE 43
вызов 51
соединение 44

операторы CALL
различные платформы 51

операторы SQL
встроенные 18
категории 41
поддержка 55

опция монитора SHOW
DETAIL 103

опция прекомпиляции
LANGLEVEL SQL92E 47

особенности SQL 6

особенности конфигурации
изменение пароля 122

особенности
программирования 41
в среде хоста или AS/400 41

ответное сообщение об
окончании единицы работы
(ENDUOWRM) 176

отображение SQLCODE 123

ошибки 163

П

пакеты
атрибуты 46
создаваемые на сервере баз
данных хоста или
AS/400 92

параметр AGENTPRI 139

параметр BIDI 82

параметр CHGPWD_SDN 82

параметр CLISCHEMA 216,
217, 219

параметр D (отсоединение) 78

параметр DB_Authentication 203

параметр
DB_Communication_Protocol 203

параметр
DB_Database_Protocol 203

параметр DB_Object_Type 204

параметр DB_Principal 203

параметр DBALIAS 220

параметр DIR_CACHE 138

параметр
INTERRUPT_ENABLED
(отсоединение) 79

параметр LOCALDATE 80

параметр
MAX_COORDAGENTS 143

параметр MAXAGENTS 139,
143

параметр MAXDARI 139

параметр NOMAP 78, 123

параметр
NUM_INITAGENTS 143

параметр
NUM_POOLAGENTS 143

параметр NUMDB 139

параметр PRDDTA (DRDA) 56

параметр PRDID 175

параметр RQRIOLBK 138

параметр SYSPLEX 80

параметр конфигурации
DFT_ACCOUNT_STR 57

параметры
AGENTPRI 139
AUTHENTICATION 114
BIDI 82

параметры (*продолжение*)

DB_Authentication 203

DB_Communication_Protocol 203,
204

DB_Database_Protocol 203

DB_Native_Database_Name 202

DB_Object_Type 202, 204

DB_Principal 203

DB_Product_Name 202

DB_Target_Database_Info 206

DFT_ACCOUNT_STR 57

DIR_CACHE 138

INTERRUPT_ENABLED
(отсоединение) 79

LOCALDATE 80

MAX_COORDAGENTS 143

MAXAGENTS 139, 143

MAXDARI 139

NOMAP 78

NUM_INITAGENTS 143

NUM_POOLAGENTS 143

NUMDB 139

PRDDTA 56

PRDID 175

RQRIOLBK 138

SYSPLEX 80

пароли
службы каталога DCE 208

первичный ключ 48

передача данных
между хостом и рабочей
станцией 110

переменная среды
DB2ACCOUNT 57

переменная среды
DB2CONNECT_IN_APP_PROCESS 97

переменные среды
DB2ACCOUNT 57

перенос прикладных
программ 41

переполнения при числовых
преобразованиях 50

поддержка CCSID с двумя
направлениями письма 82

поддержка даты и часовых
поясов 80

- поддержка национальных кодовых страниц 197
- поддержка национальных языков (NLS)
 - особенности 197
 - преобразование символьных данных 197
 - смешанные данные 43, 111
- поддержка транзакций XA
 - концентратор соединений 143
- поддержка часовых поясов 80
- поддержка языков с двумя направлениями письма 199
- подкачка
 - размер блока 138
- подпрограммы
 - буферы данных 168
 - поле SQLCODE 168
 - поле SQLERRMC 44, 54
 - поле SQLERRP 44
- поле RQRIOBLK 45
- поле SQLERRMC SQLCA 44, 54
- поле SQLERRP SQLCA 44
- полномочия, необходимые для связывания 90
- пользовательская последовательность слияния 48
- пользовательские типы поддерживаются DB2 Connect 44
- поля учетной строки 56
- порядок сортировки
 - определение 48
 - последовательность слияния 48
- последовательность слияния EBCDIC и ASCII 48
- последовательный номер клиента (монитор) 103
- построитель хранимых процедур
 - обзор 52
 - свойства 52
- поток данных 10, 129
- превышение срока блокировки 48
- прекомпилятор
 - поддержка 42
 - поддержка DB2 Connect 45
- преобразование
 - символ 43
- преобразование данных 148
- CCSID 197
- двухбайтные символы 199
- исключительные ситуации 199
- кодовые страницы 197
- подстановка символов 199
- преобразование символов 43
- пререквизиты
 - DCE (конечное оборудование для передачи данных DCE) 113
- привилегия BINDADD 90
- примеры
 - концентратор XA 143
 - концентратор соединений 143
- программы
 - связывание 89
- программы CLI/ODBC
 - CURRENTPACKAGESET 121
- программы ODBC
 - CURRENTPACKAGESET 121
- программы Web
 - обзор 26
- проектирование программ 134
- производительность
 - DB2 for OS/390 147
 - инструменты 131
 - критерии настройки SNA 157
 - критические участки 131
 - настройка 147
 - настройка
 - производительности прикладных программ ODBC и JDBC 215
 - общая 129
 - параметр CLISCHEMA 215
 - прикладные программы CLI 215
- производительность
 - (продолжение)
 - процессор командной строки 137
 - размер PIU 158
 - размер RU 158
 - сетевое оборудование 151
 - сетевые средства 132
 - советы по настройке SNA 154
 - устранение неисправностей 153
 - эталонное тестирование 131
 - производительность SNA
 - советы по настройке 154
 - производительность прикладных программ
 - параметр CLISCHEMA 215
 - производительность прикладных программ CLI/ODBC
 - производительность прикладных программ 215
 - производные таблицы
 - системные каталоги 49
 - пропускная способность
 - транзакции 129
 - процессор командной строки (CLP) 6, 109
 - команда REBIND PACKAGE 94
 - производительность 137
 - прямое соединение
 - с базами данных хоста 22
 - прямой доступ к базам данных 22
 - публикации по переносимости программ 59
 - пул соединений
 - обзор 28

Р

- различия в SQLCODE и SQLSTATE 49
- различия между программными продуктами DB2 42

- различия между сервером хоста или AS/400 и рабочей станцией 55
 - размер RQRIOBLK 158
 - размер RQRIOBLK в DB2 Connect 158
 - размер RU 158
 - размер блока 138
 - разработка прикладных программ 41, 134
 - использующих ODBC 24, 62
 - при помощи клиента разработки программ DB2 24
 - распределенная среда 41
 - распределенное требование 12
 - расширения данных на сервере хоста или AS/400 43
 - регистрация
 - менеджер драйверов ODBC 64
 - реквестеры прикладных программ
 - определение DRDA 10
 - параметры 195
 - реляционная целостность 48
- С**
- связывание
 - необходимые полномочия 90
 - пакеты 92
 - утилит и программ 89
 - утилиты 62
 - серверы прикладных программ
 - конфигурация 35
 - обзор 32
 - определение DRDA 10
 - поддержка DB2 Connect 34
 - размещение 35
 - серверы связи
 - DB2 Connect Enterprise Edition 24
 - сеть
 - адаптер или контроллер связи 152
 - сеть (*продолжение*)
 - надежность 153
 - настройка 149
 - топология 152
 - трафик 152
 - символ выделения 84
 - символическое имя
 - назначения 195
 - регистрозависимость 76
 - символы переключения 111
 - символы переключения на
 - дополнительный и стандартный регистры 43
 - символьные столбцы переменной длины 111
 - синтаксис
 - bldschem 218
 - системные ресурсы
 - конкуренция 153
 - системный каталог
 - использование 49
 - системный каталог баз данных 75, 85
 - системный монитор баз данных, утилита 7
 - скорость передачи данных 129
 - производительность 151
 - служебный журнал первого отказа 167
 - смешанные данные 43
 - импорт 111
 - экспорт 111
 - собрания 46
 - события
 - трассировка 169
 - соединение
 - CONNECT без параметров 44
 - неявное соединение 44
 - оператор CONNECT RESET 44
 - оператор CONNECT TO 44
 - соединения с хостами DRDA
 - прямые с хостом DRDA 22
 - сообщение ENDUOWRM 176
 - составной SQL
 - NOT ATOMIC 53
 - составные операторы SQL
 - NOT ATOMIC 134
 - составные операторы SQL ATOMIC
 - не поддерживается 53
 - составные операторы SQL NOT ATOMIC 53
 - сохранение сведений
 - каталога 138
 - список связывания 89, 214
 - средства диагностики 167
 - средства наблюдения за использованием памяти 132
 - средства наблюдения за использованием процессора 132
 - средства определения производительности сети 132
 - стабильность на уровне указателя 48
 - статические операторы SQL 6, 137
 - поддержка DB2 Connect 41
 - строка параметров 195
 - строки с переменной длиной 43
- Т**
- территория
 - в поле SQLERRMC SQLCA 44
 - тип DSS (трассировка) 168
 - тип аутентификации CLIENT 114
 - тип аутентификации DCS_ENCRYPT 114
 - тип аутентификации SERVER 114
 - тип аутентификации SERVER_ENCRYPT 114
 - тип данных 64-битное целое (BIGINT)
 - поддерживается DB2 Connect Версии 7 44
 - тип данных CHAR 149
 - тип данных LOB
 - поддерживается DB2 Connect Версии 7 43

- тип данных ROWID
 - поддерживается DB2 Connect Версии 7 44
 - тип данных VARCHAR 149
 - тип данных с плавающей точкой 148
 - тип защиты NONE 116
 - тип защиты PROGRAM 116
 - тип защиты SAME 115
 - типы
 - ROWID 44
 - аутентификация 114
 - защита 115
 - типы аутентификации
 - CLIENT 114
 - DCE (конечное оборудование для передачи данных DCE) 114
 - DCS 114
 - DCS_ENCRYPT 114
 - SERVER 114
 - SERVER_ENCRYPT 114
 - по умолчанию 114
 - службы каталога DCE 208
 - типы данных
 - CHAR 149
 - VARCHAR 149
 - зонный десятичный 148
 - плавающая точка 148
 - преобразование 148
 - упакованный десятичный 148
 - целое 148
 - численные 43
 - транзакции
 - пропускная способность 129
- у**
- удаленная единица работы 11
 - указатели
 - динамические 46
 - неоднозначные 45
 - однозначные 45
 - упакованный десятичный тип данных 148
 - управление распределенными данными 168
 - уровень изоляции 50
 - уровни изоляции 50
 - условие GROUP BY
 - порядок сортировки 48
 - условие ORDER BY
 - порядок сортировки 48
 - усовершенствования
 - DB2 Connect, Версия 5.0 190
 - DB2 Connect, Версия 5.2 189
 - DB2 Connect, Версия 6.1 189
 - DDCS, Версия 2.3 193
 - DDCS, Версия 2.4 192
 - установка
 - DB2 Connect 5
 - устранение ошибок 163
 - переполнения при числовых преобразованиях 50
 - утилита db2cli.exe 217
 - утилита ddcstrc 167
 - выходной файл 168
 - программа на выходе 170
 - синтаксис 168
 - утилита ps (состояние процесса) 167, 174
 - утилита состояния процесса 167, 174
 - утилита трассировки 167
 - выходной файл 168
 - программа на выходе 170
 - синтаксис 168
 - утилиты
 - bldschem 217
 - db2cli 217
 - db2ocat 220
 - ddcspkgn 93, 95
 - ddcstrc 167
 - ps 174
 - импорт 110
 - связывание 61, 89
 - системный монитор баз данных 7
 - состояние процесса 174
 - трассировка 167
 - управление 6, 109
 - экспорт 110
 - утилиты импорта
 - общее описание 110
 - утилиты управления 6
 - утилиты экспорта
 - общее описание 110
 - учет оплаты
 - DB2 Universal Database for OS/390 56
 - определение 56
 - учетная строка 56
- Ф**
- файл db2cli.ini
 - настройка
 - производительности прикладных программ ODBC и JDBC 216
 - файл dcs1ari.map 123
 - файл dcs1dsn.map 123
 - файл dcs1qsq.map 123
 - файл ddc400.lst 89
 - файл ddcsmvs.lst 89
 - файл ddcstrc.dmp 169
 - файл ddcsvm.lst 89
 - файл ddcsvse.lst 89
 - файл отображения SQLCODE
 - * (звездочка) 124
 - сс 124
 - i 126
 - P 124
 - s 125, 126
 - U 124
 - W 124
 - звездочка 124
 - синтаксис 124
 - файл ядра 167
 - файловый формат PC/IXF 111
- Х**
- хранимые процедуры
 - обзор 27
 - общие 51
 - хэндл программы 102

Ц

целый тип данных 148

Ч

числовые типы данных 43

Э

экспорт

ограничения 111

элементы сообщения и

SQLCODE 123

эталонное тестирование

производительность 131

Я

язык манипулирования данными

(DML) 43

язык определения данных

(DDL) 42

язык управления данными

(DCL) 44

Как связаться с IBM

Если у вас имеется техническая проблема, пожалуйста, перед обращением к службе поддержки пользователей DB2 просмотрите еще раз и выполните действия, рекомендуемые в руководстве *Troubleshooting Guide*. В этом руководстве описано, какую информацию надо собрать, чтобы служба поддержки пользователей DB2 могла лучше помочь вам.

Чтобы получить информацию или заказать любой из продуктов DB2 Universal Database, обратитесь к представителю IBM в местном отделении или к авторизованному продавцу программных продуктов IBM.

Если вы находитесь в США, позвоните по одному из следующих номеров:

- 1-800-237-5511, чтобы обратиться в службу поддержки
- 1-888-426-4343, чтобы узнать о доступных формах обслуживания.

Информация о продукте

Если вы находитесь в США, позвоните по одному из следующих номеров:

- 1-800-IBM-CALL (1-800-426-2255) или 1-800-3IBM-OS2 (1-800-342-6672), чтобы заказать продукты или получить общую информацию.
- 1-800-879-2755, чтобы заказать публикации.

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

На страницах DB2 в WWW содержится текущая информация DB2: новости, описания продуктов, учебные планы и т.д.

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

DB2 Product and Service Technical Library содержит ответы на часто задаваемые вопросы, исправления, книги и свежую техническую информацию по DB2.

Примечание: Эта информация может быть только в английском варианте.

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

На сайте заказов International Publications приводится информация о том, как заказывать книги.

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

На этом сайте представлена программа Professional Certification Program IBM и приводится информация о сертификационных испытаниях для многих продуктов IBM, в том числе DB2.

<ftp://software.ibm.com>

Зарегистрируйтесь как аноним. В каталоге /ps/products/db2 можно найти демо-версии, исправления, информацию и инструменты для DB2 и многих других продуктов.

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

В этих группах новостей пользователи обмениваются опытом работы с продуктами DB2.

В Compuserve: GO IBMDB2

Введите эту команду, чтобы попасть на форумы IBM DB2 Family. Через эти форумы поддерживаются все продукты DB2.

Информацию о том, как связаться с IBM из других стран, смотрите в Приложении А книги *IBM Software Support Handbook*. Этот документ можно найти в Web, обратившись по адресу: <http://www.ibm.com/support/> и выбрав ссылку на IBM Software Support Handbook у нижнего края страницы.

Примечание: В некоторых странах авторизованные дилеры IBM должны обращаться не в центр поддержки IBM, а в структуры поддержки дилеров.



SH43-0130-00

