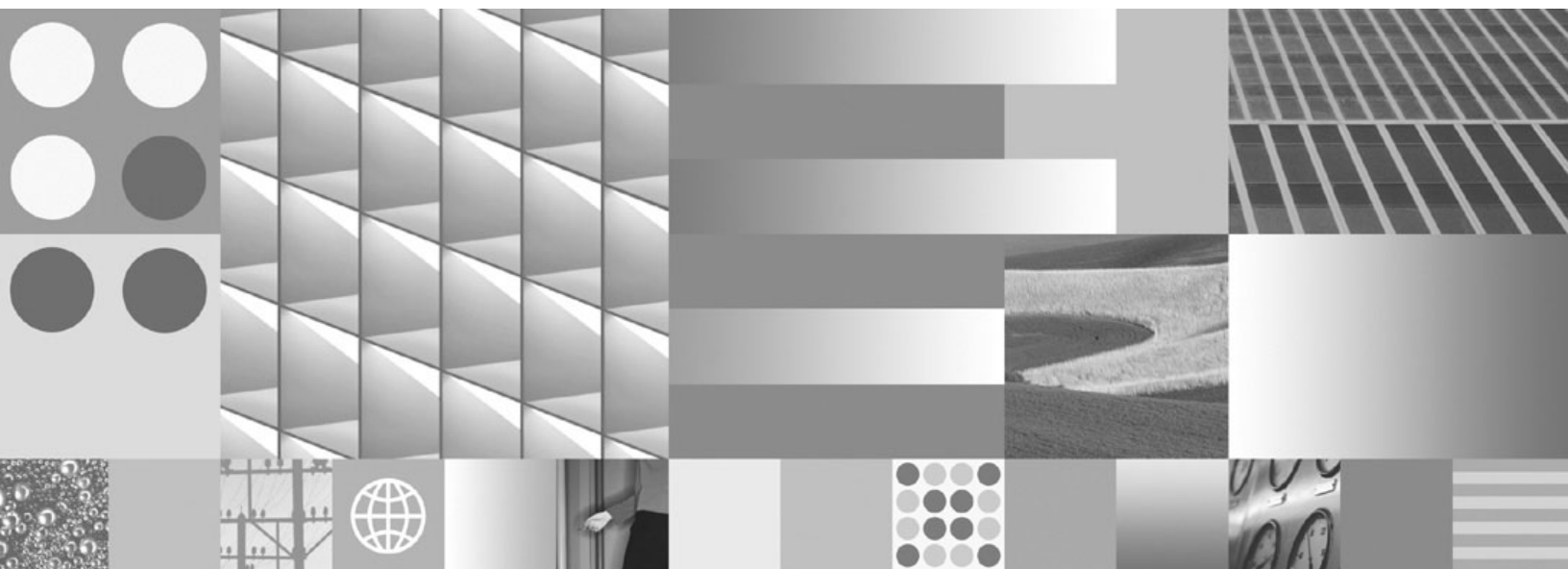


DB2 Connect, Версия 9.5



версия 9 выпуск 5

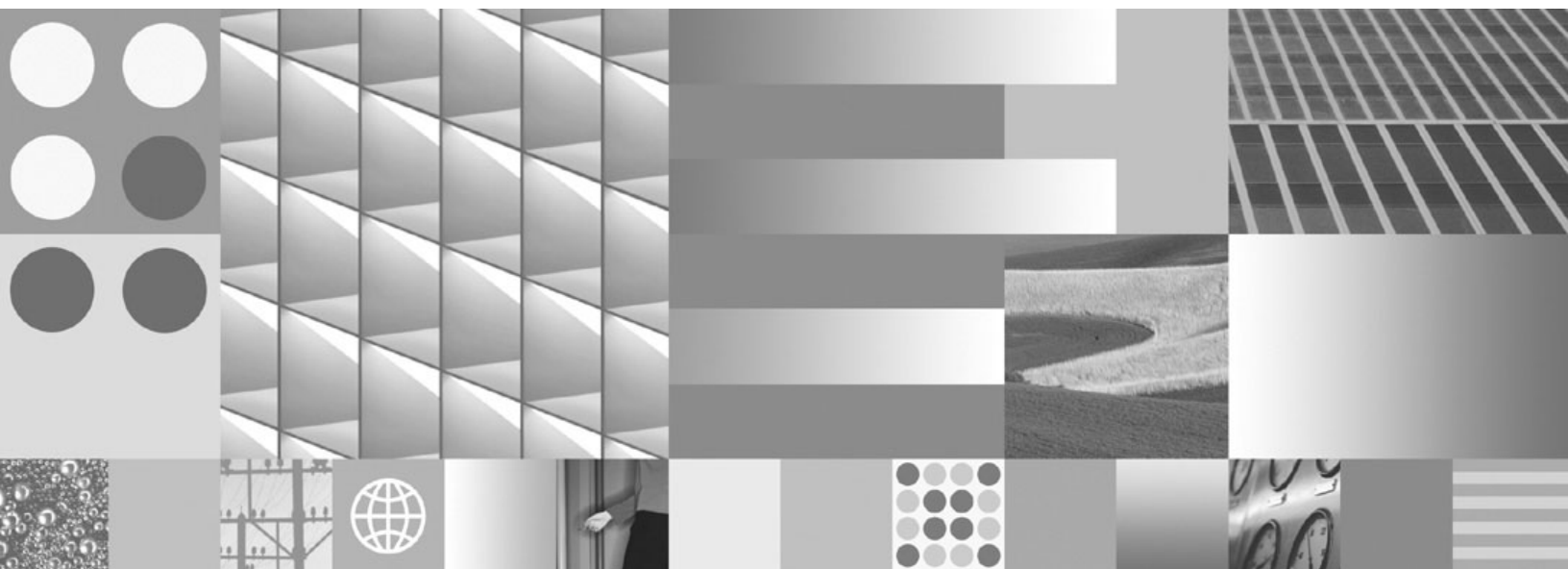


DB2 Connect. Руководство пользователя

DB2 Connect, Версия 9.5



версия 9 выпуск 5



DB2 Connect. Руководство пользователя

Замечание

Прежде чем использовать эту информацию и описанный в ней продукт, прочтите общие сведения в разделе Приложение В, “Замечания”, на стр. 167.

Замечание по изданию

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

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

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

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

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

© Copyright International Business Machines Corporation 1993, 2007. Все права защищены.

Содержание

Об этой книге	v
-------------------------	---

Часть 1. Понятия DB2 Connect . . . 1

Глава 1. DB2 Connect 3

Разновидности продукта DB2 Connect	3
Функции, введенные в Версии 9 и в предыдущих выпусках.	3
Базы данных хоста	6
DB2 Connect и операторы SQL.	6
Утилиты управления DB2 Connect	7
WebSphere Federation Server и DB2 Connect	8

Глава 2. Архитектура распределенных реляционных баз данных (DRDA) . . . 9

DRDA и доступ к данным	9
DB2 Connect и DRDA	10
Удаленная единица работы	11
Распределенное требование	12

Глава 3. Сценарии DB2 Connect . . . 15

Прямой доступ к базам данных хоста	15
Доступ к данным DB2 хоста или System i при помощи DB2 Connect Personal Edition	17
Продукты сервера DB2 Connect в качестве серверов соединений	18
DB2 Connect и программы Web	19
DB2 Connect и IBM WebSphere	21
DB2 Connect в качестве сервера прикладных программ Java	21
DB2 Connect на Web-сервере	22
DB2 Connect и серверы прикладных программ	23
DB2 Connect и мониторы транзакций	26

Часть 2. Справочник по DB2 Connect. 29

Глава 4. Изменение каталогов баз данных 31

Значения системного каталога баз данных	31
Значения каталога узла	31
Значения каталога DCS	33
Рабочий лист настройки каталогов	37
Задание нескольких записей для одной базы данных	38
Обработка данных с двумя направлениями письма	38

Глава 5. Защита DB2 Connect. 41

Доверенные соединения при помощи DB2 Connect	41
Создание и прекращение доверенного соединения при помощи CLI	42
Переключение пользователей в доверенном соединении при помощи CLI	44

Информация об аутентификации в DB2 Connect.	46
Поддержка Kerberos.	47
Сведения о защите DB2 Connect в OS/390 и z/OS	48
Дополнительные советы и замечания по организации защиты для OS/390 и z/OS	48
Типы защиты, поддерживаемые DB2 Connect	49

Глава 6. Связывание программ и утилит (DB2 Connect) 51

Глава 7. Многоузловые изменения . . 55

Установление многоузлового изменения с помощью Центра управления	56
Проверка многоузлового изменения с помощью Центра управления	56
Многоузловое изменение и менеджер точки синхронизации	57
Настройка DB2 Connect с менеджером транзакций стандарта XA.	57
Поддержка связанных транзакций в DB2 Connect	58

Глава 8. Перемещение данных с помощью DB2 Connect 59

Глава 9. Отображение SQLCODE . . . 63

Отключение отображения SQLCODE	63
Оптимизация отображения SQLCODE.	63

Глава 10. Мониторинг систем баз данных и DB2 Connect 69

Монитор соединений с удаленными клиентами	69
Отслеживание производительности с помощью монитора производительности Windows	69
Использование команд GET SNAPSHOT	70
Состояние прикладной программы DCS	72
Монитор работоспособности и оповещения	76
Обзор монитора работоспособности DB2 for z/OS	76
Запуск, остановка и обновление монитора работоспособности DB2 for z/OS	77
Просмотр, передача на выполнение и сохранение рекомендованных действий	78
Просмотр сводных данных оповещения работоспособности	81
Просмотр объектов оповещения работоспособности	83

Часть 3. Высокая доступность и DB2 Connect 85

Глава 11. Высокая доступность и сбалансированная нагрузка в соединениях с базами данных хоста . . . 87

Глава 12. Описание и установка автоматического перенаправления клиента (DB2 Connect)	89
---	-----------

Глава 13. Конфигурирование автоматического перенаправления клиента для технологии распределителя соединений клиентов	91
---	-----------

Часть 4. Настройка и DB2 Connect

Глава 14. Производительность DB2 Connect	95
---	-----------

Глава 15. Оптимизация доступа ODBC	99
---	-----------

Глава 16. Разработка программ	101
--------------------------------------	------------

Глава 17. Управление соединениями	105
--	------------

Пулы соединений	105
Концентратор соединений	107
Пулы соединений и концентратор соединений	112
Для WebSphere MQ Transaction Manager и DB2 for OS/390 требуется концентратор соединений	112

Глава 18. Поддержка Sysplex в DB2 Connect	113
--	------------

Особенности применения SYSPLEX для OS/390 и zSeries	113
Применение DB2 Sysplex	114
Требования конфигурации для Sysplex	115

Глава 19. Настройка DB2 Connect	117
--	------------

Настройка баз данных хоста	119
Особенности настройки сети	119
Конфликты из-за системных ресурсов	120
Устранение неисправностей, связанных с производительностью DB2 Connect	121
Настройка DB2 в OS/390 и z/OS	121
Увеличение скорости передачи данных DB2 Connect	121
Дополнительные блоки запроса	122
Изменение размеров окна RFC-1323	123
Преобразование данных хоста	124
Символьные типы данных	124
Сетевые устройства	124

Глава 20. Настройка производительности прикладных программ CLI/ODBC	127
--	------------

Часть 5. Устранение неисправностей	129
---	------------

Глава 21. Устранение неисправностей	131
--	------------

Сбор связанной информации	131
Ошибка при начальном соединении	131
Ошибки, возникающие после начального соединения	132
Средства диагностики	133

Глава 22. Трассировки DB2 в DB2 Connect	135
--	------------

Получение трассировки DB2 при помощи db2trc	135
Дамп файла трассировки DB2	136
Форматирование файла трассировки DB2	137

Глава 23. Файлы трассировки DRDA	139
---	------------

Утилита трассировки	139
Вывод трассировки	140
Анализ файла трассировки	140
Примеры файлов вывода трассировки	142
Информация последующего буфера для трассировки DRDA	146

Часть 6. Сообщения	149
---------------------------	------------

Глава 24. Часто встречающиеся ошибки при работе с DB2 Connect	151
--	------------

Часть 7. Приложения	155
----------------------------	------------

Приложение А. Обзор технической информации DB2	157
---	------------

Техническая библиотека DB2 в печатном виде и в формате PDF	157
Заказ печатных копий книг DB2	160
Вызов справки по SQLSTATE из командной строки	160
Доступ к различным версиям Информационного центра DB2	161
Вывод тем Информационного центра DB2 на предпочитаемом вами языке	161
Обновление Информационного центра DB2, установленного на вашем компьютере или на сервере интранета	162
Учебные материалы по DB2	163
Информация об устранении неисправностей DB2	164
Положения и условия	164

Приложение В. Замечания	167
--------------------------------	------------

Индекс	171
---------------	------------

Об этой книге

Руководство пользователя DB2 Connect содержит всю информацию, которую нужно знать, чтобы использовать продукт DB2 Connect. Понятия DB2 Connect представлены с типичным сценарием, иллюстрирующим взаимосвязи между DB2 Connect и другими компонентами рабочей среды. Рассмотрены особенности каталогов баз данных, защиты между системами, многоузловых изменений, перемещения данных и мониторинга DB2 Connect. Описана поддержка DB2 Connect высокой доступности в сетевой среде. Сведения о гарантии высокой производительности, обеспечиваемой DB2 Connect и в сети, приведены в нескольких темах по устранению возможных неисправностей.

Для кого предназначена эта книга

Вся книга или отдельные ее разделы будут полезны системным администраторам, администраторам баз данных и специалистам по связи.

Часть 1. Понятия DB2 Connect

Глава 1. DB2 Connect

DB2 Connect обеспечивает быструю и надежную связь прикладных программ электронной коммерции и других программ, работающих в операционных системах Linux, UNIX и Windows с базами данных хоста и System i.

DB2 Connect Personal Edition обеспечивает прямое соединение с серверами хоста и System i DB2, а продукты серверов DB2 Connect обеспечивают не прямое соединение, при котором клиенты могут обращаться к серверам хоста и System i DB2 через шлюз DB2 Connect. Уникальные решения комплектации и лицензирования ряда серверных продуктов DB2 Connect позволяют выбрать подходящий для вашей среды продукт.

Разновидности продукта DB2 Connect

Есть несколько редакций DB2 Connect в том числе DB2 Connect Personal Edition и продукты серверов DB2 Connect:

- DB2 Connect Enterprise Edition
- DB2 Connect Application Server Edition
- DB2 Connect Unlimited Edition for zSeries
- DB2 Connect Unlimited Edition for i5/OS

Подробную информацию о продуктах DB2 Connect смотрите по адресу <http://www.ibm.com/support/docview.wss?rs=73&uid=swg21219983>

Функции, введенные в Версии 9 и в предыдущих выпусках

В этом разделе представлен краткий перечень улучшений в каждой версии и выпуске.

Функции, введенные в DB2 Connect Версии 9

DB2 Connect Версии 9 содержит следующие усовершенствования:

- Поддержка клиентами доверенных соединений
Клиент может при помощи ODBC, XA или новых методов Java создать доверенные соединения с серверами баз данных, поддерживающими доверенный контекст (в настоящее время - только с DB2 for z/OS). После этого можно переключиться на другое имя пользователя клиента без выполнения сервером баз данных полной аутентификации нового имени.
- Поддержка типов данных BINARY, VARBINARY и DECFLOAT
DB2 for z/OS теперь поддерживает типы данных BINARY, VARBINARY и DECFLOAT. Поддержка этих типов данных добавлена в DB2 CLI и IBM Data Server Provider for .NET. Прикладные программы, использующие DB2 Connect для доступа к DB2 for z/OS, при использовании DB2 CLI и IBM Data Server Provider for .NET могут извлечь преимущества из этих новых типов данных. Новый параметр соединения `SQL_ATTR_DECFLOAT_ROUNDING_MODE` позволяет клиенту задать, какой тип округления применять, когда каким-либо операциям на стороне сервера требуется округление десятичного значения с плавающей точкой.
- Протоколы связи NetBIOS и SNA больше не поддерживаются

Заказчики, использующие эти протоколы, должны будут выполнить повторную каталогизацию своих узлов и баз данных при помощи поддерживаемого протокола, например, TCP/IP.

- Добавлена поддержка протокола связи IPv6
Добавлена поддержка Internet Protocol Версии 6 (IPv6), поэтому при соединении с серверами теперь можно использовать адреса IPv4 или IPv6.
- Снято 64-Кбайтное ограничение процессора командной строки для операторов SQL
Новое ограничение процессора командной строки для операторов SQL и команд, содержащих компоненты операторов SQL, равное 2 Мбайт, сравнимо с ограничениями для других инструментов DB2. Теперь прикладные программы, использующие DB2 Connect, могут извлечь преимущества от этого нового ограничения.
- Усовершенствования IBM Data Server Provider for .NET, в том числе поддержка .NET Framework 2.0
Эта поддержка и усовершенствования будут способствовать разработке более мощных программ .NET для использования с DB2 Connect. Вот неполный список новых возможностей:
 - Программы могут выполнять предварительную выборку данных из конкретного набора строк, не прокручивая весь набор результатов.
 - Программы могут выполнять операцию массового копирования данных.
 - Программы могут определить число операторов SQL, собираемых перед применением в качестве пакета для сервера баз данных DB2. При этом уменьшается число отдельных операций по передаче данных между прикладной программой клиента и сервером баз данных.
- Двухфазное принятие для источников данных нескольких поставщиков при использовании сервера WebSphere Federation Server
Прикладные программы DB2 Connect могут использовать WebSphere Federation Server для поиска источников данных, предлагаемых многими поставщиками IBM и другими разработчиками (не IBM).
- Поддержка истечения срока ожидания для прикладных программ базы данных
Для прикладных программ баз данных DB2 Connect можно также ограничить срок ожидания установления соединения. Особенно это полезно при недоступности сервера баз данных.
- Упрощение обновления DB2 Connect Personal Edition
DB2 Connect Personal Edition в операционных системах Windows и Linux можно обновить, предоставив соответствующий файл электронных сертификатов (Electronic Certificate File). Выполнять полную установку при обновлении больше не нужно.
- Изменения в поддержке лицензий DB2
Изменения в комплекте поставки продукта DB2 Connect - часть усовершенствований Центра лицензий и Средства управления лицензиями (команда db2licm).

Функции, введенные в DB2 Connect, Версия 8 Выпуск 2

DB2 Connect версии 8.2 содержит следующие усовершенствования:

- Автоматическая переадресация клиента
При разрыве соединения TCP/IP с сервером или сервером DB2 Connect клиент автоматически попытается переустановить его с альтернативным

сервером, если он задан. Альтернативный сервер задается на экземпляре сервера; его положение сообщается клиенту при установке соединения.

- Шифрование данных

Связь между клиентом и сервером не обеспечивает шифрования пользовательских данных, передаваемых через сеть.

Функции, введенные в DB2 Connect, Версия 8 Выпуск 1 (со всеми пакетами FixPак и уровнями модификации)

DB2 Connect Версии 8.1 содержит следующие усовершенствования:

- Поддержка более длинных операторов SQL (до 2 Мбайт)

В прикладных программах CLI и JDBC могут использоваться операторы SQL длиной до 2 Мбайт. Однако для встроенного интерфейса предел - по-прежнему 64 Кбайт.

- Диагностическая информация, определяющая происхождение операторов SQL

Позволяет определять, какая из прикладных программ поместила конкретный оператор в кэш динамических операторов SQL DB2 for z/OS.

- Массив с вводом по столбцам

Позволяет прикладным программам задавать для одного оператора SQL несколько наборов параметров.

- Мониторинг сетевого времени

Новые элементы монитора помогают лучше представлять себе активность базы данных и оценить сетевой трафик на уровне базы данных и прикладной программы.

- Поддержка указателей с динамической прокруткой DB2 CLI

Теперь в DB2 CLI поддерживаются указатели с динамической прокруткой при обращении к серверам DB2 Universal Database (UDB) for z/OS Версии 8.1 или новее.

- Поддержка eWLM

Позволяет отслеживать сквозные единицы работы через промежуточные группы для выявления критических по производительности мест.

- Усовершенствования в команде DB2 ping

Команда DB2 ping теперь поддерживает задание размера пакета требования и ответа.

Примечание: DB2 Connect не поддерживает команду PING, если её выдаёт хосту клиент версии 7 через шлюз версии 9.

Функции, введенные в DB2 Connect, Версия 7 Выпуск 2

DB2 Connect версии 7.2 содержит следующие усовершенствования:

- Улучшенная поддержка технологий Microsoft Transaction Server (MTS) и COM+
- DB2 Connect Web Starter Kit
- DB2 Connect для Linux в S/390

Функции, введенные в DB2 Connect, Версия 7 Выпуск 1

DB2 Connect версии 7.1 содержит следующие усовершенствования:

- Концентратор ХА
- Усовершенствования в функции многоузлового обновления

Базы данных хоста

Термин *база данных* используется в этом документе 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:

OS/390 или z/OS

DB2 Universal Database (UDB) for OS/390 and z/OS Версии 7 или DB2 UDB for z/OS Версии 8. Подсистема DB2 Universal Database for z/OS and OS/390 идентифицируется значением 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. LOCATION NAME поддерживает до 8 алиасов имен положений, что позволяет прикладным программам использовать разные имена dbalias (алиаса базы данных) для доступа к серверу z/OS Версии 8. Получить имя положения сервера DB2, имя домена, IP-адрес и порт можно при помощи команды z/OS -display ddf.

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

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

OS/400

DB2 for i5/OS, входящая в операционную систему OS/400. На сервере System i может существовать только одна база данных, если только система не сконфигурирована на использование дополнительных пулов хранения.

DB2 Connect и операторы SQL

DB2 Connect передает операторы SQL от прикладных программ на серверы баз данных хоста или System i.

DB2 Connect может передать практически любой допустимый оператор SQL, а также поддерживаемые API DB2:

- JDBC
- SQLJ
- ADO.NET
- OLE DB
- ODBC
- Perl
- PHP
- DB2 CLI
- Встроенный SQL

Поддержка встроенного SQL

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 System i. Динамический SQL обеспечивает большую гибкость, однако производительность обычно снижается. Программист прикладных программ должен выбрать, какой именно тип операторов SQL использовать - статический или динамический. Оба эти типа поддерживаются DB2 Connect.

Различные серверы баз данных хоста или System i используют разные реализации SQL. DB2 Connect полностью поддерживает общие операторы SQL IBM, а также реализации SQL в DB2 для OS/390 и z/OS, DB2 Server for VSE & VM (прежнее название - SQL/DS) и DB2 для System. Для обеспечения независимости базы данных настоятельно рекомендуется использовать IBM SQL.

Утилиты управления DB2 Connect

Системному администратору DB2 Connect могут помочь следующие утилиты:

- Процессор командной строки (CLP) позволяет вводить операторы SQL для базы данных сервера баз данных хоста или System i. Он передает эти операторы SQL указанной базе данных.
- Командный центр DB2 поддерживает графический интерфейс для Процессор командной строки (CLP).
- 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 System i. Эти файлы можно затем использовать для импорта данных в базы данных, электронные таблицы и другие прикладные программы на рабочей станции.
- Если у вас запущен серверный продукт DB2 Connect, можно 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 для OS/390 или z/OS, такими как таблицы, производные таблицы, пулы буферов и потоки.
- Утилита системный монитор баз данных позволяет системному администратору следить за соединениями системы. Однако это возможно только в том случае, если DB2 Connect играет роль сервера. Данная утилита 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 System i.

Примечание: В прежних выпусках инструменты управления с графическим интерфейсом DB2, например, Центр управления, были доступны на всех платформах. Начиная с Версии 9, инструменты управления с графическим интерфейсом DB2 будут доступны только для Windows x86, Windows x64 (AMD64/EM64T), Linux на x86 и Linux на AMD64/EM64T. На всех платформах для управления можно пользоваться процессором командной строки DB2 (CLP).

WebSphere Federation Server и DB2 Connect

WebSphere Federation Server - это отдельный продукт, обеспечивающий доступ к данным в источниках данных различных поставщиков и интеграцию этих данных, тогда как DB2 Connect позволяет использовать большие объемы данных, хранящихся на существующих больших компьютерах.

WebSphere Federation Server помогает интегрировать информацию, позволяя просматривать собрания источников данных и работать с ними, как с одним источником. Этот продукт делает доступ к источникам данных полностью прозрачным для вызывающей их программы. WebSphere Federation Server работает в сочетании с продуктами сервера DB2 Connect. WebSphere Federation Server поддерживает возможности чтения и записи для баз данных семейства продуктов DB2, Informix, Oracle, Sybase, Teradata и Microsoft SQL Server. WebSphere Federation Server поддерживает также доступ для чтения к нереляционным источникам данных, таким как BLAST, Documentum, Entrez, IBM Lotus Extended Search, файлам с табличной структурой и файлам XML. При помощи этого продукта можно формулировать запросы к данным в системе объединения.

Глава 2. Архитектура распределенных реляционных баз данных (DRDA)

Архитектура распределенных реляционных баз данных (Distributed Relational Database Architecture - DRDA) - 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.

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.

Распределенная единица работы

В *распределенной единице работы (DUOW)*, называемой также многоузловым изменением, участвуют несколько серверов баз данных. DUOW обладает следующими свойствами:

- За единицу работы изменения выполняются на нескольких серверах баз данных.
- 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.

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 говоря, 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 операционных систем Windows, UNIX и Linux. Эти средства входят в клиент DB2. Менеджер баз данных DB2 поддерживает ряд интерфейсов программирования: ADO.NET, JDBC, SQLJ, PHP, Perl DBI, встроенный SQL, интерфейс уровня вызовов DB2 (интерфейс уровня вызовов DB2) и OLE DB. Эти API могут использовать программисты, пишущие программы на многих языках программирования.

DB2 Connect и DRDA

DB2 Connect применяет архитектуру DRDA, чтобы удешевить и упростить доступ к данным, хранящимся в DB2 Universal Database (UDB) for iSeries, DB2 for System i, DB2 UDB for OS/390 and z/OS, DB2 for z/OS, DB2 Server for VSE & VM и на 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 ту сторону соединения, где находятся прикладные программы. Реквестер - программа, запрашивающая данные. DB2 Connect действует как реквестер прикладных программ от имени прикладных программ, которые могут быть локальными по отношению к рабочей станции DB2 Connect, или находиться на другом клиенте, отдельном, от DB2 Connect.

сервер прикладных программ - это код, обслуживающий ту сторону, где расположена база данных.

DRDA поддерживает также многоуровневые соединения между реквестером прикладных программ и сервером. В этой топологии сервер, с которым соединяется реквестер прикладных программ - это сервер прикладных программ; любой другой сервер, следующий за ним по направлению трафика, называется сервером баз данных (database server, DS), поскольку он не взаимодействует напрямую с реквестером прикладных программ. Кроме того, чтобы подчеркнуть, что серверы прикладных программ и серверы баз данных не являются ни системами, откуда исходят требования для баз данных, ни системами, выполняющими затребованные функции баз данных, каждый из этих серверов между реквестером прикладных программ и конечным сервером баз данных называют промежуточным сервером. В DB2 Connect поддерживается использование серверов баз данных и промежуточных серверов.

На рис. 1 показан поток данных между рабочей станцией DB2 Connect и сервером хоста или System i в случае, когда есть только локальные клиенты.

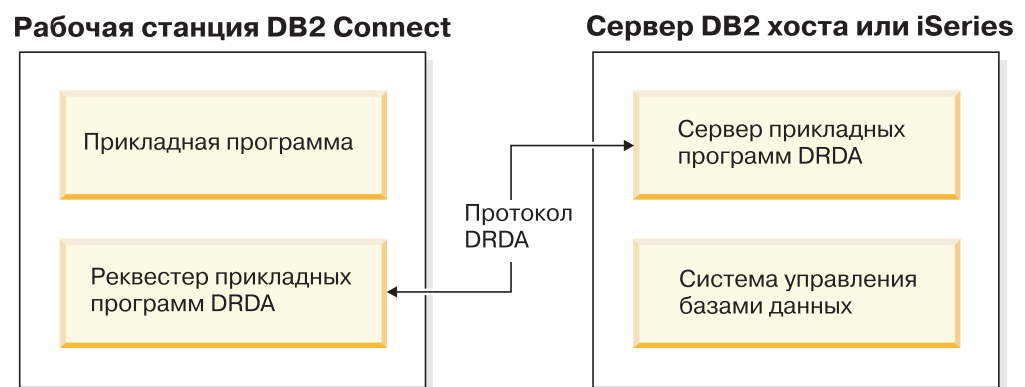


Рисунок 1. Поток данных между сервером DB2 Connect и сервером хоста или System i

Для поддержки соединений между системами управления базами данных на сервере DRDA и клиентами IBM Data Server в DRDA используются 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)
- Протокол TCP/IP (Transmission Control Protocol/Internet Protocol - Протокол управления передачей/Протокол Internet).

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 стоимости транзакции. 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).

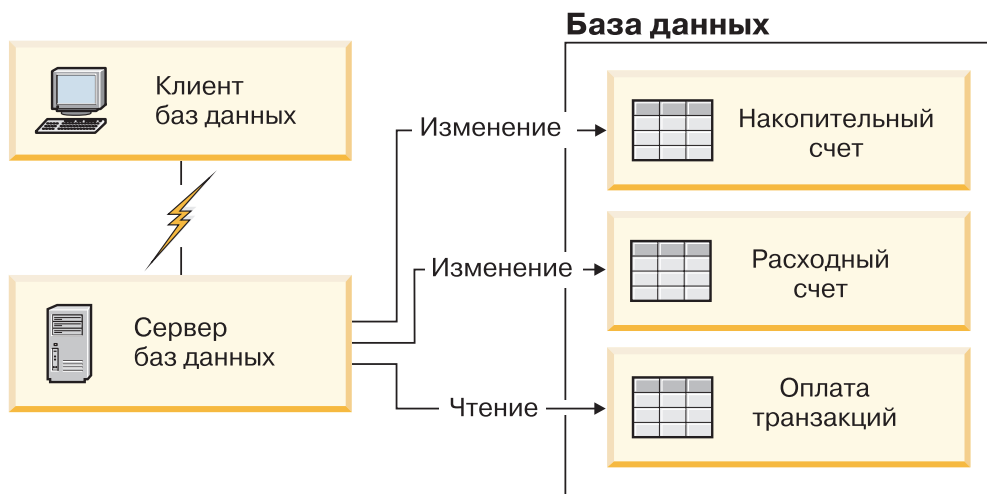


Рисунок 2. Использование в транзакции одной базы данных

Для конфигурирования такой программы надо:

1. Создать таблицы для накопительного счета, чекового счета и плана оплаты стоимости транзакции в одной базе данных.
2. Для физически удаленной базы данных: задать для сервера баз данных использование подходящего протокола связи.
3. Для физически удаленной базы данных - внести в каталог узел и базу данных, чтобы определить эту базу для сервера.
4. Прекомпилировать прикладную программу, задав соединение типа 1; для этого надо указать `CONNECT(1)` в команде `PREP`.

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 для OS/390 или z/OS.

DB2 Connect поддерживает распределенные требования для баз данных и СУБД. Например, можно выполнить операцию UNION между таблицей DB2 и представлением Oracle. Поддерживаются СУБД семейства DB2 (DB2 Database для Linux, UNIX и Windows, DB2 для OS/390, z/OS и DB2 для System i) и Oracle. При использовании DB2 Connect вместе с WebSphere Federation Server доступна поддержка нескольких поставщиков.

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 монополизирует 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, 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* понимаются 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.

Глава 3. Сценарии DB2 Connect

DB2 Connect позволяет реализовать множество вариантов доступа к базам данных хоста или System i. В этом разделе описаны основные сценарии, которые можно применять в зависимости от конкретных задач и используемых систем.

Прямой доступ к базам данных хоста

Основная возможность DB2 Connect - обеспечение прямого соединения с базой данных хоста из прикладных программ, выполняющихся на рабочих станциях Windows или Linux. Простейший способ реализовать такое соединение - использовать DB2 Connect Personal Edition.

Каждая рабочая станция, на которой установлен DB2 Connect Personal Edition, может устанавливать прямое соединение TCP/IP с серверами DB2 Universal Database (UDB) for OS/390 and z/OS, DB2 for z/OS, DB2 UDB for iSeries, DB2 for i5/OS и DB2 Database для Linux, UNIX и Windows. Кроме этого, прикладные программы могут в одной транзакции соединяться с несколькими базами данных семейства DB2 и изменять в них данные в одной транзакции; при этом протокол двухфазного принятия обеспечивает целостность всех данных.

На рис. 3 на стр. 16 показано прямое соединение с сервером баз данных хоста или System i с рабочей станции, на которой установлена DB2 Connect Personal Edition.

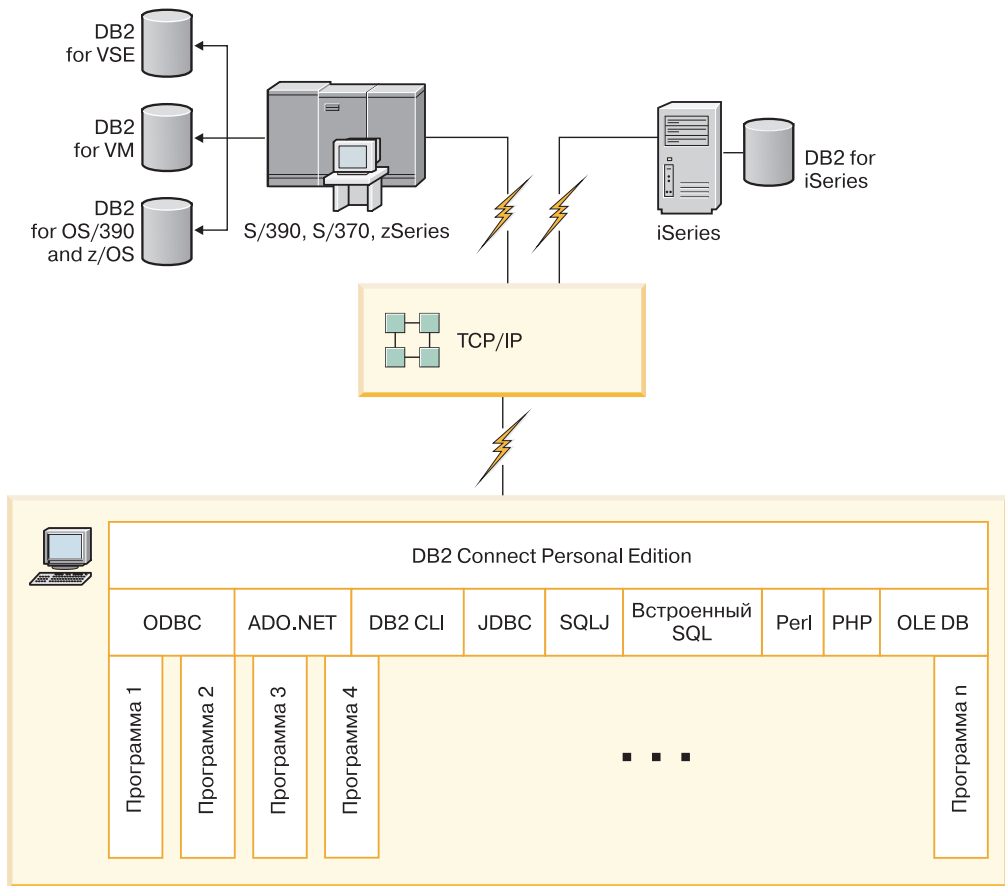


Рисунок 3. Прямое соединение между DB2 Connect и сервером баз данных хоста или System i

Примечание:

1. На рабочей станции DB2 Connect установка DB2 не обязательна. Но, если на рабочей станции DB2 Connect нужна полная система управления реляционными базами данных, закажите DB2.
2. В пакет DB2 Connect теперь входит IBM Data Server, 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 и z/OS.
3. 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.
4. Если соединение с сервером баз данных DB2 for z/OS при разрешенном использовании Sysplex разорвано, клиент автоматически попытается восстановить его.

Доступ к данным DB2 хоста или System i при помощи DB2 Connect Personal Edition

Прямое соединение без промежуточных серверов удобно и весьма желательно. Это особенно важно в ситуациях, когда сервер баз данных хоста или System i поддерживает соединения TCP/IP. В таких ситуациях каждая рабочая станция DB2 Connect устанавливает прямое соединение с сервером баз данных хоста или System i.

Для применения соединений TCP/IP необходимо, чтобы база данных хоста или System i поддерживала TCP/IP. Следующие версии содержат встроенную поддержку соединений TCP/IP:

- DB2 Universal Database (UDB) for OS/390 and z/OS Версии 7.1 или новее
- DB2 UDB for iSeries Версия 5 Выпуск 1 и новее; и
- DB2 Server for VSE & VM Версия 7 и новее

Чтобы соединиться с сервером баз данных хоста или System i IBM, требуется лицензированный продукт DB2 Connect. Нельзя соединяться напрямую с сервером хоста IBM или сервером данных System i, используя IBM data server.

На рис. 4 на стр. 18 показана рабочая станция с установленной DB2 Connect Personal Edition, напрямую соединенная с сервером баз данных хоста или System i.

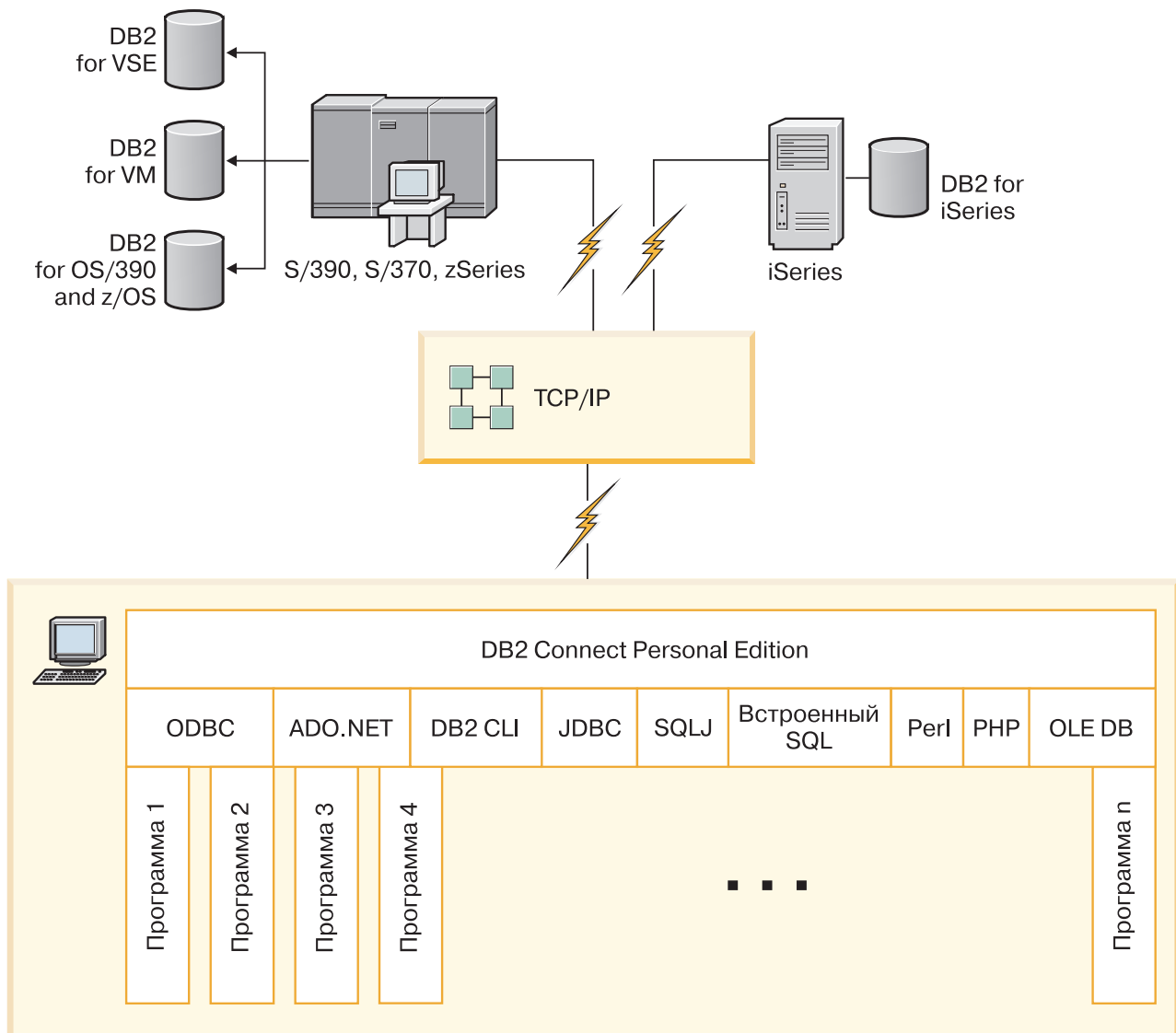


Рисунок 4. Прямое соединение между DB2 Connect и сервером баз данных хоста или System i

Продукты сервера DB2 Connect в качестве серверов соединений

Сервер DB2 Connect позволяет нескольким клиентам соединяться с базами данных хоста или System i и может значительно уменьшить затраты на организацию доступа к данным предприятия. На рис. 5 на стр. 19 показано предлагаемое IBM решение для случаев, когда вы хотите, чтобы клиент DB2 устанавливал не прямое соединение с сервером баз данных хоста или System i через продукт сервера DB2 Connect, такой как DB2 Connect Enterprise Server Edition.

Примечание: Непрямые соединения поддерживаются только с клиентами DB2 или клиентами JCC, работающими в Linux, UNIX или Windows. Попытка соединиться с сервером базы данных хоста или System i через продукт сервера DB2 Connect с использованием иного клиента приведет к ошибке SQL1334.

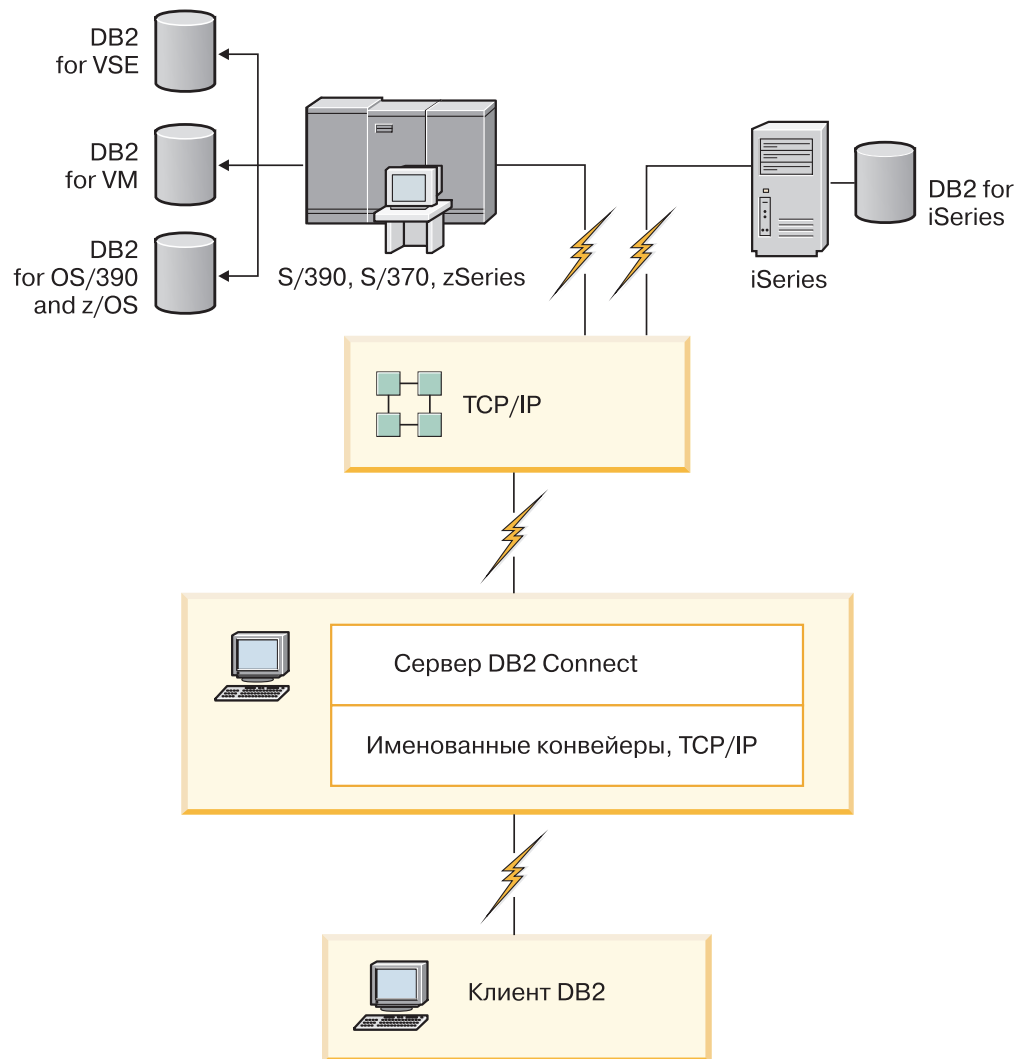


Рисунок 5. DB2 Connect Enterprise Server Edition

Если соединение TCP/IP с сервером DB2 Connect разорвано, клиент автоматически попытается восстановить его. Сначала клиент попытается восстановить соединение с исходным сервером. Если установить соединение не удастся, клиент перейдет к альтернативному серверу DB2 Connect. (Альтернативный сервер задается на экземпляре сервера, и его положение сообщается клиенту во время установки соединения.) Если соединение с альтернативным сервером не устанавливается, клиент попытается повторно установить соединение с исходным сервером. Клиент будет поочередно пытаться восстановить соединение с исходным и альтернативным сервером вплоть до установления соединения или исчерпания допустимого числа попыток.

DB2 Connect и программы Web

Браузер Web постепенно 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.

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, где 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 или PHP).

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, проблемой может стать масштабируемость для широкомасштабной электронной коммерции. При каждом запуске программы CGI на сервере Web создается новый процесс. Каждый процесс должен создать свое собственное соединение с базой данных и передать свой собственный запрос. 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 позволяют пользователям DB2 Connect уменьшить число требований, посылаемых базе данных.

Объединение соединений в пулы сокращает количество операций установления и завершения соединений с базой данных.

Использование PHP как модуля Web-сервера или встраиваемого модуля

Хотя PHP можно использовать для программирования CGI, обычно его используют как модуль Web-сервера или встраиваемый модуль. В многопроцессорном Web-сервере, например, Apache, для снятия проблемы масштабируемости можно использовать драйвер IBM DB2 для PHP. В многопроцессорном Web-сервере пул процессов многократно используется для обслуживания требований Web-сервера. Чтобы устранить необходимость в построении соединения с базой данных для каждого Web-требования, можно создать постоянное соединение. В такой среде постоянное соединение может существовать вне области действия отдельного сценария PHP. Соединение будет повторно использоваться, если такое же соединение потребуется в дальнейшем для другого Web-требования.

DB2 Connect и IBM WebSphere

IBM WebSphere позволяет реализовать более полные решения для электронного бизнеса по сравнению с использованием традиционных сценарных средств, таких как PHP. WebSphere Application Servers поддерживают не только сценарии PHP, но позволяют также реализовать сложные системы для работы через Web, в которых используются сервлеты, Active Server Pages и Enterprise JavaBeans, Java, TCP/IP, HTTP, HTTPS, HTML, DHTML, XML, MIME, SMTP, POP и X.509, а также другие возможности. 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 (AIX, HP-UX, Linux, Novell NetWare, OS/390, z/OS, OS/400, операционной системе Solaris, Microsoft Windows)
- 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 Application Server.

WebSphere Application Server 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.

Прикладные программы WebSphere могут размещаться на той же платформе, что и сервер Web и DB2. При использовании DB2 Universal Database (UDB) for OS/390 and z/OS, DB2 for z/OS, DB2 for VM, DB2 for VSE, DB2 UDB for iSeries и DB2 for i5/OS, WebSphere размещается на той же платформе, что и продукт сервера DB2 Connect.

Есть несколько вариантов WebSphere, а также среда разработки Rational Application Developer (RAD). Подробности смотрите по адресу <http://www.ibm.com/software/webservers/appserv/was/>

DB2 Connect в качестве сервера прикладных программ Java

Можно избежать многих недостатков языков сценариев, используя вместо них язык Java. IBM предоставляет апплеты и прикладные программы Java, позволяющие использовать Java на каждом шаге транзакции Web. Предлагаемые IBM решения позволяют использовать различные техники - для работы с DB2 можно использовать сценарии (например, Perl DBI или Microsoft Active Server Pages с DB2) 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 использоваться 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 for OS/390 and z/OS, DB2 for VSE and VM, DB2 Universal Database (UDB) for iSeries, and DB2 for i5/OS есть два способа внедрения прикладных программ Java. Можно использовать прямое соединение через DB2 Connect Personal Edition с TCP/IP или соединение через продукт сервера DB2 Connect, обеспечивающий связь с сервером данных хоста или System i.

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

Главный компонент для DB2 for OS/390 and z/OS - это продукт сервера DB2 Connect, работающий на сервере среднего уровня. Именно этот компонент обеспечивает возможности использования сервера JDBC, а также связь с сервером DB2 for OS/390 and z/OS, DB2 for VSE and VM, DB2 Universal Database (UDB) for iSeries и DB2 for i5/OS. Еще раз подчеркнем, что для браузера Web клиента не требуется никакого специально программного обеспечения.

IBM предоставляет всестороннюю поддержку и большой набор инструментов для разработки прикладных программ и апплетов Java. Для разработки прикладных программ баз данных можно использовать DB2 Database Enterprise Developer Edition, содержащий Rational Web Developer, IBM Data Server Developer Workbench, встроенный сервер прикладных программ DB2 и Cloudscape Версии 10.2; кроме того, в нем есть DB2 Connect для тестирования программ. Для работы с базами данных IBM можно также использовать продукты других производителей (например, NetBeans, Borland JBuilder или Symantec Visual Cafe).

DB2 Connect на Web-сервере

IBM предоставляет серверы HTTP (Web) со всеми продуктами DB2 Connect. Продукты серверов DB2 Connect, например, DB2 Connect Enterprise Server Edition, обеспечивают дополнительную поддержку для серверов Apache или Lotus Domino Go и могут 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 zSeries, System i, VM и VSE, на сервере Web должен быть установлен продукт сервера DB2 Connect. Продукты сервера DB2 Connect обеспечивают библиотеки и интерфейсы связи, позволяющие серверам Web обращаться к платформам хоста и System i. Для связи между сервером Web и базой данных, работающей в системе zSeries, System i, VM или VSE, можно использовать TCP/IP.

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

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 Database для Linux, UNIX и Windows и DB2 Universal Database (UDB) on OS/390 and z/OS, DB2 for z/OS, DB2 UDB for iSeries, DB2 for i5/OS и DB2 for VSE используется одно и то же соглашение о передаче параметров при вызовах хранимых процедур.

Как и в случае обычных сценариев с Web-интерфейсом, браузер посылает форму на сервер Web, на котором выполняется сценарий с Web-интерфейсом. Однако базе данных DB2 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 с Web-интерфейсом и базой данных.

Главное преимущество использования хранимых процедур - уменьшение потока данных между сервером HTTP и базой данных DB2.

DB2 Connect и серверы прикладных программ

С появлением прикладных программ типа клиент-сервер разработчики смогли создавать программы, более удобные в использовании и требующие меньших затрат на обучение персонала. Эти программы применяют графический интерфейс пользователя и работают на таких платформах, как Windows. 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. DB2 Connect обеспечивает полную поддержку двухуровневых прикладных программ клиент-сервер при использовании в качестве серверов баз данных DB2 Universal Database (UDB) for OS/390 and z/OS, DB2 for z/OS, DB2 UDB for iSeries, DB2 for i5/OS или DB2 for VM and 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.

Реализация сервера 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

Серверные продукты DB2 Connect обеспечивают полную поддержку многоуровневых прикладных программ. DB2 Connect обеспечивает поддержку множества API, которые могут использоваться при разработке прикладных программ (ODBC, ADO.NET, DB2 CLI, Встроенный SQL, JDBC, SQLJ, Perl, PHP и OLE DB), 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 и в базе данных DB2 для OS/390 и z/OS, и в DB2 Database для Linux, UNIX и Windows. 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 серверным продуктом DB2 Connect, таким как DB2 Connect Enterprise Server Edition.

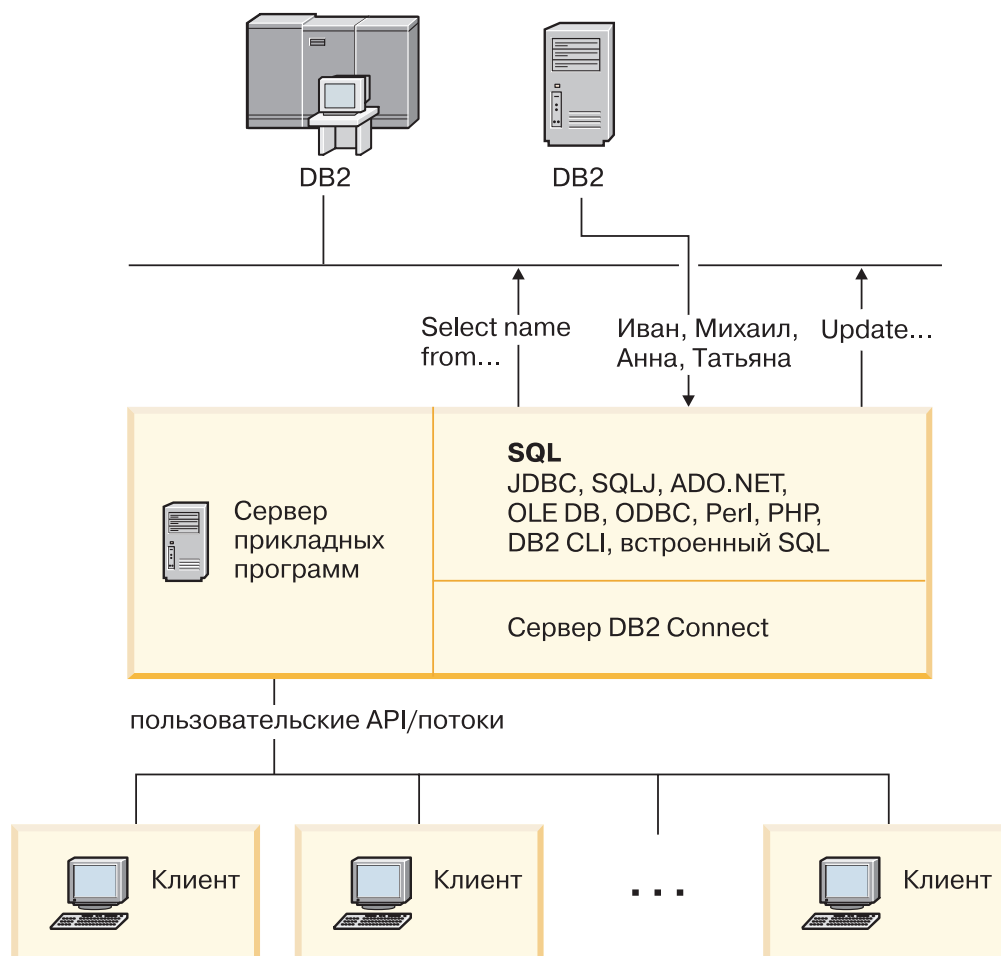


Рисунок 6. Поддержка DB2 Connect для серверов прикладных программ

Дополнительные возможности DB2 Connect, такие как динамические соединения, 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 серверный продукт DB2 Connect. 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. Серверный продукт DB2 Connect может быть установлен:

- либо на компьютере сервера прикладных программ,
- либо 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 серверный продукт DB2 Connect недоступен для операционной системы или аппаратной платформы, где работает сервер прикладных программ, DB2 Connect можно установить на отдельном сервере.

DB2 Connect и мониторы транзакций

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.

Обработка 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 Linux, UNIX и Windows выполнять транзакции с удаленными серверами баз данных в локальной сети, на хосте или System i и координировать транзакции при помощи монитора транзакций.

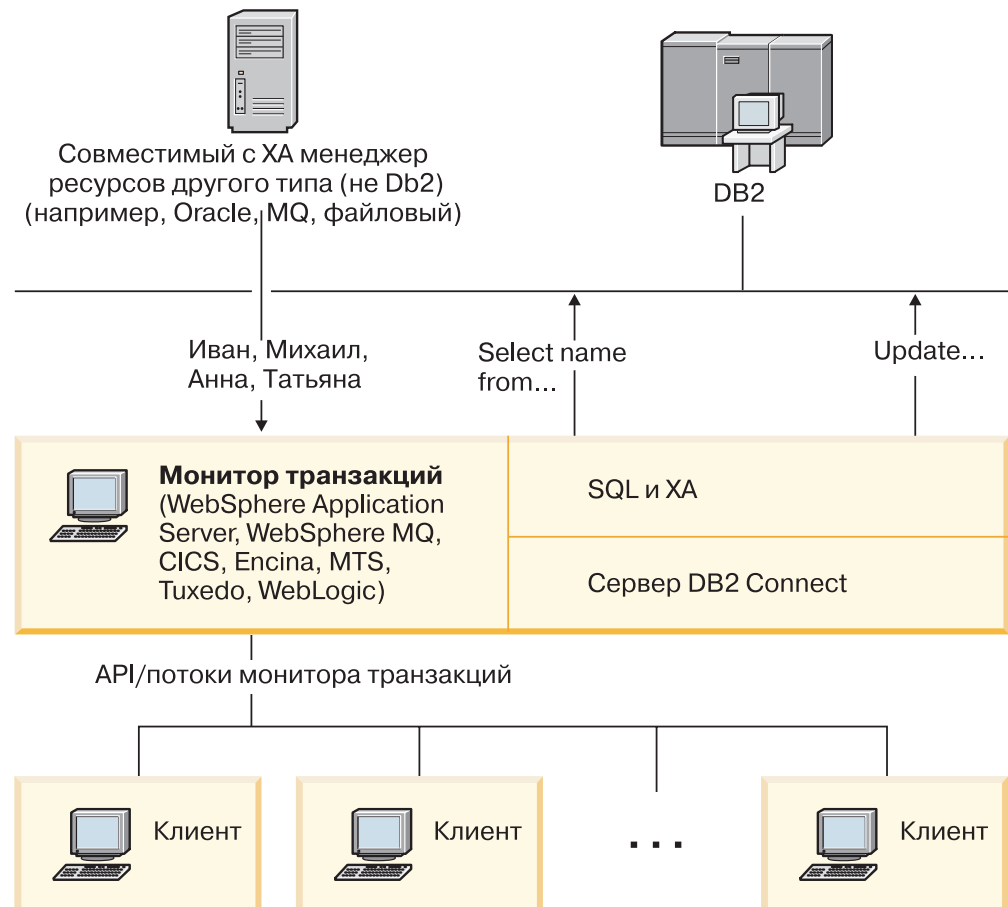


Рисунок 7. Поддержка DB2 Connect для мониторов транзакций

На рис. 7 показаны API и механизм связи между сервером прикладных программ и серверами баз данных, обеспечиваемые продуктом сервера DB2 Connect, например DB2 Connect Enterprise Server Edition.

Примеры мониторов транзакций

Наиболее распространенные в настоящее время мониторы транзакций:

- IBM WebSphere Application Server
- IBM WebSphere MQ
- IBM TxSeries CICS
- IBM TxSeries Encina Monitor
- BEA Tuxedo
- BEA WebLogic
- Microsoft Transaction Server (MTS)

В транзакциях, координируемых этими мониторами, могут использоваться серверы баз данных удаленной System i, zSeries и локальной сети.

Модель распределенной обработки транзакций (DTP) X/Open

Реализующая коммерческий алгоритм прикладная программа может внутри одной транзакции потребовать 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 от разных производителей. Например, одна база данных - это DB2 Universal Database for OS/390 and z/OS, 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. Базы данных удаленного хоста, System i и DB2 в локальной сети при доступе к ним через DB2 Connect представляют собой менеджеры ресурсов стандарта XA. Поэтому любой монитор транзакций, совместимый со стандартом XA, может при выполнении транзакций коммерческих прикладных программ работать с базами данных хоста, System i и DB2 в локальной сети.

Часть 2. Справочник по DB2 Connect

Глава 4. Изменение каталогов баз данных

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

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

Примечание:

1. Перед внесением изменений в эти каталоги надо сконфигурировать связь на сервере баз данных хоста или System i и на рабочих станциях.
2. Каталоги баз данных можно изменить с помощью Ассистента конфигурирования (CA).

Для изменения каталогов баз данных выполните следующие действия:

1. Соберите информацию о каталогах баз данных с помощью рабочего листа настройки каталогов
2. Смотрите раздел “Добавление в каталоги информации об удаленных компьютерах серверов баз данных” в Центре управления

Значения системного каталога 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 System i. 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 (_). Имя не может начинаться с символа подчеркивания или цифры.

Имя узла

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

Аутентификация

Этот параметр задает, где будет проверяться имя пользователя и пароль при подключении с сервера DB2 Connect. Допустимы следующие опции: SERVER, SERVER_ENCRYPT, CLIENT, KERBEROS и DATA_ENCRYPT. Тип аутентификации GSSPLUGIN не поддерживается в системном каталоге базы данных.

Значения каталога узла

В каталоге узла можно задать следующую информацию:

Имя узла

Псевдоним для системы сервера баз данных хоста или System i, 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 (_). Имя не может начинаться с символа подчеркивания или цифры.

Протокол

Только TCP/IP.

Тип защиты

Тип защитной проверки, который будет использоваться. 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.

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 и z/OS имя хоста выводится в сообщении DSNL004I (DOMAIN=имя_хоста) при запуске DDF. Можно использовать также команду -DISplay DDF.

При обращении к группе совместного использования данных z/OS имя домена должно отображаться на динамический адрес VIPA группы DB2. Этот адрес указывает на наименее загруженный элемент DB2. Для доступа к конкретному элементу DB2 используйте его динамический адрес VIPA, отключив маршрутизацию sysplex. Сообщение DSNL004I для каждого элемента содержит его доменное имя.

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 и z/OS номер порта задается в наборе данных начальной загрузки (BSDS) как PORT и также появляется в сообщении DSNL004I (TCPPORT=номер_порта) при запуске DDF. Можно использовать также команду -DISplay DDF.

При обращении к группе совместного использования данных z/OS имя домена должно отображаться на динамический адрес VIPA группы DB2. Этот адрес указывает на наименее загруженный элемент DB2. Для доступа к конкретному элементу DB2 используйте его динамический адрес VIPA, отключив маршрутизацию sysplex. Сообщение DSNL004I для каждого элемента содержит его доменное имя.

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

Значения каталога DCS

В каталоге DCS можно задать следующую информацию:

Имя базы данных

Задаваемый пользователем псевдоним для сервера баз данных хоста или System i. 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 (_). Имя не может начинаться с символа подчеркивания или цифры.

Имя базы данных потребителя

База данных в системе сервера баз данных хоста или System i:

OS/390 и z/OS

Подсистема DB2 Universal Database for z/OS and OS/390, определяемая своим именем LOCATION NAME или одним из алиасов LOCATION, заданных на сервере z/OS.

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

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

Несколько LOCATION NAME указываются также в наборе данных Boot Strap Data Set (BSDS) и в сообщении DSNL004I (LOCATION=имя_положения), появляющемся при запуске DDF. Можно использовать также команду -DISplay DDF.

При обращении к группе совместного использования данных z/OS имя домена должно отображаться на динамический адрес VIPA группы DB2. Этот адрес указывает на наименее загруженный элемент DB2. Для доступа к конкретному элементу DB2 используйте его динамический адрес VIPA, отключив маршрутизацию sysplex. Сообщение DSNL004I для каждого элемента содержит его доменное имя.

VSE или VM

Имя базы данных (DBNAME)

OS/400 и z/OS

Имя реляционной базы данных (RDBNAME)

Другие Для операционных систем Windows, Linux и UNIX - алиас базы данных в каталоге баз данных.

Строка параметров

При желании изменить значения по умолчанию задайте произвольное сочетание приведенных ниже параметров в заданном порядке.

файл отображения

Имя файла отображения SQLCODE для замены отображения по умолчанию SQLCODE. Чтобы отключить отображение SQLCODE, задайте **NOMAP**.

Примечание: При отработке запроса сервер DRDA возвращает данные в виде набора строк, представляющих набор результатов. Вместе с каждой строкой возвращается

SQLCA, обычно содержащий число ноль или положительный sqlcode (например, +12 или +802). Если на сервере DB2 Connect применяется настраиваемый файл отображения, положительные значения sqlcode не будут отображены, если в этом файле для них есть отображение (то есть они сопоставлены другим кодам sqlcode или маркерам).

Обязательно учтите следующее:

1. Положительные значения соответствуют предупреждениям, а отрицательные - ошибкам. Отрицательные значения будут отображены в любом случае, вне зависимости от применяемого файла отображений. Все положительные значения sqlcode, содержащиеся в файле отображений отображение в самого себя, также будут отображены. Кроме того, всегда будут отображены положительные значения sqlcode, не содержащиеся в файле отображений сервера DB2 Connect.
2. Если используется файл отображений по умолчанию, или если вы установили прямое соединение с базой данных хоста, то отображение sqlcode также будет выполнено для всех значений sqlcode.

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

SQL3000N
SQL30040N
SQL30050N
SQL30051N
SQL30053N
SQL30060N
SQL30070N
SQL30071N
SQL30072N
SQL30073N
SQL30074N
SQL30090N

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

SQL30020N
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. INTERRUPT_ENABLED применимо только в случае, если конечный сервер не поддерживает прерывания. Если сервер поддерживает поток прерываний DRDA, то DB2 Connect просто передаст запрос на прерывание серверу.

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 System i, DB2 Connect обработает это прерывание, разорвав соединение и выполнив откат единицы работы. Такая обработка прерывания поддерживается в AIX и Windows.

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 System i новое соединение. На всех платформах, кроме AIX Версии 5.2 или новее и Windows, 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 z/OS and 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.

,,,,,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 TCP/IP node nynode remote myhost server myport
catalog dcs database nydb1 as new_york
catalog database nydb1 as newyork1 at node nynode
authentication server
```

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 кавычках, 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.

```

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

```

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:

```

"YyуuMmDd" - (Y, M и D в любом регистре)
"MM+DD+YYYY" - Маска может быть длиннее 10 байт
                 и содержать символы, отличные
                 от Y, M и D
"abcYY+MM" - Посл. D может отсутствовать.

```

Следующие маски дат недопустимы:

```

"YYYYуMMDD" - недопустима, в посл. 5 Y
"YYYYMDDM" - недопустима, есть 2 посл. 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.

,,,,,,**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 CCSID с двумя направлениями письма базы данных сервера.
Например:

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

где xyz - новый CCSID.

Рабочий лист настройки каталогов

В рабочем листе настройки каталогов перечислена информация, которую необходимо собрать. Полезно сделать копию этого рабочего листа и вписать в нее данные своей системы.

Параметры каталога узла

Таблица 1. Параметры каталога узла

Параметр	Пример	Ваше значение
Имя узла	DB2NODE	
Имя удаленного хоста (узел TCP/IP)	ZOSHOST	
Сервер (имя службы или номер порта TCP/IP)	db2inst1c (или 446)	

Примечание:

1. Номер порта TCP/IP по умолчанию для DRDA - 446
2. Если вы не уверены, что сервер баз данных хоста или System i поддерживает SECURITY SOCKS, не задавайте защиту для узла TCP/IP.

Параметры каталога DCS

Таблица 2. Параметры каталога DCS

Параметр	Пример	Ваше значение
Имя базы данных	DB2DB	
Имя базы данных потребителя	NEW_YORK3	
Реквестер прикладных программ		
Строка параметров	" ,,,,,,LOCALDATE=\\\"ГГММДД\\\""	

Параметры системного каталога баз данных

Таблица 3. Параметры системного каталога баз данных

Параметр	Пример	Ваше значение
Имя базы данных	DB2DB	
Алиас базы данных	NYC3	
Имя узла	DB2NODE	
Аутентификация	SERVER	

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 System i, 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.

Обработка данных с двумя направлениями письма

Следующий раздел относится только к серверам OS/390 и z/OS. Эту возможность не нужно включать для сервера DB2 for i5/OS, поскольку на нем уже обеспечена полная поддержка двух направлений письма.

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 в OS/390 или z/OS обычно имеют формат SHAPED VISUAL. Следовательно, без поддержки атрибутов BiDi данные, переданные от DB2 for OS/390 или z/OS 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. Это достигается с помощью параметра `VIDI` в поле `PARMS` записи каталога базы данных `DCS` для базы данных сервера.

Использование этой функции лучше всего показать на примере.

Рассмотрим `IBM data server` 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, содержащихся в базе данных хоста `DB2`, используется `CCSID` 62245 (строки с двумя направлениями письма типа 10).

В этой ситуации существует две проблемы. Во-первых, базе данных `DB2` хоста не известно различие между типами строк `CCSID` 424 и 62245. Во-вторых, база данных `DB2` хоста не распознает `CCSID` 62213 для клиента `IBM Data Server`. Она поддерживает только `CCSID` 62209 (тип строки с двумя направлениями 10), основанный на той же кодовой странице, что и `CCSID` 62213.

Сначала нужно убедиться, что данные, передаваемые на базу данных `DB2` хоста, 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 `DB2` хоста. Выполните для базы данных хоста `DB2` следующую команду занесения в каталог:

```
catalog dcs database nydb1 as TELAVIV parms ",,,,,,,VIDI=62245"
```

Для `DB2 Connect` это указание заменить `CCSID` 424 базы данных хоста `DB2` на `CCSID` 62245. Эта замена заключается в следующем:

1. `DB2 Connect` соединится с базой данных хоста `DB2`, используя `CCSID` 62209 (тип строки с двумя направлениями письма 10).
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` 62209 (строки с двумя направлениями типа 10).
3. `DB2 Connect` будет выполнять преобразование строк с двумя направлениями письма для данных, получаемых от базы данных хоста `DB2`, заменяя `CCSID` 62245 (строки с двумя направлениями письма типа 10) на `CCSID` 62213 (строки с двумя направлениями письма типа 5).

Примечание:

1. Чтобы параметр `VIDI` вступил в силу, для переменной среды или реестра `DB2VIDI` нужно установить значение `YES`.
2. Если от `DB2 Connect` требуется выполнять преобразование размещения для данных перед их передачей в базу данных хоста `DB2`, даже без замены их `CCSID`, в поле `PARMS` каталога базы данных `DCS` все равно нужно добавить параметр `VIDI`. В этом случае требуемое значение `CCSID` будет значением по умолчанию `CCSID` базы данных хоста `DB2`.
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.

Спецификации строк параметров

Ниже приведены примеры параметров DCS (каждая строка - набор параметров):

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

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

Примечание: При работе с процессором командной строки из командной строки операционной системы в системах UNIX следует использовать эскейп-символ операционной системы "\" (обратная косая черта) в связи с тем, что при задании маски LOCALDATE в строке параметров требуются две пары двойных кавычек. 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
```

Глава 5. Защита DB2 Connect

При использовании DB2 Connect важна аутентификация пользователей, поскольку пользователи DB2 Connect и базы данных, к данным которой они обращаются, могут быть как локальными, так и удаленными. Наряду с особенностями защиты для баз данных на компьютерах хоста представлены доверенные соединения и поддержка Kerberos.

Доверенные соединения при помощи DB2 Connect

Некоторые серверы баз данных DB2 поддерживают доверенный контекст. *Доверенный контекст* позволяет администратору баз данных, наряду с прочим, определить условия, при которых клиентской программе разрешается создавать доверенное соединение. При *доверенном соединении* допустимы действия, запрещенные при обычном соединении.

Есть два типа доверенного соединения - явное и неявное. При создании соединения его тип (явное доверенное соединение, неявное доверенное соединение, обычное соединение) зависит от того, запросили ли вы доверенное соединение и отвечает ли это соединение критериям, определенным в доверенном контексте на сервере, как показано в Табл. 4.

Таблица 4. Какой тип соединения получается при данном сочетании действий

	Соединение отвечает критериям доверенного соединения на сервере	Соединение не отвечает критериям доверенного соединения на сервере
Вы запросили доверенное соединение	Явное доверенное соединение	Обычное соединение и сообщение SQL20360W (SQLSTATE 01679).
Вы не запросили доверенное соединение	Неявное доверенное соединение	Обычное соединение

Неявное доверенное соединение подобно обычному соединению, но оно дает временные привилегии роли пользователю, который использует это соединение. Предоставляемые привилегии роли (если они есть) указываются в доверенном контексте, в котором создается доверенное соединение.

Неявные доверенные соединения может создавать любая программа, которая использует при соединении DB2 Connect. Неявные доверенные соединения создаются и используются точно, как обычные соединения. Это означает, что не требуется вносить изменения в код, чтобы существующие программы могли воспользоваться преимуществами неявных доверенных соединений, если эти программы выполняют соединения через DB2 Connect.

Явное доверенное соединение дает пользователю временные привилегии роли точно так же, как и неявное. Дополнительно явное доверенное соединение позволяет изменять ID авторизации при выполнении различных действий во время этого соединения. Изменение ID авторизации при явном доверенном соединении называется *переключением пользователей*. ID авторизации, на которые можно переключаться, и требуется ли пароль при переключении на данный ID авторизации, определяется как часть доверенного контекста, позволяющего создать это доверенное соединение.

Переключение пользователей может существенно снизить дополнительные издержки при совместном использовании соединения несколькими пользователями, особенно для имен пользователей, для которых не требуется пароль, поскольку в этом случае сервер баз данных не выполняет аутентификацию ID авторизации. Однако при использовании этой функции надо быть полностью уверены в том, что ваша прикладная программа не допускает переключение на ID авторизации без проверки и аутентификации этого ID авторизации. В противном случае вы создадите брешь в защите вашей системы.

При соединении через DB2 Connect с использованием CLI или JDBC, включая соединения, устанавливаемые при использовании XA, можно создавать явные доверенные соединения и выполнять переключение пользователей. Создание явного доверенного соединения и переключение пользователей требует установки специальных атрибутов соединения. Это означает, что существующие прикладные программы необходимо изменять, чтобы воспользоваться преимуществами явных доверенных соединений.

За исключением только что описанных различий доверенные соединения (как явные, так и неявные) можно использовать точно так же, как и обычные соединения. Однако, закончив использование явного доверенного соединения, нужно явным образом отключить его, даже если оно уже разорвано или отключено неявно. В противном случае могут быть не высвобождены ресурсы, использованные соединением. Это не относится к неявным доверенным соединениям.

Примечание:

1.

Внимание: При переключении пользователей без ввода пароля не выполняется аутентификация на сервере баз данных. Ваша прикладная программа не должна позволять переключение на другой ID авторизации, если он уже не проверен и не аутентифицирован этой программой. В противном случае возможно нарушение защиты.

2. Явные доверенные соединения не должны использовать аутентификацию CLIENT. Это не относится к неявным доверенным соединениям.
3. Прикладные программы, использующие явные доверенные соединения, должны запускаться на защищенных паролями компьютерах, доступ к которым возможен только у авторизованных сотрудников. Это не относится к неявным доверенным соединениям.

Создание и прекращение доверенного соединения при помощи CLI

Если сервер баз данных, с которым вы соединяетесь, это допускает, при соединении через CLI можно явно создать доверенное соединение.

В этой процедуре предполагается, что менеджер транзакций не используется. Если вы используете менеджер транзакций XA, надо только убедиться, что в конфигурации менеджера транзакций в момент вызова `xa_open` для параметра конфигурации `TCSTX` задано значение `TRUE`. Если это так, любое соединение, которое можно явно задать как доверенное, будет доверенным. Чтобы проверить, является ли соединение явно доверенным, посмотрите шаг 3.

- База данных, с которой устанавливается соединение, должна поддерживать доверенные контексты.

- Должен быть определен доверенный контекст, который будет распознавать клиент в качестве клиента, которому можно доверять.
- Вы должны знать ID авторизации системы, указанный в доверенном контексте. ID авторизации системы доверенного соединения представляет собой ID авторизации, который вы сообщаете серверу в качестве имени пользователя при создании соединения. Чтобы соединение было доверенным в соответствии с конкретным доверенным контекстом, ID авторизации системы должен быть тем, который указан в этом доверенном контексте. Действительный ID авторизации системы и пароль для этого ID запрашивайте у своего администратора защиты.

В примерах этих действий используется язык C и предполагается, что `conn` - это указатель на допустимый, но еще не используемый хэндл соединения. Для переменной `rc` предполагается тип данных `SQLRETURN`.

1. Наряду с прочими атрибутами соединения, которые можно задать для обычного соединения, задайте атрибут соединения `SQL_ATTR_USE_TRUSTED_CONTEXT` со значением `SQL_TRUE`, вызвав функцию `SQLSetConnectAttr`.


```
rc = SQLSetConnectAttr(
    conn,
    SQL_ATTR_USE_TRUSTED_CONTEXT, SQL_TRUE, SQL_IS_INTEGER
);
```
2. Соединитесь с базой данных, как и в случае обычного соединения, вызвав функцию `SQLConnect` для экземпляра. В качестве имени пользователя используйте ID авторизации системы, а в качестве пароля - соответствующий пароль. Обязательно просмотрите сообщения об ошибках и предупреждения (особенно перечисленные в Табл. 5).

Таблица 5. Сообщения об ошибках, указывающие на неудачное создание доверенного соединения

SQLCODE	SQLSTATE	Смысл
SQL20360W	01679	Соединение не удалось установить как доверенное соединение. Вместо доверенного установлено обычное соединение.

Если в сообщениях об ошибках не оговорено иного, то соединение установлено, и оно является явным доверенным.

3. (Необязательно) Вы можете проверить, является ли установленное соединение явным доверенным соединением, проверив значение атрибута соединения `SQL_ATTR_USE_TRUSTED_CONTEXT` при помощи функции `SQLGetConnectAttr`. Если для него установлено `SQL_TRUE`, соединение является явным доверенным соединением.
4. Закончив использование соединения, обязательно нужно явно отключить его, даже если уже разорвано или отключено неявно. Если вы явным образом не отключите явное доверенное соединение, могут не быть высвобождены отдельные используемые им ресурсы.

Примечание:

1. Явные доверенные соединения не должны использовать аутентификацию `CLIENT`. Это не относится к неявным доверенным соединениям.
2. Прикладные программы, использующие явные доверенные соединения, должны запускаться только на защищенных паролями компьютерах, доступ к которым возможен только у авторизованных сотрудников. Это не относится к неявным доверенным соединениям.

Переключение пользователей в доверенном соединении при помощи CLI

В явном доверенном соединении можно переключать пользователей при помощи интерфейса командной строки. Описание того, что означает переключение пользователей, использующих доверенное соединение, смотрите в соответствующем разделе, указанном в дополнительных ссылках.

- Соединение должно быть создано успешно как явное доверенное соединение.
- Это явное доверенное соединение не должно участвовать в транзакции.
- Доверенный контекст, позволивший создание данного явного доверенного соединения, должен быть сконфигурирован так, чтобы позволять переключение на необходимый ID авторизации.

В примерах этих действий используется язык Си и предполагается, что **conn** - это указатель на установленное явное доверенное соединение. Для переменной *rc* предполагается тип данных SQLRETURN. Предполагается, что переменная *newuser* - это указатель на символьную строку, содержащую ID авторизации пользователя, на которого вы хотите переключиться. Предполагается, что переменная *passwd* - это указатель на символьную строку, содержащую пароль для этого ID авторизации.

1. Вызовите функцию SQLSetConnectAttr, чтобы задать атрибут SQL_ATTR_TRUSTED_CONTEXT_USERID. Задайте в качестве его значения ID авторизации, на который вы хотите переключиться.

```
rc = SQLSetConnectAttr(  
    conn,  
    SQL_ATTR_TRUSTED_CONTEXT_USERID, newuser, SQL_NTS  
);  
//Просмотрите сообщения об ошибках
```

Обязательно просмотрите сообщения об ошибках и предупреждения (особенно перечисленные в Табл. 6).

Таблица 6. Сообщения об ошибках, указывающие на неудачное задание нового ID авторизации при переключении пользователей

SQLCODE	Смысл
CLI0106E	Соединение находится в несоединенном состоянии.
CLI0197E	Соединение не является доверенным соединением.
CLI0124E	Ошибка при вводе значения. Проверьте, что оно не пустое или, например, не слишком длинное.
CLI0196E	Соединение задействовано в единице работы, не допускающей переключение пользователей. Для возможности переключения пользователей соединение не должно участвовать в транзакции.

2. (Необязательно, если только для доверенного контекста, разрешившего это доверенное соединение, не требуется пароль для ID авторизации, на который вы переключаетесь) Вызовите функцию SQLSetConnectAttr, чтобы задать атрибут SQL_ATTR_TRUSTED_CONTEXT_PASSWORD. Задайте в качестве его значения пароль для нового ID авторизации.

```
rc = SQLSetConnectAttr(  
    conn,  
    SQL_ATTR_TRUSTED_CONTEXT_PASSWORD, passwd, SQL_NTS  
);  
//Просмотрите сообщения об ошибках
```

Обязательно просмотрите сообщения об ошибках и предупреждения, перечисленные в таблицах Табл. 6 на стр. 44 и Табл. 7.

Таблица 7. Сообщения об ошибках, указывающие на неудачное задание пароля при переключении пользователей

SQLCODE	Смысл
CLI0198E	Атрибут SQL_ATTR_TRUSTED_CONTEXT_USERID еще не задан.

3. Действуйте так же, как в случае обычного соединения. Если вы используете менеджер транзакций XA, попытка переключения пользователя выполняется как часть нового требования; в противном случае попытка переключения пользователя выполняется непосредственно перед инициализацией следующего вызова функции, обращающейся к базе данных (например, SQLExecDirect). В любом случае в дополнение к обычным ошибкам и предупреждениям проверяйте наличие ошибок, перечисленных в Табл. 8. Ошибки, перечисленные в Табл. 8 свидетельствуют о неудачном переключении пользователя.

Таблица 8. Ошибки, указывающие на неудачное переключение пользователей

SQLCODE	Смысл
SQL1046N	Доверенный контекст, позволивший создание данного явного доверенного соединения, не сконфигурирован так, чтобы позволять переключение на необходимый ID авторизации. Вы не сможете переключиться на этот ID авторизации, пока не будет изменен доверенный контекст.
SQL30082N	Введенный пароль недействителен для ID авторизации, на который вы переключаетесь.
SQL0969N с собственной ошибкой -20361	Есть ограничение на уровне базы данных, препятствующее переключению пользователя.

Если переключение пользователя завершилось неудачно, соединение будет находиться в неподключенном состоянии до тех пор, пока вы успешно не переключитесь на другого пользователя. В неподключенном состоянии на доверенном соединении можно переключать пользователей, но доступ к серверу баз данных запрещен. Соединение будет оставаться в неподключенном состоянии до тех пор, пока вы не выполните успешно переключение пользователей.

Примечания:

1. **Важно:** Переключение пользователей без ввода пароля обходит аутентификацию сервера баз данных. Ваша прикладная программа не должна позволять переключение на другой ID авторизации, если он уже не проверен и не аутентифицирован этой программой. В противном случае возможно нарушение защиты.
2. Задание пустого значения для атрибута SQL_ATTR_TRUSTED_CONTEXT_USERID эквивалентно заданию ID авторизации системы доверенного контекста (ID пользователя, использованного при создании явного доверенного соединения).
3. После успешной установки значения для атрибута SQL_ATTR_TRUSTED_CONTEXT_USERID в явном доверенном соединении это соединение немедленно переустанавливается. Результат такой переустановки эквивалентен созданию нового соединения с использованием исходных атрибутов этого соединения. Такая переустановка происходит, даже если установить для атрибута соединения ID авторизации системы, пустое значение или текущее значение этого атрибута.

4. Если задан атрибут `SQL_ATTR_TRUSTED_CONTEXT_PASSWORD`, пароль будет проверяться во время переключения пользователя, даже если доверенный контекст, позволивший доверенное соединение, не требует аутентификации при переключении пользователя на этот ID авторизации. Это ведет к излишним издержкам. Это правило не относится к ID авторизации системы в доверенном контексте. Если при переключении на ID авторизации системы в доверенном контексте не требуется аутентификация, она и не выполняется, даже если передан пароль.

Информация об аутентификации в DB2 Connect

Как администратор DB2 Connect, вместе с администратором базы данных хоста или System i, вы можете определить, где проверяются имена пользователей и пароли:

- На клиенте
- На хосте или сервере System i
- При одноразовом входе в систему и проверке в третьей системе (Kerberos).

Примечание: Если удаленный клиент не указал тип аутентификации, то по умолчанию на клиенте принимается тип `SERVER_ENCRYPT`. Если сервер отклоняет этот тип, то клиент пытается еще раз установить соединение с другим значением, возвращенным сервером. Во избежание ненужных затрат на передачу подобной информации по сети всегда указывайте тип аутентификации на клиенте.

Начиная с версии 8.2.2 DB2 Connect (эквивалентна версии Version 8.1 FixPak 9) шлюз больше не является пассивным участником в процессе аутентификации. Теперь он играет активную роль. Тип аутентификации, указанный в записи каталога баз данных на шлюзе, переопределяет тип аутентификации, заданный в каталоге на клиенте. На клиенте, на шлюзе и на сервере должны быть заданы совместимые типы. Если тип аутентификации в записи каталога баз данных на шлюзе не указан, по умолчанию на сервере будет требоваться аутентификация `SERVER`. Однако если сервер не поддерживает аутентификацию `SERVER`, будет выполнено согласование между сервером и клиентом. Это поведение отличается от поведения на клиенте, где, если тип аутентификации не задан, по умолчанию предполагается `SERVER_ENCRYPT`.

Тип аутентификации, внесенный в каталог на шлюзе, не используется, если на клиенте для API Set Client задана опция `DB2NODE` или `SQL_CONNECT_NODE`. В таких случаях согласование ведется непосредственно между клиентом и сервером.

В DB2 Connect разрешены следующие типы аутентификации:

CLIENT

Имя пользователя и пароль проверяются на клиенте.

SERVER

Имя пользователя и пароль проверяются на сервере баз данных хоста или System i.

SERVER_ENCRYPT

Как и при аутентификации типа `SERVER`, имя пользователя и пароль проверяются на сервере баз данных хоста или System i, но передаваемые пароли шифруются на клиенте.

DATA_ENCRYPT

Обеспечивает возможность шифрования данных для связи клиент/сервер.

KERBEROS

Позволяет клиенту зарегистрироваться на сервере с помощью аутентификации Kerberos, вместо того чтобы указывать обычное сочетание

из ID пользователя и пароля. Такой тип аутентификации требует, чтобы и на сервере, и на клиенте была установлена поддержка Kerberos.

Аутентификация Kerberos уникальна в том смысле, что клиент не передает свой ID пользователя и пароль непосредственно на сервер. Вместо этого, Kerberos выступает в роли промежуточного механизма аутентификации. Пользователь вводит ID и пароль один раз на терминале клиента, а Kerberos проверяет их правильность. После этого Kerberos автоматически и в защищенном режиме передает информацию авторизации пользователя соответствующим локальным и сетевым службам. Это означает, что пользователю не нужно еще раз вводить ID и пароль для регистрации на удаленном сервере DB2. Для аутентификации с помощью Kerberos необходимо, чтобы и в DB2 Connect, и на сервере баз данных, с которым устанавливается соединение, присутствовала поддержка Kerberos.

Примечание: Тип аутентификации GSSPLUGIN не поддерживается.

Поддержка Kerberos

Уровень аутентификации Kerberos, который создает систему извещений, встроен в механизм Active Directory Windows 2000. Сторона клиента и сторона сервера программы соединяются соответственно при помощи модулей клиента и сервера Kerberos SSP (Security Support Provider). Интерфейс Security Support Provider Interface (SSPI) обеспечивает интерфейс высокого уровня для Kerberos SSP и других протоколов защиты.

Обычная установка

Для конфигурирования DB2 с аутентификацией Kerberos, настройте:

- политику авторизации для DB2 (как службы) в Active Directory, используемую в сети, и
- доверенное соединение между центрами распределения ключей (KDC) Kerberos

В простейшем сценарии надо сконфигурировать хотя бы одно доверенное соединение KDC, а именно соединение между KDC, который управляет рабочей станцией, и системой System i, OS/390 или z/OS. OS/390 Версия 2 Выпуск 10 или z/OS Версия 1 Выпуск 2 обеспечивают обработку извещений Kerberos при помощи своей утилиты RACF, что позволяет хосту играть роль KDC в UNIX.

DB2 Connect, как обычно, выполняет функцию маршрутизатора в трехуровневой модели. Когда используется защита Kerberos, DB2 Connect не участвует в аутентификации. Она только передает маркер защиты клиента в DB2 для i5/OS или в DB2 для OS/390 и z/OS. Поэтому шлюз DB2 Connect не должен входить в сферу ответственности клиента или хоста Kerberos.

Совместимость с предыдущими уровнями

Минимальные требования к DB2 для поддержки Kerberos:

IBM data server:

Версия 8

DB2 Connect:

Версия 8

DB2 Universal Database (UDB) для OS/390 и z/OS:

Версия 7

Сведения о защите DB2 Connect в OS/390 и z/OS

В этой главе описываются возможности защиты DB2 Connect, в том числе типы аутентификации и параметры защиты. Здесь приводятся также дополнительные советы относительно защиты пользователям DB2 for OS/390 and z/OS.

Дополнительные советы и замечания по организации защиты для OS/390 и z/OS

В этом разделе приводятся некоторые дополнительные советы и замечания по организации защиты соединений DB2 Connect с серверами баз данных DB2 для OS/390 и z/OS.

Поле Extended Security

Убедитесь, что в поле "DB2 OS/390 and z/OS Extended Security" задано значение YES. Это поле выводится на панели DSNTIPR DB2 для OS/390 и z/OS.

Расширенные коды защиты

В версиях DB2 Universal Database for z/OS and 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 z/OS and 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 z/OS and OS/390 ZPARM должен иметь значение YES. На панели установки DB2 Universal Database for z/OS and OS/390 DSN6SYSP задайте EXTSEC=YES. Для задания этого параметра можно также использовать панель 1 DDF (DSNTIPR). Значение по умолчанию - EXTSEC=NO. В случае истечения срока действия пароля операционные системы Windows, Linux и UNIX и прикладные программы Web, использующие DB2 Connect, получают сообщение об ошибке SQL30082.

Поле TCP/IP security already verified

Если вы хотите обеспечить поддержку опции защиты DB2 AUTHENTICATION=CLIENT, на панели установки DB2 Universal Database for z/OS and OS/390 DSNTIP4 (панель 2 DDF) задайте в поле "TCP/IP already verified security" 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 z/OS and OS/390 введена новая опция связывания DYNAMICRULES(BIND), которая позволяет по выбору выполнять динамический SQL с авторизацией владельца или же того, кто выполнял связывание.

В DB2 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.

Поддержка смены пароля

Если для ID пользователя истек срок действия пароля, и оператор SQL CONNECT возвращает сообщение об ошибке, (например, SQLCODE: -30082, код причины: 1), в DB2 Connect можно изменить пароль удаленно. DB2 Universal Database for z/OS and OS/390 может сменить пароль для вас через DRDA, вызвав следующий оператор CONNECT:

```
CONNECT TO <база_данных> USER <id_пользователя> USING <пароль>  
NEW <новый_пароль> CONFIRM <новый_пароль>
```

Пароль можно изменить также при помощи диалогового окна Изменение пароля Ассистента конфигурирования DB2.

Типы защиты, поддерживаемые DB2 Connect

В этом разделе перечислены различные сочетания параметров аутентификации и защиты, поддерживаемые DB2 Connect.

Типы защиты для соединений TCP/IP

Протокол связи TCP/IP не поддерживает параметры защиты на уровне сетевого протокола. Положение, в котором происходит аутентификация, определяется типом аутентификации. Соединения DB2 Connect поддерживают только сочетания, перечисленные в следующей таблице. Параметр аутентификации задается в записи каталога баз данных на сервере DB2 Connect.

Таблица 9. Допустимые сценарии защиты

Сценарий	Параметр аутентификации	Проверка
1	CLIENT	Клиент
2	SERVER	Сервер баз данных хоста или System i
3	SERVER_ENCRYPT	Сервер баз данных хоста или System i
4	KERBEROS	Защита Kerberos
5	DATA_ENCRYPT	Сервер баз данных хоста или System i

Обсуждение типов защиты

Следующее обсуждение относится к соединениям, описанным выше и перечисленным в Табл. 9. Все сценарии подробно рассмотрены ниже:

- В сценарии 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 аутентификация пользователя происходит там, где он первоначально регистрируется. По сети посылаются ID пользователя, но не пароль. Используйте этот тип защиты, только если на всех рабочих станциях клиентов имеются адекватные механизмы защиты, которым можно доверять.

- В сценарии 2 имя пользователя и пароль проверяются только на сервере баз данных хоста или System i. 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 System i.

- Сценарий 3 совпадает со сценарием 2, за исключением того, что ID пользователя и пароль зашифрованы.
- В сценарии 4 клиент получает от Kerberos KDC квитанцию Kerberos. Эта квитанция передается в неизменном виде через DB2 Connect на сервер, где она проверяется.
- Сценарий 5 совпадает со сценарием 3, за исключением того, что данные пользователя также зашифрованы.

Глава 6. Связывание программ и утилит (DB2 Connect)

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 (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 System i.

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

- ddcsmvs.lst для OS/390 или z/OS
- 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 продукт сервера DB2 Connect установлен, утилиты DB2 Connect надо связать с каждым сервером баз данных хоста или System i, чтобы их можно было использовать с этой системой. Если у клиентов один и тот же уровень пакетов FixPack, связывать утилиты нужно только один раз, независимо от числа платформ клиентов.

Например, если у вас 10 клиентов Windows и 10 клиентов AIX, подключенных к DB2 Universal Database (UDB) for OS/390 and z/OS через DB2 Connect Enterprise Server Edition на сервере Windows, выполните одно из следующих действий:

- Свяжите ddcsmvs.lst с одного из клиентов Windows.
- Свяжите ddcsmvs.lst с одного из клиентов AIX.
- Свяжите ddcsmvs.lst с сервера DB2 Connect.

В этом примере предполагается, что:

- У всех клиентов один и тот же служебный уровень. Если это не так, может дополнительно понадобится выполнить связывание с каждого клиента со своим служебным уровнем
- У сервера тот же служебный уровень, что и у клиентов. Если это не так, необходимо также выполнить связывание с сервера.

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 System i, с которым вы выполняете связывание, сделайте следующее:

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

OS/390 или z/OS

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 выполнить связывание.

B) Если первоначальное связывание выполнял кто-то еще, а повторное связывание выполняете вы, 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 System i, ddcsmvs.lst - это файл списка связывания z/OS, а *путь* - это каталог, где находится файл списка связывания.

Например, *диск:\sqllib\bnd* для операционных систем платформ Windows, или *НАЧАЛЬНЫЙ_КАТАЛОГ/sqllib/bnd/* для всех операционных систем Linux и 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 @файл_связывания.lst
```

Например:

```
ddcspkgn @ddcsmvs.lst
```

может быть таким:

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

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
```

Примечание:

- Использование опции связывания `sqlerror continue` обязательно, однако если вы связываете программы при помощи средств DB2 или процессора командной строки (CLP), 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 System i можно ожидать появления предупреждений.
 - Если вы подключаетесь к базе данных DB2 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 System i. Рекомендуется также при подключении к базе данных DB2 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, может понадобиться связать утилиты этих клиентов с DB2 Connect.

Глава 7. Многоузловые изменения

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 мониторы 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 монитора транзакций прикладная программа должна затребовать принятие или откат, используя API монитора транзакций, 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:

- DB2 для Linux, UNIX и Windows Версии 8 или новее
- DB2 Universal Database (UDB) for OS/390 and z/OS Версии 7
- DB2 for z/OS Версии 8

- DB2 for i5/OS требует OS/400 Версии 5 Выпуска 1 или новее.

Распределенная транзакция может производить изменения на любом наборе поддерживаемых серверов баз данных. Например, в одной транзакции прикладная программа может изменять несколько таблиц в базе данных DB2 в Windows, базе данных DB2 for OS/390 и z/OS и в базе данных DB2 for i5/OS.

Установка многоузлового изменения с помощью Центра управления

Для многоузлового изменения можно использовать Центр управления.

Чтобы включить многоузловое изменение:

1. Запустите Центр управления.
2. Щелкните по символу [+], чтобы развернуть дерево.
3. Правой кнопкой мыши щелкните по экземпляру, который хотите конфигурировать. Появится всплывающее меню.
4. Выберите пункт **Многоузловое изменение** —> **Конфигурировать**. Откроется мастер по многоузловому изменению.
5. Выберите **Использовать монитор транзакций, указанный ниже** и задайте монитор транзакций. В этом поле выводятся умолчания для монитора транзакций, который вы включили. Если вы не хотите использовать монитор транзакций, выберите **Не использовать монитор TP**.

Нажмите кнопку **Далее**.

6. Если вы используете монитор транзакций, задайте параметры менеджера точек синхронизации. Если вы не используете монитор транзакций, задайте базу данных менеджера транзакций.
7. Нажмите кнопку **Готово**.

Проверка многоузлового изменения с помощью Центра управления

С помощью Центра управления вы можете проверить конфигурацию многоузлового изменения.

Чтобы проверить конфигурацию многоузлового изменения, выполните следующие действия:

1. Щелкните по экземпляру правой кнопкой мыши и выберите из всплывающего меню **Многоузловое изменение** —> **Проверить**. Откроется окно Проверка многоузлового изменения.
2. Выберите базу данных, которую хотите проверить, в списке **Доступные**. Кнопки со стрелками (> и >>) в центре окна позволяют перемещать выбранные базы данных в подокно **Выбранные** или из него. Можно также изменять выбранные ID пользователя и пароль, редактируя их прямо в окне **Выбранные**.
3. Закончив выбор, нажмите кнопку **ОК**. Откроется окно Результаты проверки многоузлового изменения.
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. Чтобы закрыть окно, нажмите кнопку **Заккрыть**.
5. Чтобы закрыть окно Проверка многоузлового изменения, нажмите кнопку **Заккрыть**.

Многоузловое изменение и менеджер точки синхронизации

Серверам баз данных хоста и System i для участия в распределенной транзакции, начатой Linux, Windows, UNIX или прикладными программами Web, необходим DB2 Connect. Кроме того, многие сценарии многоузлового изменения, в котором участвуют серверы баз данных хоста и System i, 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 (TCP/IP) и использованием монитора транзакций. В следующей таблице приведена сводка сценариев, которые требуют SPM. В этой таблице показано также, требуется ли DB2 Connect для любого вида доступа к хосту или System i с компьютеров Intel или UNIX. Для многоузловых изменений требуется DB2 Connect SPM, если используется монитор транзакций.

Таблица 10. Сценарии многоузлового изменения, требующие SPM - TCP/IP

Используется монитор транзакций	Требуется менеджер точки синхронизации	Требуемый продукт (выберите один)	Поддерживаемые базы данных хоста и System i
Да	Да	Продукт сервера DB2 Connect Применяется лицензия на DB2 Enterprise Server Edition с DB2 Connect	DB2 Universal Database (UDB) for OS/390 and z/OS V7 DB2 UDB for z/OS V8 или новее
Нет	Нет	DB2 Connect Personal Edition Продукт сервера DB2 Connect Применяется лицензия на DB2 Enterprise Server Edition с DB2 Connect	DB2 UDB for OS/390 and z/OS V7 DB2 UDB for z/OS V8 или новее

Примечание: Распределенная транзакция может производить изменения на любом наборе поддерживаемых серверов баз данных. Например, прикладная программа может в одной транзакции изменять несколько таблиц в базе данных DB2 в Windows, базе данных DB2 for OS/390 и базе данных DB2 for i5/OS.

Настройка DB2 Connect с менеджером транзакций стандарта XA

В этом разделе описываются необходимые шаги конфигурирования для использования серверов баз данных S/390, System i и zSeries с монитором транзакций.

У вас должен работать монитор транзакций, должна быть установлена DB2 Connect, а также сконфигурировано и протестировано соединение с сервером баз данных хоста или System i.

Чтобы сконфигурировать DB2 Connect для использования серверов баз данных S/390, System i и zSeries в мониторе транзакций, выполните следующие действия:

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.
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 по монитору.
3. При необходимости измените значения по умолчанию параметров конфигурации менеджера точек синхронизации (SPM) DB2 Connect. Серверы баз данных хостов и System i (Версия 5, Выпуск 3 и более ранние) не поддерживают интерфейс XA. System i Версия 5, Выпуск 4 и последующие имеют полную поддержку XA. SPM - это компонент DB2 Connect, преобразующий протокол двухфазного принятия стандарта XA в протокол двухфазного принятия, используемый серверами баз данных хоста и System i. 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 и z/OS 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.

Поддержка связанных транзакций в DB2 Connect

В DB2 Connect есть поддержка свободно связанных транзакций, предназначенная для пользователей, реализующих распределенные программы XA, которые обращаются к DB2 for i5/OS Версии 5 Выпуска 4 или новее и DB2 Universal Database (UDB) for OS/390 and z/OS Версии 7 или новее. Эта поддержка позволяет разным ветвям одной глобальной транзакции использовать общее пространство блокировок в DB2 для OS/390 и z/OS.

Поддержка свободно связанных транзакций предназначена только для программ .NET и COM+.

Это уменьшает вероятность ситуации, когда одна ветвь распределенной транзакции сталкивается с истечением срока блокировки или тупиковой ситуацией из-за другой ветви той же глобальной транзакции.

Глава 8. Перемещение данных с помощью DB2 Connect

При работе в сложной среде, где требуется перемещать данные между системой баз данных и рабочей станцией, можно использовать DB2 Connect - шлюз для передачи данных между хостом и рабочей станцией (смотрите раздел рис. 8).

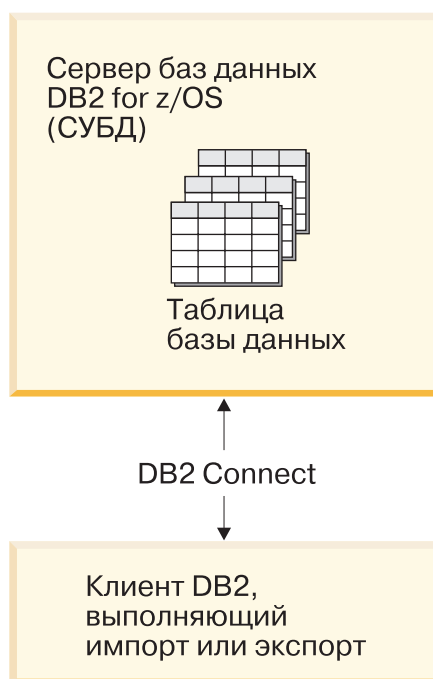


Рисунок 8. Импорт/экспорт через DB2 Connect

Утилиты экспорта и импорта DB2 позволяют перемещать данные из базы данных сервера хоста или System i в файл на рабочей станции DB2 Connect и в обратном направлении. После этого данные можно использовать с любой другой программой или системой управления реляционными базами данных, поддерживающей этот формат экспорта или импорта. Например, можно экспортировать данные из базы данных сервера хоста или System i в файл PC/IXF, а затем импортировать их в базу данных DB2 for Windows.

Операции экспорта и импорта можно выполнять с клиента баз данных или рабочей станции DB2 Connect.

Примечание:

1. Экспортируемые или импортируемые данные должны соответствовать ограничениям на размер и тип данных, действующим для обеих баз данных.
2. Для повышения производительности импорта можно использовать составные запросы. Задайте в утилите импорта модификатор типа файла compound, чтобы сгруппировать заданное число операторов запросов в блок. Этим можно сократить сетевые затраты и уменьшить время ответа.

При использовании DB2 Connect для операций экспорта и импорта должны соблюдаться следующие условия:

- Файл должен быть типа PC/IXF.
- Для возможности импорта данных на сервер назначения на нем нужно создать таблицу назначения с атрибутами, совместимыми с импортируемыми данными. Атрибуты исходной таблицы можно получить при помощи утилиты db2look. Операция импорта через DB2 Connect не может создать таблицу; единственная поддерживаемая опция для нее - INSERT.

Если какие-либо из этих условий не будут выполнены, операция завершится неудачно и будет возвращено сообщение об ошибке.

Примечание: Определения индексов не сохраняются при экспорте и не используются при импорте.

При экспорте смешанных данных (столбцы которых содержат и однобайтные, и двухбайтные данные) нужно учитывать следующее:

- В системах, где данные хранятся в EBCDIC (Extended Binary Coded Decimal Interchange Code - расширенный двоичный код для обмена информацией) (в MVS, OS/390, OS/400, VM и VSE), начало и конец двухбайтных данных помечен символами включения и отключения. При определении длины столбцов для таблиц баз данных обязательно выделите для этих символов достаточно места.
- Если данные столбцов не отвечают согласованному шаблону, рекомендуется использовать столбцы переменной длины.

Перемещение данных с рабочей станции на сервер хоста

Для перемещения данных в базу данных сервера хоста или System i:

1. Экспортируйте данные из таблицы DB2 в файл PC/IXF.
2. При помощи опции INSERT импортируйте файл PC/IXF в совместимую таблицу базы данных сервера хоста.

Для перемещения данных из базы данных сервера хоста на рабочую станцию:

1. Экспортируйте данные из таблицы базы данных сервера хоста в файл PC/IXF.
2. Импортируйте файл PC/IXF в таблицу DB2.

Пример

В следующем примере показано, как переместить данные с рабочей станции в базу данных сервера хоста или System i.

Экспортируйте данные во внешний формат IXF при помощи команды:

```
db2 export to staff.ixf of ixf select * from userid.staff
```

Установите соединение DRDA с базой данных DB2 назначения, введя команду:

```
db2 connect to cbc664 user admin using xxx
```

Создайте таблицу назначения в экземпляре базы данных DB2 (если она еще не существует):

```
CREATE TABLE mydb.staff (ID SMALLINT NOT NULL, NAME VARCHAR(9),
DEPT SMALLINT, JOB CHAR(5), YEARS SMALLINT, SALARY DECIMAL(7,2),
COMM DECIMAL(7,2))
```

Чтобы импортировать данные, введите следующую команду:

```
db2 import from staff.ixf of ixf insert into mydb.staff
```

Из файла в формате IXF будет прочитана каждая строка, и для ее вставки в таблицу mydb.staff будет вызван оператор SQL INSERT. Вставка отдельных строк будет продолжаться, пока в таблицу назначения не будут перемещены все данные.

Подробную информацию смотрите в публикации IBM Redbooks "Moving Data Across the DB2 Family". Эту публикацию Redbooks можно найти по адресу:
<http://www.redbooks.ibm.com/redbooks/SG246905>.

Глава 9. Отображение SQLCODE

Различные программные продукты для реляционных баз данных фирмы IBM не всегда выдают одинаковые SQLCODE для похожих 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 хоста или System i в соответствующие SQLCODE DB2.

Чтобы отключить отображение SQLCODE, укажите **NOMAP** в строке параметров каталога DCS.

Если вы переносите прикладную программу непосредственно с сервера баз данных хоста или System (например, DB2 Universal Database для OS/390 и z/OS), то может понадобиться отключить отображение SQLCODE. Это позволит использовать прикладную программу, не меняя соответствующие ей SQLCODE.

Отключение отображения SQLCODE

Чтобы отключить отображение SQLCODE, укажите **NOMAP** в строке параметров каталога DCS.

Если вы переносите прикладную программу непосредственно с сервера баз данных хоста или System i (например, DB2 Universal Database (UDB) для OS/390 и z/OS), то 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 хоста или System i в соответствующие SQLCODE DB2.

Следующие файлы содержат копии отображения SQLCODE по умолчанию:

- dcs1dsn.map отображает коды SQLCODE DB2 Universal Database (UDB) for OS/390 and z/OS и DB2 for z/OS.
- dcs1ari.map отображает SQLCODE DB2 UDB для VSE и VM.
- dcs1sq.map отображает коды SQLCODE DB2 UDB for iSeries и DB2 for i5/OS.

Для DB2 в операционных системах Linux или UNIX отображение не требуется.

1. Если вы хотите переопределить отображение SQLCODE по умолчанию или используете сервер баз данных хоста или System i, 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.
2. В строке параметров каталога DCS укажите имя нового файла отображения SQLCODE.
3. Файл отображения - это простой ASCII-файл, и его можно создавать и редактировать с помощью ASCII-редактора. При начальной установке этот файл находится в каталоге map 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]]

Здесь *входной_код* имеет одно из следующих значений:

sqlcode

SQLCODE сервера баз данных хоста или System i.

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 System i. *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 **сс** 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 интерпретируется так:

- 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 System *i* поддерживают только элементы типа CHAR. Символ **i** задается в нижнем регистре.
- n* Одно или несколько чисел, указывающих, какие элементы сервера баз данных хоста или System *i* используются. Они указывают требуемый порядок размещения элементов в выходной области SQLCA. Число означает порядковый номер элемента сервера баз данных хоста или System *i*; порядок этих чисел соответствует порядку, в котором элементы будут помещены в SQLCA.

Например, сервер баз данных хоста или System *i* 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. 9 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 9. Файл отображения SQLCODE

Эти описания соотносятся с соответствующим номером строки в предыдущем примере:

1. SQLCODE -007 отображается в -007. Первый входной элемент, полученный от сервера баз данных хоста или System i, 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 System i, 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 System i, 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 System i, 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 System i, 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 10. Мониторинг систем баз данных и DB2 Connect

В этой главе обсуждаются несколько способов мониторинга соединений и производительности в среде, где используется DB2 Connect. Тип выполняемого мониторинга зависит от конкретной операционной системы.

Монитор соединений с удаленными клиентами

Используя системный монитор баз данных с продуктом сервера DB2 Connect, таким как DB2 Connect Enterprise Edition, можно выполнять мониторинг соединений с удаленными клиентами. Чтобы выполнять мониторинг клиентов, локальных для сервера DB2 Connect (которые работают в той же системе), надо задать следующую переменную:

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

Например, когда ошибка возникает на системе хоста или System i, 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 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.

Просмотр состояния переключателей монитора

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

Отслеживание производительности с помощью монитора производительности Windows

В операционных системах Windows 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.rmc. Можно сохранить столько файлов РМС, сколько вам надо для отражения разных факторов, влияющих на производительность системы.

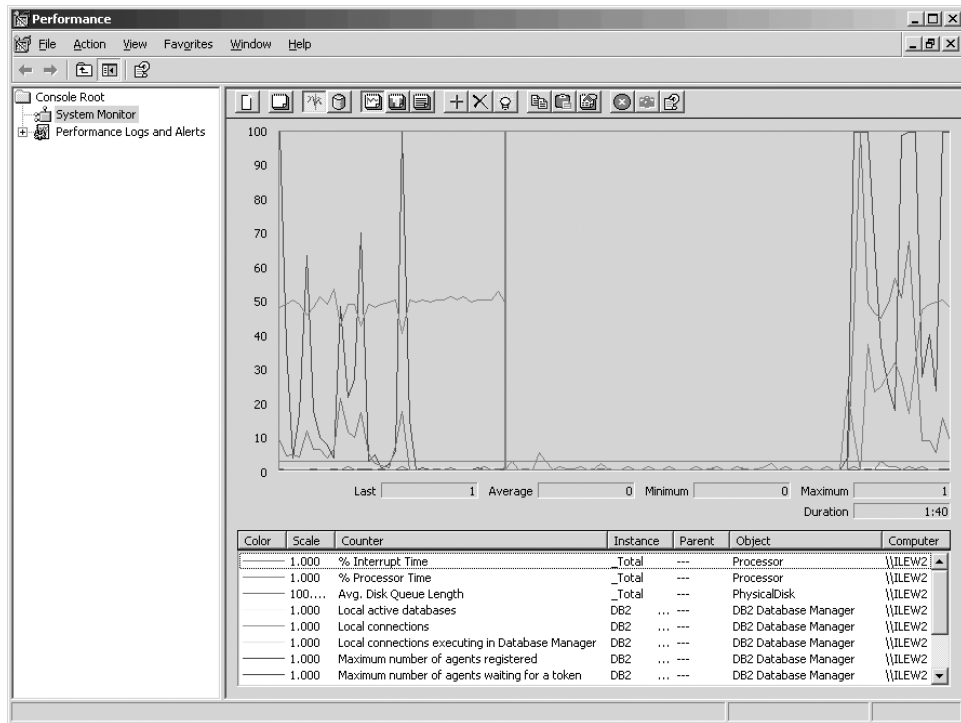


Рисунок 10. Монитор производительности

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

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. Это следующие команды:

- 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 = 12-15-2001 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 = 12-15-2001 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 = MVSDB
Имя базы данных DCS = DCSDB
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 = 12-15-2001 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 = 12-15-2001 10:28:25.592631
Временная отметка запуска единицы работы = 12-15-2001 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 = 12-15-2001 10:29:06.142790
Время конца последней операции = 12-15-2001 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 = 12-15-2001 10:29:06.142790
Временная отметка конца оператора = 12-15-2001 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)

```

Состояние прикладной программы DCS

Системный монитор поддерживает три формы команды LIST DCS APPLICATIONS:

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

В выводе этой команды формат значений для ID программы хоста и ID программы клиента может различаться в зависимости от версии базы данных хоста или iSeries и уровня поддержки TCP/IP.

Таблица 11. Формат ID программы в зависимости от версии хоста и уровня поддержки TCP/IP

Сценарий	Формат ID программы
Клиенты обращаются к серверам данных с уровнем поддержки RDBM меньше 7	G91A0D3A.P8BC.060306212019
Клиенты обращаются к серверам данных с уровнем поддержки менеджера RDB 8 или выше по TCP/IP v4	9.26.13.61.65289.060306213816
Клиенты обращаются к серверам данных с уровнем поддержки менеджера RDB 8 или выше по TCP/IP v6	2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741

LIST DCS APPLICATIONS

Чтобы просмотреть регистрируемую монитором информацию уровня программы, введите команду DB2 LIST DCS APPLICATIONS.

Для соединения TCP/IP (DB2 Connect с DB2 Universal Database for z/OS and OS/390) будет получена следующая информация:

ID авт.	Имя программы	Хэндл программы	ID программы хоста
NEWTON	db2cli.exe	7	G91A0D3A.P8BC.060306212019
NEWTON	db2cli.exe	25	9.26.13.61.65289.060306213816
NEWTON	db2cli.exe	20	2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741

ID авт. (ID авторизации)

ID авторизации, использованный при регистрации на сервере баз данных хоста или System i. 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.

Хэндл программы

Агент, исполняемый на рабочей станции DB2 Connect. С помощью этого элемента можно связать информацию, которую собрал системный монитор баз данных, с другой диагностической информацией. 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 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 System i. 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	
NEWTON	db2cli.exe		37	2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224	
Нп/п клиента	Алиас БД клиента	Узел клиента	Выпуск клиента	Код.стр. клиента	ID программы хоста
00001	MDB	SAYYID	SQL09000	1252	G91A0D3A.P982.060306214231
Нп/п	Имя БД хоста		Выпуск хоста		
00001	MEXICO		DSN08015		

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.

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 System i.

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. 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:

- SQL07010 для Версии 7.1 продуктов DB2 Universal Database и DB2 Connect и их клиентов
- SQL08010 для Версии 8.1 продуктов DB2 Universal Database и DB2 Connect и их клиентов.
- SQL08020 для Версии 8.2 продуктов DB2 Universal Database и DB2 Connect и их клиентов.
- SQL09120 для Версии 9.1 продуктов DB2, продуктов DB2 Connect и их клиентов.

ID кодовой страницы (Код. стран.)

Идентификатор кодовой страницы узла, на котором запущена отслеживаемая программа.

Вы можете использовать эту информацию, чтобы убедиться, что поддерживается преобразование данных между кодовыми страницами прикладной программы и базы данных (или, для серверных баз данных хоста или System i, CCSID сервера баз данных хоста или System i).

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)

Фактическое имя базы данных, к которой подключена 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 System i (например, DSN для DB2 Universal Database for z/OS and OS/390, ARI для DB2 Server for VSE & VM, QSQ для DB2 для i5/OS)
- VV** Двухзначный номер версии, например, 08
- RR** Двухзначный номер выпуска, например, 01
- M** Односимвольный уровень модификации (0-9 или A-Z).

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	= 2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224
Последовательный номер	= 00001
ID авторизации	= NEWTON
Доверенный ID авторизации	=
Имя прикладной программы	= db2cli.exe
Хэндл прикладной программы	= 37
Состояние программы	= ожидание требования
Время изменения состояния	= Не собрана
Узел клиента	= SAYYID
Уровень выпуска клиента	= SQL09000
Платформа клиента	= NT
Протокол клиента	= TCP/IP
Кодовая страница клиента	= 1252
ID процесса программы клиента	= 1192
ID регистрации клиента	= ISAYYID
ID программы хоста	= G91A0D3A.P982.060306214231
Последовательный номер	= 00001
Алиас базы данных на шлюзе	= MDB
Имя базы данных DCS	= MDB
Имя базы данных хоста	= MEXICO
Уровень версии хоста	= DSN08015
CCSID хоста	= 1208

Поле состояния программы содержит одно из следующих трех значений:

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

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".

Монитор работоспособности и оповещения

Монитор работоспособности DB2 for z/OS оценивает политики обслуживания объектов на регулярной основе. Если монитор работоспособности определяет необходимость обслуживания объектов, создаются оповещения монитора работоспособности. Действия в ответ на оповещения монитора работоспособности можно просмотреть, принять для выполнения и сохранить.

Обзор монитора работоспособности DB2 for z/OS

В системах z/OS монитор работоспособности DB2 for z/OS запускается как задача для каждой подсистемы DB2 или отдельного элемента группы совместного использования данных, для которых нужно выполнять мониторинг.

Монитор работоспособности DB2 for z/OS будет выполнять оценку политик обслуживания объектов в запланированные моменты и через запланированные интервалы, заданные в политике. Политики обслуживания объектов создаются с помощью мастера по созданию правил обслуживания объектов Центра управления

DB2. При каждой оценке политики проверяется необходимость обслуживания по набору пороговых критериев, заданных в политике обслуживания объектов, то есть необходимость выполнения операций COPY, REORG, RUNSTATS, STOSPACE, ALTER TABLESPACE и ALTER INDEX, а также выявляются состояния с ограничениями, например, СЖКР, для табличных пространств, индексов и объектов групп хранения, если это применимо. Если при оценке политики обнаружены объекты в состоянии оповещения, адресатам оповещений монитора работоспособности будут направлены оповещения по электронной почте или пейджеру. Список адресатов оповещений монитора работоспособности для каждой подсистемы DB2 определяется Центром управления и управляется им.

При запуске монитора работоспособности делается снимок расписания оценки для политик, позволяющий монитору работоспособности определять, когда запускать оценку. Это снимок расписания обновляется в моменты времени, заданные при запуске монитора работоспособности, или при получении монитором команды REFRESH. Все изменения в расписании оценки для правил воспринимаются монитором работоспособности при обновлениях расписания.

Запуск и остановка монитора работоспособности производится с консоли при помощи команд системы MVS START и STOP, соответственно.

Процедура примера из каталога (DSNHMONP), запускающая монитор работоспособности DB2, и процедура примера из каталога (DSNHMONA), запускающая несколько мониторов работоспособности DB2 в системе MVS или Parallel Sysplex, помещены заданием установки DSNTIJHM в библиотеку процедур.

Производные таблицы, таблицы, наборы данных, процедуры из каталога, хранимые процедуры, пользовательские функции и таблица наборов результатов, используемые монитором производительности DB2 или перечисленными ниже связанными с ним задачами, создаются и устанавливаются заданиями установки DSNTIJCC и DSNTIJHM. DSNTIJCC и DSNTIJHM поставляются с FMID JDB771D и JDB881D.

Журнал оценки для политики

Результаты оценки для политики, запущенной монитором работоспособности DB2, заносятся в таблицу DSNACC.HM_EVAL_LOG. Запись в журнале делается в начале оценки и в ее конце. Записи журнала хранятся 7 дней, после чего удаляются из таблицы. Посмотреть все политики, для которых последняя попытка оценки завершилась неудачно, можно с помощью производной таблицы DB2 DSNACC.HM_ALERT_PO_EV, созданной для этой таблицы заданием установки DSNTIJCC.

Запуск, остановка и обновление монитора работоспособности DB2 for z/OS

В системе z/OS монитор работоспособности DB2 for z/OS запускается как задача для каждой подсистемы DB2 или отдельного элемента группы совместного использования данных, для которых нужно выполнять мониторинг.

- Чтобы запустить монитор работоспособности DB2, введите команду START системы MVS:

```
S имя_элемента,DB2SSN=ID_подсистемы,JOBNAME=ID_подсистемы_HMON,  
TRACE=трассировка,REFRESH=чч
```

Параметры TRACE и REFRESH обязательны.

имя_элемента

Задает элемент библиотеки процедур, выполняемый для запуска монитора работоспособности DB2, то есть DSNHMONP. Эта каталогизированная процедура создается заданием установки DSNTIJHM.

ID_подсистемы

Задает имя или идентификатор отслеживаемой подсистемы DB2.

трассировка

Задает флаг трассировки. Возможные значения:

- ON - Включить трассировку. Информация трассировки будет записываться в SYSOUT
- OFF - Не включать трассировку

Значение по умолчанию - OFF.

чч

Задает час (в 24-часовом формате), когда монитор работоспособности обновляет снимок расписания оценки, используемый для запуска оценки политики. Значение по умолчанию - 22.

- Чтобы запустить несколько мониторов работоспособности DB2, введите следующую команду START системы MVS:

S имя_элемента

имя_элемента

Элемент библиотеки процедур, выполняемый для запуска нескольких мониторов работоспособности DB2, то есть DSNHMONA.

Примечание: Перед запуском нескольких мониторов работоспособности DB2 одной командой START с использованием DSNHMONA набор данных HMONPARM, заданный в процедуре DSNHMONA, необходимо заполнить списком подсистем, для которых выполняется мониторинг. Эта каталогизированная процедура и набор данных создаются заданием установки DSNTIJHM.

- Чтобы обновить расписание оценки для политики, с помощью которого монитор работоспособности DB2, определяет, когда запускать оценку, введите команду MODIFY системы MVS:

F HMONID_подсистемы,APPL=REFRESH

ID_подсистемы

Имя или идентификатор подсистемы DB2, которую отслеживает обновляемый вами монитор работоспособности DB2.

- Чтобы остановить монитор работоспособности DB2, введите команду STOP системы MVS:

STOP HMONID_подсистемы или P HMONID_подсистемы

ID_подсистемы

Имя или идентификатор подсистемы DB2, которую отслеживает останавливаемый вами монитор работоспособности DB2.

Просмотр, передача на выполнение и сохранение рекомендованных действий

Чтобы просматривать, передавать на выполнение и сохранять действия, рекомендованные для объектов оповещения, которые выявлены в ходе проверки для правил, вызовите хранимую процедуру DB2 SYSPROC.DSNACCHR, создаваемую заданием установки DSNTIJJCC. DSNACCHR - это хранимая процедура, определяющая рекомендуемые действия для объектов оповещения, выявленных в ходе проверки для правил, и генерирующая задание JCL для выполнения этих действий.

Ниже на синтаксической диаграмме показан оператор SQL CALL для вызова DSNACCHR. Поскольку для DSNACCHR используется соглашение о связях GENERAL WITH NULLS, в случае передачи параметров в переменных хоста с каждой из них надо будет передать индикатор пустых значений. Перед выполнением оператора CALL необходимо инициализировать индикаторы пустых значений для входных переменных хоста.

Синтаксис

```

▶—CALL DSNACCHR (—тип-запроса—индикатор-работоспособности,—ID-политики,—рабочий-набор,—
▶—имя-набора-данных,—имя-элемента,—опция-сохранения,—флаг-трассировки,—
  [NULL] [NULL] [NULL]
▶—ID-задания,—имя-задания,—время-процедуры-jcl,—флаг-трассировки,—последний-оператор,—
▶—код-возврата,—сообщение-об-ошибке )—

```

тип-запроса

Задаёт, что нужно сделать с действиями, рекомендованными для объектов, для которых проверка для правил обнаружила состояние оповещения. Возможные значения:

- 0 - Просмотреть рекомендованные действия для объектов оповещения в виде задания JCL
- 1 - Передать на выполнение задание JCL, выполняющее эти рекомендованные действия
- 2 - Передать на выполнение задание JCL, выполняющее рекомендованные действия для объектов оповещения и поместить это задание в очередь
- 3 Сохранить рекомендованные действия для объектов оповещения как задание JCL в элементе библиотеки

тип_запроса - входной параметр типа INTEGER.

индикатор_работоспособности

Задаёт тип оповещения, который процедура DSNACCHR включает в задание JCL. Возможные значения:

- RS - Состояние с ограничениями (Restricted State)
- EX - Превышено число экстенгов (Extents Exceeded)
- RR - Требуется REORG (REORG Required)
- CR - Требуется COPY (COPY Required)
- RT - Требуется RUNSTATS (RUNSTATS Required)
- SS - Требуется STOSPACE (STOSPACE Required)

индикатор_работоспособности - входной параметр типа VARCHAR(4).

ID-политики

Задаёт политику обслуживания объекта. *ID_политики* - входной параметр типа VARCHAR(7).

рабочий-набор

Задаёт рабочий набор правил обслуживания объектов, определяющий объекты оповещения, которые DSNACCHR включает в задание JCL. Этот рабочий набор должен определяться данными правилами и типом оповещения, заданного параметрами *ID_политики* и *индикатор_работоспособности. рабочий_набор* - входной параметр типа INTEGER.

имя-набора-данных

Задаёт полное имя секционированного набора данных (PDS) или расширенного секционированного набора данных (PDSE). Это значение должно быть задано, если для *тип_запроса* задано 3. *имя_набора_данных* - входной параметр типа VARCHAR(44).

имя-элемента

Задаёт элемент секционированного набора данных (PDS) или расширенного секционированного набора данных (PDSE), заданный параметром *имя_набора_данных*, в котором будет сохранено задание JCL обслуживания объектов. Это значение необходимо задать, если *тип_запроса* - 3. *имя_элемента* входной параметр типа VARCHAR(8).

опция-сохранения

Задаёт, как сохранить задание обслуживания объектов JCL. Это значение должно быть указано, если для *типа_запроса* задано 3. Возможные значения:

- R - Замена
- A - Присоединение
- NM - Новый элемент

опция_сохранения - входной параметр типа VARCHAR(2).

флаг-трассировки

Задаёт включение или выключение трассировки. Возможные значения:

- Y - Включить трассировку
- N - Выключить трассировку

флаг_трассировки - входной параметр типа CHAR(1).

ID-задания

Если значение *тип_запроса* - 1 или 2, задаёт ID переданного на выполнение задания. *ID_задания* выходной параметр типа VARCHAR(8).

имя-задания

Если значение *тип_запроса* - 1 или 2, задаёт имя переданного на выполнение задания. *имя_задания* выходной параметр типа VARCHAR(8).

время-процедуры-jcl

Задаёт время обработки требования. *время_процедуры_jcl* - выходной параметр типа TIMESTAMP.

последний-оператор

Если DSNACCHR возвращает серьезную ошибку (код возврата 12), в этом поле выводится оператор SQL, который выполнялся на момент ошибки.
последний_оператор - выходной параметр типа VARCHAR(2500).

код-возврата

Код возврата при выполнении DSNACCHR. Возможные значения:

- 0 - DSNACCHR выполнена успешно
- 12 - DSNACCHR завершена с серьезными ошибками. В параметре *сообщение_об_ошибке* содержится сообщение с описанием ошибки. Параметр *последний_оператор* содержит оператор SQL, который выполнялся на момент ошибки.

код_возврата - выходной параметр типа INTEGER.

сообщение-об-ошибке

Если DSNACCHR возвращает серьезную ошибку (код возврата 12), в этом поле выводятся сообщения об ошибках, включая форматированную SQLCA.
сообщение_об_ошибке - выходной параметр типа VARCHAR(1331).

Если значение параметра *тип-запроса* - 0, то DSNACCHR возвращает один набор результатов. Этот набор результатов содержит задание JCL, сгенерированное DSNACCHR. Таблица набора результатов DSNACCHR создается заданием установки DSNTIJCC. В Табл. 12 показан формат этого набора результатов.

Таблица 12. Формат набора результатов DSNACCHR

Имя столбца	Тип данных	Описание
JCLSEQNO	INTEGER	Последовательный номер строки таблицы (1,...,n)
JCLSTMT	VARCHAR(80)	Задаёт оператор JCL

Просмотр сводных данных оповещения работоспособности

Функция HEALTH_OVERVIEW возвращает информацию из набора данных VSAM KSDS сводки оповещений работоспособности в виде таблицы DB2. Этот набор данных создается заданием установки DSNTIJHM.

Набор данных сводки оповещений работоспособности содержит информацию о состоянии монитора работоспособности DB2 и статистику сводки оповещений для каждой подсистемы DB2, отслеживаемой сейчас или ранее этим монитором работоспособности в данной системе MVS или Parallel Sysplex. Для каждой подсистемы DB2 клиенту возвращается строка с этой информацией и рекомендацией оповещения.

Результат выполнения этой функции - таблица DB2 со столбцами:

IP-адрес

Адрес IP сервера DB2. Это столбец типа VARCHAR(40).

ID_подсистемы_DB2

Идентификатор подсистемы DB2. Это столбец типа VARCHAR(4).

индикатор_работоспособности

Тип оповещения. Возможные значения:

- RS - Состояние с ограничениями (Restricted State)
- EX - Превышено число экстенгов (Extents Exceeded)
- RR - Требуется REORG (REORG Required)
- CR - Требуется COPY (COPY Required)
- RT - Требуется RUNSTATS (RUNSTATS Required)
- SS - Требуется STOSPACE (STOSPACE Required)
- PO - Оценка политики завершилась неудачно (Failed Policy Evaluation)
- HM - Состояние монитора работоспособности (Health Monitor State)

Индикатор_работоспособности - это столбец типа VARCHAR(4).

Имя_хоста

Полное имя домена сервера DB2. Это столбец типа VARCHAR(255).

Сводка_статистики

Состояние монитора работоспособности DB2, если значение *индикатор_работоспособности* - 'HM'. Возможные значения:

- 0 Монитор работоспособности не запущен
- 1 Монитор работоспособности запущен
- -1 Состояние монитора работоспособности неизвестно

В остальных случаях это число объектов оповещения с типом оповещения, заданным в столбце *индикатор_работоспособности*. Это столбец типа INTEGER.

состояние_оповещения

Состояние оповещения, заданное в *индикатор_работоспособности*. Возможные значения:

- 5 - Тревога
- 4 - Внимание
- 3 - Предупреждение
- 0 - Нормальное

Если значение в столбце *индикатор_работоспособности* - 'HM', значение *состояние_оповещения* - всегда 0. Это столбец типа INTEGER.

Имя внешней программы для этой функции - HEALTH_OVERVIEW, а уникальное имя - DSNACC.DSNACCНО. Эта функция создается заданием установки DSNTIJCС.

Пример: Найти общее число объектов оповещения, требующих выполнения COPY, для подсистемы DB2 'ABCD':

```
SELECT SUMMARYSTATS FROM TABLE (DSNACC.HEALTH_OVERVIEW()) AS T
WHERE DB2SSID = 'ABCD'
AND HEALTHIND = 'CR';
```

Просмотр объектов оповещения работоспособности

Объекты оповещения, выявленные во время последней успешной проверки для правил хранятся в таблицах репозитория для этих объектов, которые зависят от типа объекта.

Существуют следующие объекты оповещения:

- DSNACC.HM_MAINT_TS для табличных пространств
- DSNACC.HM_MAINT_IX для индексов
- DSNACC.HM_MAINT_SG для групп хранения

DB2 создает для этих таблиц репозитория объектов оповещения несколько производных таблиц. Эти производные таблицы, как и таблицы репозитория объектов оповещения создаются заданием установки DSNTIJCC. В Табл. 13 перечислены таблицы, для которых определены производные таблицы, и даются описания этих производных таблиц. Спецификатор имен всех таблиц и производных таблиц - DSNACC.

Таблица 13. Производные таблицы для объектов оповещений работоспособности

Имя производной таблицы	По таблице	Описание производной таблицы
HM_ALERT_TS_RS	HM_MAINT_TS	Содержит все табличные пространства в состоянии с ограничениями
HM_ALERT_TS_EX	HM_MAINT_TS	Содержит все табличные пространства, экстенды которых превышают заданный пользователем предел
HM_ALERT_TS_RR	HM_MAINT_TS	Содержит все табличные пространства, для которых требуется выполнение утилиты REORG
HM_ALERT_TS_CR	HM_MAINT_TS	Содержит все табличные пространства, для которых требуется выполнение утилиты COPY
HM_ALERT_TS_RT	HM_MAINT_TS	Содержит все табличные пространства, для которых требуется выполнение утилиты RUNSTATS
HM_ALERT_IX_RS	HM_MAINT_IX	Содержит все индексы в состоянии с ограничениями
HM_ALERT_IX_EX	HM_MAINT_IX	Содержит все индексы, экстенды которых превышают заданный пользователем предел
HM_ALERT_IX_RR	HM_MAINT_IX	Содержит все индексы, для которых требуется выполнение утилиты REORG
HM_ALERT_IX_CR	HM_MAINT_IX	Содержит все индексы, для которых требуется выполнение утилиты COPY
HM_ALERT_IX_RT	HM_MAINT_IX	Содержит все индексы, для которых требуется выполнение утилиты RUNSTATS
HM_ALERT_SG_SS	HM_MAINT_SG	Содержит все группы хранения, для которых требуется выполнение утилиты STOSPACE

Часть 3. Высокая доступность и DB2 Connect

Использование высокой доступности в среды, где применяется DB2 Connect, отличается некоторыми особенностями. В случае недоступности по какой-то причине сервера баз данных в сети для использования становится критичной возможность перенаправления рабочей станции клиента на альтернативный сервер баз данных.

Глава 11. Высокая доступность и сбалансированная нагрузка в соединениях с базами данных хоста

На современном рынке информационных технологий преимущество получают продукты и системы, обеспечивающие круглосуточную доступность данных. Применение таких систем позволяет опережать конкурентов. Многим современным прикладным программам в области Web-коммуникаций, электронного бизнеса и электронных таблиц требуется доступ к данным предприятия. Возникает необходимость в надежном, быстродействующем и защищенном соединении с базами данных хоста и System i. Такое соединение должно быть доступно постоянно, при этом оно должно обеспечивать заданный уровень обслуживания в условиях критической нагрузки. Каким образом можно создать такое соединение?

Сценарий обеспечения высокой доступности

Компания располагает несколькими рабочими станциями и серверами прикладных программ, работающими под управлением Windows, Linux и UNIX. Этим компьютерам необходим доступ к данным, хранящимся в различных базах данных хоста и System i. Прикладным программам этих компьютеров требуются быстродействующие и надежные соединения с базами данных. Все компьютеры объединены в сеть Ethernet с протоколом TCP/IP.

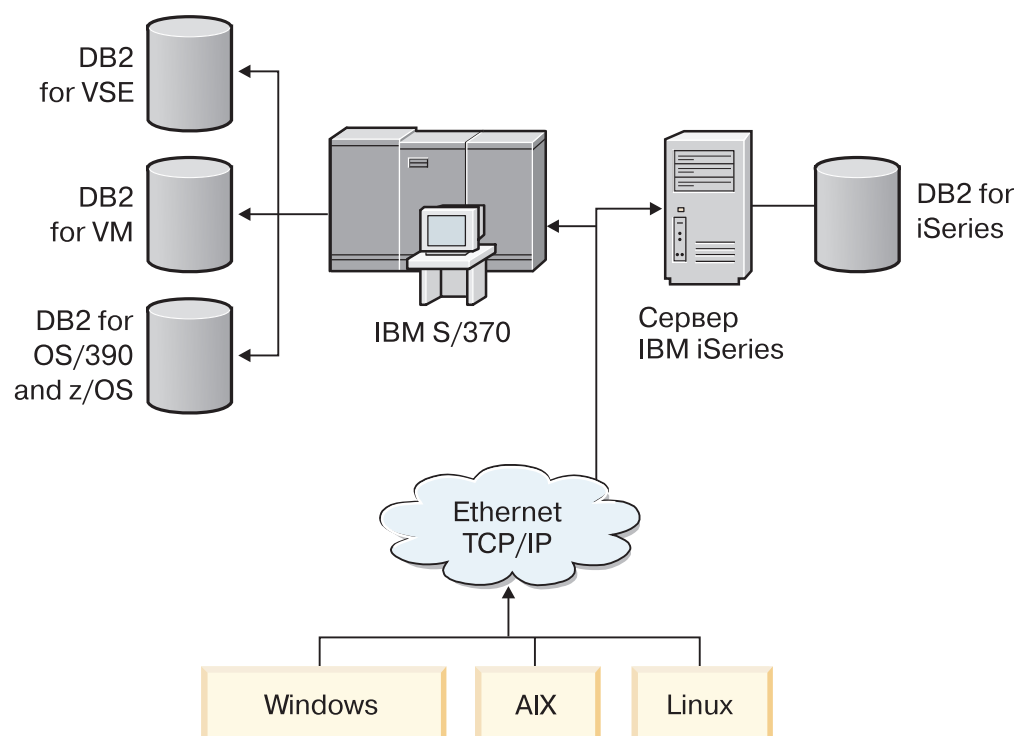


Рисунок 11. Сетевой сценарий примера

Чтобы рабочие станции и серверы прикладных программ могли обращаться к базам данных хоста и System i, необходимо установить промежуточный узел - компонент связи. Этот компонент должен обеспечивать высокодоступное, устойчивое и

быстродействующее соединение с базами данных хоста и System i. Кроме того, он должен обладать свойством масштабируемости, так как со временем число соединений будет возрастать.

Подробности решения с использованием DB2 Connect и возможности автоматического перенаправления клиента смотрите в дополнительных ссылках в этой теме.

Глава 12. Описание и установка автоматического перенаправления клиента (DB2 Connect)

Основная цель функции автоматического перенаправления клиента - дать прикладной программе клиента базы данных DB2 восстановить потерянную связь, чтобы эта прикладная программа могла продолжать работу с минимальным перерывом. Как подсказывает название, перенаправление - ключевой момент поддержки длительной непрерывной работы. Но перенаправление возможно только в том случае, когда есть альтернативный сервер, идентифицированный для соединения клиента. В другой среде высокой доступности (не DB2 Connect) синхронизация рабочей базы данных обычно производится с исходным сервером DB2 и альтернативным сервером DB2 с использованием такого средства, как, например, HACMP (High availability cluster multiprocessor - кластерный мультипроцессор с высокой доступностью) или HADR (High availability disaster recovery - высокая доступность - восстановление после аварий).

Однако в случае сервера DB2 Connect, когда требования для синхронизации локальных баз данных отсутствуют, необходимо только убедиться, что и у исходного, и у альтернативного сервера DB2 Connect база данных хоста или System i каталогизированы способом, который доступен при использовании идентичного алиаса базы данных.

Примечание: В среде DB2 Connect конфигурируется перенаправление клиента между клиентом и сервером DB2 Connect. Не конфигурируется и не поддерживается перенаправление клиента между DB2 Connect и сервером базы данных хоста или System i.

Чтобы у клиента DB2 была возможность восстанавливаться после потери связи с сервером DB2 Connect, положение альтернативного сервера DB2 Connect должно быть указано до того, как такая потеря связи произойдет. Команда UPDATE ALTERNATE SERVER FOR DATABASE служит для задания положения альтернативного сервера DB2 Connect для конкретной базы данных или хоста System i. В этой команде указываются имя и номер порта альтернативного хоста. Этот адрес хранится в файле системного каталога баз данных на сервере DB2 Connect. Чтобы положение альтернативного сервера DB2 Connect использовалось всеми клиентами, его надо задать на сервере DB2 Connect. Альтернативный сервер игнорируется, если он задан в экземпляре клиента.

Пусть, например, база данных хоста или System i каталогизирована с использованием алиаса базы данных db1 на сервере DB2 Connect S1 (с именем хоста db2conn1 и номером порта 122). Администратор базы данных хочет задать альтернативный сервер DB2 Connect S2 с именем хоста db2conn2 и номером порта 123. Администратор базы данных должен ввести следующую команду на сервере DB2 Connect S1:

```
db2 update alternate server for database db1 using hostname db2conn2 port 123
```

Когда вы указали альтернативное положение сервера DB2 Connect для алиаса базы данных db1 на сервере DB2 Connect S1, информация об этом положении возвращается клиенту как часть процесса установления соединения. Если по какой-либо причине связь между клиентом и сервером DB2 Connect S1 будет потеряна (обычно из-за ошибки соединения, например, с кодом SQL -30081 или -1224), клиент DB2 попытается пересоединиться с db1 через исходный или альтернативный сервер DB2

Connect (S1 или S2 соответственно), чередуя попытки связаться то с одним, то с другим сервером. Сначала такие попытки повторяются часто, но с каждой попыткой интервал между ними постепенно увеличивается.

В случае успешного соединения возвращается код SQL -30108, указывающий на успешное восстановление соединения с базой данных после ошибки связи. Возвращаются имя хоста/IP-адрес и имя службы/номер порта. При потере связи с исходным сервером клиент возвращает прикладной программе ошибку лишь в том случае, если восстановление связи с клиентом невозможно ни для исходного, ни для альтернативного сервера.

Необходимо также отметить следующие особенности перенаправления клиента в среде серверов DB2 Connect:

- При использовании сервера DB2 Connect для обеспечения доступа к базе данных хоста или System i и с удаленных, и с локальных клиентов может возникнуть путаница из-за информации соединения с альтернативным сервером, хранящейся в записи системного каталога баз данных. Для устранения этой путаницы попробуйте внести в системный каталог базы данных две записи, представляющие одну и ту же базу данных хоста или System i. Добавьте в каталог одну запись для удаленных клиентов, а другую - для локальных.
- Любая информация SYSPLEX, возвращаемая с сервера назначения DB2 for z/OS хранится только в кэше на сервере DB2 Connect. При завершении процесса DB2 на сервере DB2 Connect эта кэшированная информация теряется, даже если в дальнейшем восстанавливается соединение с этим же сервером DB2 Connect.

Глава 13. Конфигурирование автоматического перенаправления клиента для технологии распределителя соединений клиентов

Технологии распределителей соединений или диспетчеров связи, такие как WebSphere EdgeServer, распределяют требования восстановления соединений программ клиентов в определенный набор систем в случае отказа первичного сервера баз данных. При использовании технологии диспетчера связи с автоматическим перенаправлением клиентов DB2 сам диспетчер связи нужно определить как альтернативный сервер для автоматического перенаправлением клиентов DB2.

Возможно, вы используете технологию диспетчера связи примерно в такой среде:

Клиент → технология диспетчера связи → (сервер DB2 Connect 1 или сервер DB2 Connect 2) → DB2 z/OS

где:

- Имя хоста TCP/IP для компонента технологии диспетчера связи - **DThostname**.
- Имя хоста TCP/IP для сервера DB2 Connect 1 - **GWYhostname1**
- Имя хоста TCP/IP для сервера DB2 Connect 2 - **GWYhostname2**
- Имя хоста TCP/IP сервера DB2 z/OS - **zOShostname**

Клиент каталогизируется с использованием **DThostname**, чтобы технологию диспетчера связи можно было применить для обращения к одному из двух серверов DB2 Connect. Промежуточный диспетчер связи решает, какой сервер использовать - **GWYhostname1** или **GWYhostname2**. После принятия решения у клиента будет прямое соединение гнезда с одним из этих двух шлюзов DB2 Connect. После установления соединения гнезда с выбранным сервером DB2 Connect начинает действовать обычная связь клиент - сервер DB2 Connect - DB2 z/OS.

Допустим, диспетчер связи выбрал сервер **GWYhostname2**. В этом случае среда будет следующей:

Клиент → сервер DB2 Connect 2 → DB2 z/OS

При сбоях связи диспетчер связи не восстанавливает соединение. Если вы хотите включить функцию автоматического перенаправления клиента для базы данных в такой среде, альтернативный сервер для связанной с ним базы данных или баз данных на сервере DB2 Connect (сервере DB2 Connect 1 или сервере DB2 Connect 2) следует сконфигурировать как диспетчер связи (**DThostname**). После этого, если сервер DB2 Connect 1 по какой-либо причине будет неработоспособен, автоматическое перенаправление клиента переключается, и предпринимается попытка повторного соединения клиента с диспетчером связи в качестве главного сервера и в качестве альтернативного сервера. Эта опция позволяет объединить функции автоматического перенаправления клиента DB2 с функциями диспетчера связи и поддерживать их. Если альтернативный сервер сконфигурировать с хостом, отличным от хоста диспетчера связи, клиенты все же смогут использовать функцию автоматического перенаправления клиента. Однако клиенты будут устанавливать соединения напрямую с заданным альтернативным сервером и не будут использовать технологию диспетчера связи и ее преимущества.

Функция автоматического перенаправления клиента перехватывает следующие коды SQL:

- sqlcode -20157
- sqlcode -1768 (код причины = 7)

Примечание: Если значение параметра конфигурации операционной системы "TCP Keepalive" задано слишком высоким, функция перенаправления клиента может быть не проинформирована об ошибках гнезд вовремя. (Обратите внимание на то, что имя этого параметра конфигурации может быть разным в зависимости от платформы).

Часть 4. Настройка и DB2 Connect

Среде баз данных, где для передачи требований и ответов баз данных между рабочими станциями клиентов и серверами баз данных используется DB2 Connect, присущи свои особенности решения проблем производительности. Для улучшения или поддержания производительности в этой среде предусмотрено несколько способов.

Глава 14. Производительность DB2 Connect

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, выполняемых 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 System i 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.

Потоки данных

На рис. 12 на стр. 96 показан путь передачи данных между сервером баз данных хоста или System i и рабочей станцией через DB2 Connect.

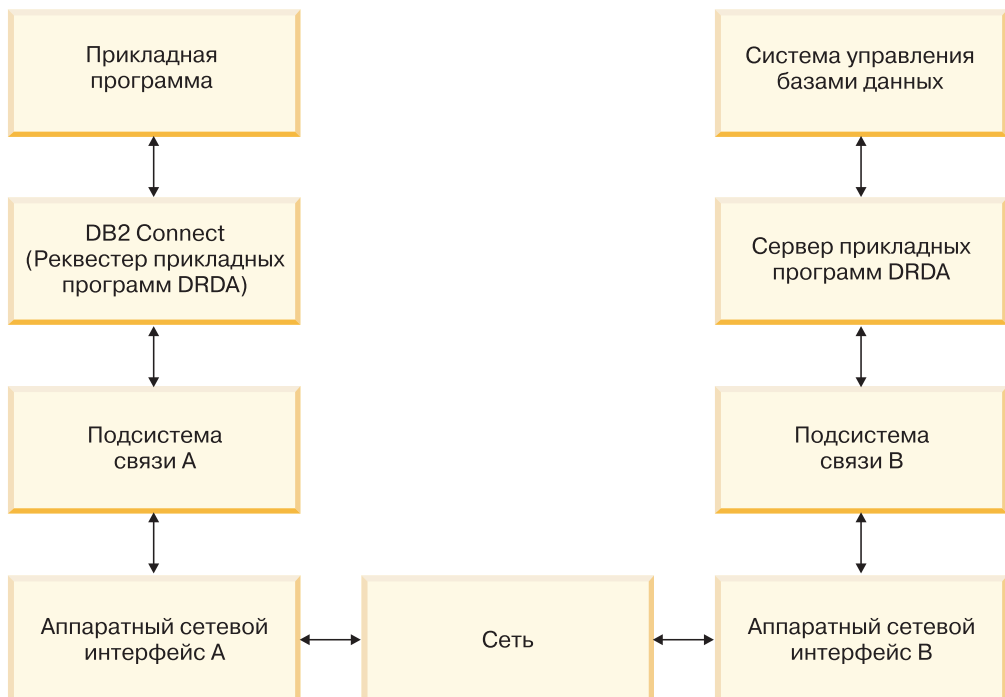


Рисунок 12. Потоки данных в DB2 Connect

- База данных хоста или System i и часть В подсистемы связи обычно работают на одной системе. Эта система состоит из одного или нескольких процессоров, памяти подсистемы ввода-вывода, запоминающего устройства прямого доступа и операционной системы. Эти компоненты могут одновременно использоваться другими программами, поэтому 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 сервера System i. 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.
- Продукт DB2 Connect и подсистема связи А обычно расположены на одной системе. В данном разделе 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.

Тестирование производительности

Тестирование производительности позволяет 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 только 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.

Performance Tools

В следующих таблицах перечислены некоторые программы, которые помогают измерить производительность системы. Они сами используют системные ресурсы, поэтому, возможно, лучше не держать их постоянно активными.

Таблица 14. Инструменты измерения производительности использования процессора и памяти

Система	Программа	Описание
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	
Windows	Монитор производительности Microsoft	

Таблица 15. Инструменты измерения производительности операций базы данных

Система	Программа	Описание
Все	Монитор баз данных	Определяет, является ли база данных источником затруднений.

Таблица 15. Инструменты измерения производительности операций базы данных (продолжение)

Система	Программа	Описание
OS/390 или zSeries	DB2PM (IBM), OMEGAMON/DB2 (Candle), TMON (Landmark), INSIGHT (Goal Systems) и DB2AM (BMC)	
Windows	Монитор производительности Microsoft	

Таблица 16. Инструменты измерения производительности операций в сети

Система	Программа	Описание
AIX	netpmn	Дает статистику низкого уровня, включая статистику TCP/IP, такую как число пакетов или кадров, получаемых за секунду.
Сетевые контроллеры типа 3745	Монитор производительности NetView	Дает сведения об управлении связью и VTAM.
Linux и UNIX	netstat	Обрабатывает трафик TCP/IP.

Глава 15. Оптимизация доступа ODBC

База данных DB2 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, воспользуйтесь Ассистент конфигурирования (CA) DB2.

Для активации оптимизированной связи ODBC выполните следующие действия:

- При определении нового соединения:
 1. Запустите CA DB2.
 2. Откройте меню Выбранные и выберите пункт Добавить базу данных при помощи мастера...
 3. Выполняйте указания страниц мастера, пока не перейдете на страницу **Источник данных**.
 4. Включите переключатель **Зарегистрировать базу данных для CLI/ODBC**.
 5. Задайте способ регистрации прикладных программ CLI/ODBC, обращающихся к этой базе данных:
 - **Как системный источник данных** - база данных будет доступна для всех пользователей в системе.
 - **Как пользовательский источник данных** - вы будете единственным пользователем, который сможет обращаться к этой базе данных.
 - **Как файловый источник данных** - будет создан файл, содержащий информацию об источнике. Если установлено соединение TCP/IP, этот файл источника данных можно совместно использовать с другими рабочими станциями. Иначе этот файл можно будет использовать только на данном компьютере
 6. Введите **Имя источника данных**.
 7. (Необязательно) В списке **Оптимизировать для программы** выберите прикладную программу, чтобы оптимизировать параметры источника данных для конкретной программы.
 8. Нажмите кнопку **ОК** и завершите работу с CA.
- При изменении существующего нового соединения:
 1. Запустите CA DB2.
 2. Щелкните дважды мышью по алиасу базы данных, работу с которой нужно оптимизировать.
 3. Выберите **Источник данных**.
 4. Включите переключатель **Зарегистрировать базу данных для CLI/ODBC**.
 5. Задайте способ регистрации прикладных программ CLI/ODBC, обращающихся к этой базе данных:
 - **Как системный источник данных** - база данных будет доступна для всех пользователей в системе.
 - **Как пользовательский источник данных** - вы будете единственным пользователем, который сможет обращаться к этой базе данных.
 - **Как файловый источник данных** - будет создан файл, содержащий информацию об источнике. Если установлено соединение TCP/IP, этот файл источника данных можно совместно использовать с другими рабочими станциями. Иначе этот файл можно будет использовать только на данном компьютере

6. Введите **Имя источника данных**.
7. (Необязательно) В списке **Оптимизировать для программы** выберите прикладную программу, чтобы оптимизировать параметры источника данных для конкретной программы.
8. Нажмите кнопку **ОК** и завершите работу с СА.

Глава 16. Разработка программ

Создавая программу, можно повысить ее производительность несколькими способами.

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
- Compound SQL
- Connect
- Prepare
- хоста
- Describe
- Rollback
- Disconnect
- Set connection
- execute immediate

Хранимые процедуры помогают уменьшить сетевой трафик за счет размещения текста программы на сервере. Принятие можно производить автоматически при выходе из процедуры. Кроме того, можно возвращать наборы результатов, что сводит к минимуму текст программы на клиенте.

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, в операторах UPDATE и DELETE можно использовать ключевые слова WHERE, IN и BETWEEN.

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.

Блокировка 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 System i, так и сервера 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.

Внутри DB2 Connect можно управлять объемом данных, передаваемых в каждом блоке.

Блокирование включается опцией BLOCKING в командах prer и bind.

Блокирование используется, если:

- Указатель является указателем только для чтения или
- Указатель неоднозначен, а в команде prer или bind задано блокирование.

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.

Для оператора SELECT без возможности изменения задайте 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 приходится еще раз послать DESCRIBE с достаточными SQLDA, чтобы заново сохранить результат. Это увеличивает сетевой трафик.

Не используйте последовательности PREPARE и DESCRIBE. Оператор PREPARE INTO более эффективен с точки зрения производительности.

- 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.

Другие особенности SQL

Процессор командной строки (CLP) 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.

Глава 17. Управление соединениями

При управлении соединениями используются операции двух типов: пула соединений и концентратора соединений. Пул соединений сокращает затраты на соединения с базами данных и управляет их числом. Концентратор соединений увеличивает масштабируемость в рабочей среде посредством оптимизации ресурсов, используемых серверами баз данных хоста. Здесь обсуждаются операции обоих типов.

Пулы соединений

Часто продукты сервера DB2 Connect, такие как 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 продуктах сервера DB2 Connect для сокращения этих затрат ресурсов поддерживается пул соединений, в котором сохраняются уже открытые соединения с базой данных.

Большинство прикладных программ Web совершают большое число коротких транзакций. Обычно такая транзакция рассматривается как часть своего соединения. Другими словами, при выполнении этой транзакции устанавливается соединение с базой данных, а после выполнения нескольких операторов SQL это соединение прерывается. Такой процесс занимает достаточно много времени. При этом создается агент DB2 Connect, устанавливается сетевое соединение между этим агентом и сервером DB2, а затем создается поток DB2 на сервере. При установлении длительных соединений такой подход оправдан, так как затраченное время компенсируется большим числом транзакций, осуществляемых по этому соединению, однако для web-транзакций время, затраченное на установление соединения, может даже превосходить время самой транзакции.

Пулы соединений позволяют повторно использовать уже установленные соединения для последующих транзакций. При запуске экземпляра DB2 Connect автоматически создается пул координирующих агентов. При поступлении запроса на соединение, этому запросу присваивается свободный агент. Агент установит соединение с сервером DB2, после чего в DB2 будет создан поток. При поступлении от прикладной программы запроса на отключение, агент не передаст этот запрос серверу DB2. Вместо этого агент будет возвращен в пул. Агент в пуле по-прежнему владеет соединением с сервером DB2 и соответствующим потоком DB2. Если будет получен запрос на соединение от другой прикладной программы, агент будет присвоен этой программе. Для обеспечения защиты потоку DB2 передается информация об идентификации пользователя, которая затем применяется для аутентификации.

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.

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.

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. Другой вариант - формировать пул бездействующих агентов 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 продукт сервера DB2 Connect - для таких прикладных программ DB2 Connect может выполняться внутри процесса прикладной программы (поведение по умолчанию) или же эти прикладные программы могут использовать соединение с продуктом сервера DB2 Connect, а для соединения с хостом - агент. Чтобы прикладные программы использовали пул соединений, для соединений с хостом должны использоваться агенты продукта сервера DB2 Connect; в этом случае для переменной DB2CONNECT_IN_APP_PROCESS нужно задать значение NO.

Пулы соединений DB2 Connect и пулы соединений сервера прикладных программ

Пулы соединений обязательно должны применяться во всех прикладных программах Web, выполняющих большой объем транзакций. Большинство серверов прикладных программ Web содержат свои инструменты создания пулов. Например, пулы соединений поддерживают Microsoft MTS (COM+) и IBM WebSphere.

Механизмы создания пулов соединений, применяемые этими серверами, существенно отличаются от механизмов DB2 Connect. Поскольку пулы соединений прикладных программ обслуживают только свои транзакции, они предполагают, что ID пользователя, пароль, уровень изоляции и другие подобные параметры совпадают у всех соединений. И, что более важно, серверы прикладных программ хранят в пуле только соединения, инициированные только одним процессом. Это означает, что соединения от других компьютеров, пользователей или прикладных программ не будут занесены в пул. Поэтому технологии работы с пулами, эффективные для соединений одного экземпляра прикладной программы, совершенно неприменимы при работе с несколькими пользователями, серверами и т.п.

Пулы соединений, создаваемые серверами DB2 Connect, не зависят от конкретных прикладных программ, компьютеров и пользователей. Одно и то же соединение может применяться различными клиентами, серверами прикладных программ и пользователями, что позволяет существенно повысить эффективность использования ресурсов пула.

Какой способ работы с пулами предпочесть? Оба. В общем случае, рекомендуется применять как пулы соединений DB2 Connect, так и пулы серверов прикладных программ, поскольку они совершенно независимы. Даже если на сервере прикладных программ включены пулы соединений, пулы DB2 Connect позволяют обеспечить эффективное распределение ресурсов между несколькими серверами или клиентами, использующими DB2 Connect.

Концентратор соединений

Концентратор соединений сокращает объем ресурсов, требуемый серверам баз данных DB2 для OS/390 и z/OS для поддержки большого числа рабочих станций. Эта функция может существенно повысить масштабируемость DB2 для OS/390 и z/OS, а также DB2 Connect, обеспечивая при этом бесперебойную работу с балансировкой уровня загрузки транзакциями в средах обмена данными DB2 для OS/390 и z/OS.

Концентратор соединений позволяет прикладным программам поддерживать связь без потребления ресурсов сервера хоста DB2. В результате для работы нескольких тысяч активных прикладных программ потребуется всего несколько активных потоков на сервере DB2.

Концентратор соединений DB2 Connect позволяет продуктам сервера DB2 Connect, таким как DB2 Connect Enterprise Edition, поддерживать тысячи одновременных коммерческих транзакций, значительно сокращая затраты ресурсов на серверах баз данных хоста S/390 или System i. Потоки данных для всех прикладных программ передаются через существенно меньшее число соединений с сервером баз данных хоста S/390 или System i. 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.

В концентраторе соединений реализована концепция агента, который разбит на две части:

- Логический агент, представляющий соединение прикладной программы.
- Координирующий агент, управляющего соединениями и потоками DB2 и выполняющего требования прикладных программ.

Когда новая прикладная программа пытается соединиться с хостом, ей назначается логический агент. Чтобы передать операторы SQL базе данных, нужен

координирующий агент; он назначается сразу после начала новой транзакции. Ключевая особенность этой архитектуры в том, что координирующий агент:

- Отделен от логического агента
- Возвращается в пул при завершении транзакции (принятии или откате)

Другая важная особенность - способ назначения координирующего агента новым транзакциям в среде совместного использования данных. В DB2 Connect реализован сложный алгоритм, использующий информацию менеджера рабочей нагрузки (WLM) OS/390 и z/OS. Эта информация используется для распределения рабочей нагрузки между участниками группы совместного использования данных на основе критериев, заданных в WLM. WLM доступна информация не только о загрузке каждого участника группы, но и о его доступности. Это позволяет DB2 Connect оперативно перемещать нагрузку с неработающих или перегруженных систем на работающие и более свободные. Концентратор соединений DB2 Connect активируется в том случае, если максимальное число логических агентов (*max_connections*) превышает число координирующих агентов (*max_coordagents*).

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.

В противоположность этому, 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.

В предыдущих версиях DB2 Connect каждая активная прикладная программа использовала отдельную управляемую единицу ядра (EDU), которая управляла соединением с базой данных и всеми требованиями прикладной программы. Эту EDU 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 своя EDU. Но концентратор соединений позволяет использовать одну EDU для многих соединений. Таким образом, используемое число блоков EDU меньше или равно числу 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 конкретную 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 *диспетчером*. Существующие на некоторых платформах ограничения на число открытых файлов могут привести к тому, что будет запущено несколько экземпляров диспетчера.

Ограничения на использование концентратора соединений

Есть ряд важных ограничений на использование концентратора DB2 Connect. Внимательно изучите следующую ниже информацию, прежде чем пытаться использовать концентратор соединений в вашей системе.

Общие ограничения:

- 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, именованные конвейеры), но для таких соединений нельзя использовать возможности концентрации XA.
- 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 экземпляр сервера DB2 Connect.
- Выигрыш от использования концентратора получается только для прикладных программ, закрывающих ресурсы WITH HOLD (такие как 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.
- Если в программе объявляются глобальные временные таблицы, их нужно явно закрыть в конце транзакции или ветвления. Неудачное закрытие этих таблиц приведет к выключению концентрации соединений, хотя работа прикладной программы не прервется.
- 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.
- Концентратор поддерживает динамический SQL только для интерфейса уровня вызовов (CLI). Прикладные программы CLI также не должны использовать KEEP DYNAMIC, поскольку концентратор зависит от повторной подготовки операторов на границе каждой транзакции.
- 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.

При работе с DB2 Версии 9 или Версии 8 FixPak 13 (или новее) для включения поддержки концентратора DB2 Connect требуется System i Версия 5 Выпуск 4 (PTF SI23726). Иначе поддерживаются только возможности концентратора соединений, связанные с XA.

Активация концентратора соединений

Параметр конфигурации менеджера баз данных *max_coordagents* задает максимальное число логических агентов. Чтобы активировать использование концентратора, задайте для параметра *max_connections* значение, большее его значения по умолчанию. Значение по умолчанию параметра *max_connections* равно значению параметра *max_coordagents*. Поскольку для каждой прикладной программы используется один логический агент, параметр *max_connections* определяет число прикладных программ, которые могут соединиться с экземпляром базы данных, в то время как параметр *max_coordagents* определяет максимальное число одновременно активных входящих соединений. Для параметра *max_connections* можно задать значение в диапазоне от значения *max_coordagents* до 64000. Число логических агентов по умолчанию равно числу *max_coordagents*.

Параметры конфигурации *max_connections* и *max_coordagents* можно настраивать автоматически. Если параметр *max_connections* задается автоматически, число соединений может вырасти выше сконфигурированного базового значения. Если параметры *max_connections* и *max_coordagents* задаются автоматически, *max_connections* может стать больше базового значения, и *max_coordagents* автоматически увеличивается для сохранения коэффициента концентрации - отношения числа соединений к числу агентов координатора.

Некоторые из параметров конфигурации используются для конфигурирования агентов. Это следующие параметры:

max_coordagents

Максимальное число активных координирующих агентов.

num_poolagents

Размер пула агентов. В этот пул входят неактивные агенты и бездействующие агенты. Для повышения производительности в параметре *num_poolagents* нужно задать значение, равное среднему числу клиентов.

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 z/OS, а также DB2 для System i. 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.

Пример поддержки транзакций XA

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_CONNECTIONS = 4000
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. Предположим, что в описанной выше ситуации вместе со шлюзом DB2 Connect и базой данных zSeries или System i используется монитор транзакций. Когда прикладная программа запрашивает соединение, концентратор использует для обслуживания этого запроса неактивный агент или создает новый рабочий агент. 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 этот 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).

Пулы соединений и концентратор соединений

Хотя пулы и концентраторы соединений во многом похожи, они по-разному реализованы и предназначены для различных целей. Пулы соединений позволяют сократить число соединений с базой данных. Концентратор соединений повышает масштабируемость сервера DB2 для OS/390 и z/OS, а также DB2 Connect путем оптимизации использования ресурсов хоста.

При использовании пулов соединений, соединение становится доступным только после того, как владеющее им приложение отправит запрос на прерывание соединения. Во многих двухуровневых приложениях типа клиент-сервер соединение не освобождается в течение всего рабочего дня. Аналогично, большинство серверов приложений в многоуровневой среде устанавливают соединение при запуске сервера, и не освобождают его до полной остановки сервера.

В таких случаях пулы соединений не обеспечат никаких преимуществ. Однако, в средах web и клиент-сервер, где частота соединений и разъединений больше, пулы соединений дают выигрыш по производительности. Концентратор соединений захватывает ресурсы базы данных хоста только на время транзакции SQL, поддерживая пользовательские приложения активными. Это позволяет создавать конфигурации, где число потоков DB2 и потребляемых ими ресурсов меньше, чем если каждое соединение приложения использует свой собственный поток.

Концентратор соединений позволяет обеспечить бесперебойную работу и балансировку ресурсов путем перемещения рабочей нагрузки для каждой новой транзакции. С другой стороны, пулы соединений обеспечивают очень ограниченную балансировку и только во время соединения.

Таким образом, хотя пулы и концентраторы соединений предназначены для разных целей, их рекомендуется использовать совместно.

Для WebSphere MQ Transaction Manager и DB2 for OS/390 требуется концентратор соединений

При выполнении прикладных программ в среде IBM WebSphere MQ (прежнее название - IBM MQSeries) WebSphere MQ может действовать как XA-совместимый менеджер транзакций, координирующий любые распределенные транзакции с двухфазным принятием. Когда WebSphere MQ действует как менеджер транзакций и источники данных - продукты семейства DB2, есть несколько требований к конфигурации.

Большинство требований к конфигурации в такой среде менеджера транзакций уже описаны в других разделах. Например, на клиенте выполнения DB2 надо задать для параметра конфигурации TP_MON_NAME значение "MQ".

Однако есть еще одно требование к конфигурации. Оно специфично для DB2 Connect при соединении с источниками данных - серверами DB2 for OS/390: когда WebSphere MQ используется для координирования распределенных транзакций с участием серверов DB2 for z/OS и DB2 for i5/OS, на шлюзе должна быть включена возможность концентратора соединений DB2 Connect. Концентратор соединений включается, когда значение параметра конфигурации **max_connections** больше значения параметра конфигурации **max_coordagents**.

Если вы не включите концентратор соединений, поведение транзакций может оказаться неожиданным.

Глава 18. Поддержка Sysplex в DB2 Connect

Sysplex - это набор серверов zSeries, совместно использующих программное и аппаратное обеспечение для выполнения работы. Sysplex координирует совместную работу, позволяя использовать большее число одновременно работающих процессоров, благодаря чему может увеличиться и объем выполняемой работы. Помимо роста возможностей обработки, Sysplex может обеспечить гибкость при совмещении различных уровней программного и аппаратного обеспечения, а также при динамическом добавлении систем.

Sysplex позволяет DB2 Connect беспрепятственно передавать входящее соединение с одного удаленного сервера базы данных на назначенный резервный сервер в случае отказа первого сервера. Поддержка DB2 Connect для Sysplex включена по умолчанию. Поддержку Sysplex для базы данных хоста можно отключить, удалив параметр SYSPLEX из записи каталога DCS, однако саму запись DCS удалять не надо, даже если в ней нет никаких параметров.

При автоматическом перенаправлении клиента для соединения со включенным Sysplex в случае ошибки связи поведение по умолчанию - повторить попытку соединения. Однако операторы SET не возвращаются при включенном перенаправлении клиента в DB2 для z/OS. Чтобы обойти это ограничение, прикладные программы должны сами восстанавливать свою среду выполнения.

Можно точно сконфигурировать поведение при повторной попытке с помощью переменных реестра DB2_MAX_CLIENT_CONNRETRIES и DB2_CONNRETRIES_INTERVAL, в том числе задать отключение.

Особенности применения SYSPLEX для OS/390 и zSeries

DB2 Connect может обеспечивать балансировку нагрузки и защиту от сбоев, если направляет соединения к нескольким Sysplex. При соединении с сервером баз данных DB2 для OS/390 и z/OS в среде совместного использования данных DB2 Connect будет распределять рабочую нагрузку по различным подсистемам DB2, входящим в группу совместного использования данных, на основе информации о системной нагрузке, предоставляемой менеджером рабочей нагрузки (WLM).

DB2 Connect получает от WLM упорядоченный по приоритетам список экземпляров Sysplex. Каждый Sysplex возвращает взвешенное значение приоритета для каждого адреса соединения. Затем DB2 Connect использует этот список для обработки входных требований CONNECT, распределяя эти требования по экземплярам Sysplex с наибольшими приоритетами. Для балансировки нагрузки список взвешенных значений приоритета Sysplex получается при каждом соединении. Если включен концентратор соединений DB2 Connect, этот список используется также, чтобы определить, куда нужно послать каждую транзакцию.

Примечание: Для использования функций Sysplex DB2 Connect не требуется менять конфигурацию утилиты распределенных данных (Distributed Data Facility, DDF) OS/390 и z/OS.

Кроме того, DB2 Connect обеспечивает защиту от сбоев, для чего при неудачном соединении пытается соединиться с альтернативным компьютером sysplex. Ошибка возвращается прикладной программе только в том случае, когда не удается установить ни одно из известных соединений.

Функция Sysplex DB2 Connect использует пул агентов. Когда поддержка Sysplex включена, при обрыве соединения с одним из участников DB2 Connect направляет соединения к другому участнику DDF. Перенаправление соединений производится в соответствии со списком серверов Sysplex.

Используя концентратор соединений, DB2 Connect может балансировать рабочую нагрузку (распределение нагрузки происходит в моменты начала транзакций). Для этого должен быть включен концентратор DB2 Connect.

Применение DB2 Sysplex

В типичном сценарии сервер DB2 Connect (сервер А) находится в диалоге с двумя серверами DB2 для OS/390 и z/OS (серверы В и С).

Сервер Sysplex В	Сервер Sysplex С
HOST_NAME=MVSHOST	HOST_NAME=MVSHOST1

Предположим, что теперь в этом сценарии прикладная программа передает команду:

```
db2 connect to aliasb user xxxxxxx using xxxxxxx
```

Устанавливается соединение с базой данных на MVSHOST. Поскольку применение Sysplex включено как на сервере DB2 Connect, так и в записи каталога DCS, DB2 для OS/390 и z/OS указывает для DB2 Connect сетевые адреса для каждого участника Sysplex (MVSHOST и MVSHOST1. Для получения этой информации используются протоколы и потоки сообщений DRDA4). После установления исходного соединения возвращенный список адресов кэшируется на рабочей станции DB2 Connect. Если исходная команда CONNECT передана для узла TCP/IP, возвращаются IP-адреса.

Информация приоритета, используемая для балансировки нагрузки и защиты от сбоев

Список адресов, который предоставляет DB2 для OS/390 и z/OS, содержит также информацию приоритета, включающую число соединений для каждого сетевого адреса. Этот список обновляется для каждого нового соединения, которое устанавливает DB2 Connect. Эта дополнительная информация используется для балансировки нагрузки и для защиты от сбоев.

DB2 Connect использует кэшированный список адресов

Если не удастся установить соединение с базой данных ALIASB, генерируется сообщение об ошибке SQL30081N и соединение отбрасывается. Если вновь получено требование соединения с ALIASB, DB2 Connect делает следующее:

1. Он пытается соединиться с сервером с наибольшим приоритетом из кэшированного списка адресов; информацию приоритета сообщает DB2 для OS/390 и z/OS. DB2 Connect всегда использует эту стратегию, которая и обеспечивает балансировку нагрузки.
2. Если это соединение не удастся установить, предпринимается попытка установить соединение с другими адресами в списке (по убыванию значения приоритета, которое сообщила DB2 для OS/390 и z/OS). Так DB2 Connect использует информацию Sysplex для защиты от сбоев.
3. Если все остальные попытки соединения окажутся неудачными, DB2 Connect попытается соединиться с ALIASB, используя адрес из каталога узлов.

Для получения информации о серверах, связанных со средой Sysplex, можно использовать команду `db2pd` с параметром `sysplex` (`db2pd -sysplex`).

Требования конфигурации для Sysplex

Sysplex применяется для конкретной базы данных, только если в записи каталога DCS для этой базы данных в шестом позиционном параметре задано Sysplex (регистр символов не имеет значения).

Глава 19. Настройка DB2 Connect

Для настройки DB2 Connect можно изменять различные параметры в файле конфигурации менеджера баз данных.

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 сервере 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 операционных систем Linux и UNIX), чтобы установить, идет ли подкачка на вашей системе.

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 (`db2agent`). 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 переносятся общая для агентов баз данных информация о каталогах.

Объем памяти, занимаемой кэшем, зависит от числа записей в каждом каталоге.

NUMDB

В предыдущих версиях на поведение DB2 Connect не влиял параметр конфигурации **NUMDB**, однако начиная с Версии 8 это изменено. Этот параметр задает максимальное число баз данных, с которыми клиенты могут соединиться через сервер DB2 Connect. Говоря точнее, это максимальное число различных алиасов баз данных, которые можно внести в каталог на сервере DB2 Connect.

Другие параметры DB2 Connect

Параметры **AGENTPRI** и **MAXAGENTS** считаются в Версии 9.5 устаревшими. Удалены следующие два абзаца:

AGENTPRI применяется только при наличии удаленных клиентов. Параметр **AGENTPRI** 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 один экземпляр DB2 Connect с высоким приоритетом и один экземпляр DB2 Connect с низким приоритетом (то есть с разными значениями **AGENTPRI**).

Каждое соединение компьютера клиента с сервером баз данных хоста или System i 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 System i через рабочую станцию DB2 Connect.

Для повышения производительности в параметре **NUM_POOLAGENTS** нужно задать то же значение, что и в параметре **MAX_COORDAGENTS**, или же среднее число клиентов.

Чтобы отправлять учетные строки с клиентской прикладной программы на сервер DB2 Connect, используйте специальные средства API для установки информации учета. Специальные средства API обеспечивают лучшее быстроедействие, чем задание переменной среды DB2ACCOUNT.

Драйвер IBM Data Server для JDBC и SQLJ

Свойство `com.ibm.db2.jcc.DB2BaseDataSource.clientAccountingInformation`

DB2 .NET Data Provider

Свойство `DB2Connection.ClientAccountingInformation`

CLI/ODBC

Ключевое слово конфигурации `ClientAcctStr` CLI/ODBC

Embedded SQL (C, C++, and COBOL)

Функция `sqlsact`

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.

Настройка баз данных хоста

Производительность системы зависит от производительности сервера баз данных хоста или System i. Разные СУБД имеют свои особенности производительности. Например, оптимизаторы SQL разных систем могут по-разному работать с одной и той же программой. Если вы хотите узнать подробнее о производительности системы сервера баз данных хоста или System i, обратитесь к документации по ним.

Производительность можно повысить с помощью опций связывания "чтение непринятого" (UR) или "без принятия" (NC), которые отменяют запись в журнал, если это вас устраивает.

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.

Особенности настройки 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.

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.

В Табл. 17 кратко изложены методы настройки производительности сети.

Таблица 17. Методы настройки производительности сети

На что обратить внимание	Пример	Значение	Примечания
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.
Буферы	RUSIZE	Оптимальный размер - 4096.	Наилучшую производительность может дать одинаковый размер RUSIZE и RQRIOBLK.
Буферы	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.
Настройки TCP	Размер сегментов	1500 для Ethernet, 4400 для Token Ring и FDDI.	Для адаптеров ESCON в сети TCP/IP следует задавать значение 4096.
Настройки TCP	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.

Конфликты из-за системных ресурсов

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?

Устранение неисправностей, связанных с производительностью DB2 Connect

Если пользователи DB2 Connect в случае больших запросов долго ожидают ответа от серверов хоста или System i, то причину низкой производительности можно искать в следующих областях:

1. Для запросов, для которых с сервера хоста или System i возвращаются большие блоки данных (обычно от 32 Кбайт и выше), установите для параметра конфигурации менеджера баз данных RQRIOBLK значение 32767. Это можно сделать, введя в командной строке команду:


```
db2 update database manager configuration using RQRIOBLK 32767
```
2. 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.

Настройка DB2 в OS/390 и z/OS

В OS/390 и z/OS можно оптимизировать обработку остановленных потоков. В Версии 5 допускается до 25000 одновременно соединенных клиентов. 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.

Увеличение скорости передачи данных DB2 Connect

В дополнение к блокированию строк для набора результатов, DB2 для OS/390 и z/OS может возвращать для удаленного клиента, например, для DB2 Connect, несколько блоков запроса в ответ на требование OPEN или FETCH. Вместо периодической передачи требований на сервер DB2 для OS/390 и z/OS, который за один раз запрашивает один блок строковых данных, теперь клиент имеет возможность посылать на сервер требования передавать обратно добавочные блоки запросов. Такие блоки запросов называются дополнительными блоками запросов.

Эта новая особенность позволяет клиенту минимизировать число циклов обработки строк в сети, что является главным фактором, влияющим на сетевую производительность. Снижение числа посылаемых клиентом на сервер требований блоков запросов приводит к существенному увеличению производительности, так как переключение между передачей и приемом - дорогая операция, сильно

сказывающаяся на производительности. DB2 Connect может теперь реализовать это улучшение производительности, запрашивая у сервера DB2 для OS/390 и z/OS дополнительные блоки запросов.

Чтобы полностью использовать преимущество возврата дополнительных блоков запросов (каждый из них может быть длиной до 32 Кбайт), если в сети применяется протокол TCP/IP, в DB2 Connect поддерживаются также расширения window scaling, как описано в RFC-1323. Эта новая особенность позволяет TCP/IP динамически подстраивать размеры окон передачи и приема для больших объемов данных, которые могут быть возвращены при использовании дополнительных блоков запросов.

Дополнительные блоки запроса

Поддержка дополнительных блоков запросов на серверах DB2 Universal Database (UDB) для OS/390 и z/OS Версии 7 или новее настраивается с помощью параметра EXTRA BLOCKS SRV на панели установки DB2 DDF. Этот параметр управляет максимальным числом дополнительных блоков запроса, которые DB2 может послать в ответ на требование назад клиенту; допустимы значения от 0 до 100. Задание значения 0 отключает возврат дополнительных блоков запросов. Для получения наибольшей выгоды от этой особенности следует использовать значение по умолчанию 100, отбросив все опасения, которые могут привести к установке для этого параметра значения меньше оптимального.

На стороне клиента, когда программа обращается к DB2 для z/OS либо непосредственно через сопутствующую этому клиенту установку DB2 Connect, либо через отдельную установку сервера DB2 Connect, существуют различные средства для активации соответствующей поддержки DB2 Connect на основе указателя или оператора:

- На основе размера набора строк запроса для указателя
- На основе условия 'OPTIMIZE for N ROWS' оператора select, связанного с указателем
- На основе условия 'FETCH FIRST N ROWS ONLY' оператора select, связанного с указателем

DB2 Connect может включить поддержку дополнительных блоков запросов с помощью нескольких API SQL:

Встроенный SQL

- Поддержка дополнительных блоков запроса вызывается для запроса заданием условия 'OPTIMIZE for N ROWS' и/или условия 'FETCH FIRST N ROWS ONLY' в самом операторе select.
- Если задано условие 'OPTIMIZE for N ROWS', DB2 для z/OS и OS/390 будет пытаться блокировать указанное число строк для возврата их в DB2 Connect согласно заданному значению параметра установки EXTRA BLOCKS SRV DDF. Конкретная программа может выбрать получение числа строк больше N, поскольку DB2 for z/OS не ограничивает числом N общее число строк, возвращаемых в конечном счете для набора результатов запроса.
- Условие 'FETCH FIRST N ROWS ONLY' работает аналогично, но DB2 для OS/390 и z/OS ограничивает набор результатов запроса N строками. Считывание числа строк больше N приведет к возврату кода SQL +100 (конец данных).

CLI/ODBC

- Поддержка дополнительных блоков запроса вызывается для запроса с использованием в нем атрибута оператора SQL_MAX_ROWS.
- Для серверов DB2 UDB для OS/390 и z/OS Версии 7.1 и новее вместо 'OPTIMIZE for N ROWS' используется условие 'FETCH FIRST N ROWS ONLY'.
 - В Версии 7 DB2 для OS/390 и z/OS ограничивает набор результатов запроса N строками. Считывание числа строк больше N приведет к возврату SQL_NO_DATA_FOUND.
 - В Версии 8 и новее CLI обеспечивает использование менеджера указателей клиента и возвращение прикладной программе только первых N строк.

JDBC Поддержка дополнительных блоков запроса вызывается для запроса с использованием метода setMaxRows. Как и в случае CLI/ODBC, DB2 Connect отмечает для сервера DB2 для OS/390 и z/OS 6.x условие 'OPTIMIZE for N ROWS'. Кроме того, для серверов DB2 для z/OS версии 7.1 и новее DB2 Connect использует условие 'FETCH FIRST N ROWS ONLY'.

Изменение размеров окна RFC-1323

Изменение размеров окна поддерживается на всех платформах Windows, Linux, и UNIX, которые поддерживают расширения RFC-1323 для TCP/IP. Эту функцию можно включить в DB2 для Windows, Linux или UNIX с помощью переменной реестра DB2 DB2SORCVBUF. Для включения изменения размеров окна задайте для этой переменной любое значение, превышающее 64К. Например, в DB2 для Windows, Linux или UNIX введите команду db2set DB2SORCVBUF =65537.

Максимальные размеры буферов передачи и приема зависят от конкретной операционной системы. Чтобы убедиться, что сконфигурированные размеры буфера приняты, пользователь может задать для параметра конфигурации DIAGLEVEL менеджера баз данных значение 4 (информационное) и просмотреть сообщения в журнале уведомлений администратора.

Чтобы изменение размеров окна действовало, оно должно быть подключено на обоих концах соединения (на рабочей станции и хосте) либо напрямую из стека TCP/IP операционной системы, либо из продукта DB2. Для DB2 for z/OS, например, в настоящее время изменение размеров окна можно активировать только через операционную систему, задав для TCPRCVBUFRSIZE значение больше 64 Кбайт. Если вы используете удаленный IBM data server для доступа к базе данных DB2 на хосте или System i через рабочую станцию сервера DB2 Connect, изменение размеров окна можно также включить на клиенте. Аналогичным образом изменение размеров окна включается и между удаленным клиентом IBM Data Server и сервером DB2 рабочей станции, когда база данных DB2 хоста или System i не используется.

Хотя изменение размеров окна предназначено для повышения сетевой производительности, ожидаемое улучшение производительности сети наблюдается не всегда. Различное сочетание факторов, таких как размер кадра сетевых адаптеров Ethernet или Token Ring, размер максимального блока передачи для IP-пакетов (IP MTU) и других параметров в маршрутизаторах по линии связи, после подключения изменение размеров окна может привести даже к снижению производительности. Поэтому по умолчанию изменение размеров окна отключено: и для буфера передачи, и для буфера приема задается размер 64 Кбайт.

Вам следует подготовить пользователя, чтобы он мог оценить воздействие от подключения изменения размеров окна и провести все необходимые настройки в

сети. Начальную информацию по настройке сети для улучшения производительности смотрите по адресу <http://www.networking.ibm.com/>.

Преобразование данных хоста

При передаче информации между различными средами (такими как Intel [Windows], IEEE [операционные системы Linux и UNIX], zSeries [VM, VSE, z/OS], System i [OS/400]) может потребоваться преобразование числовых типов данных (десятичных, целых, десятичных с плавающей точкой). Оно может влиять на производительность.

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 System i.

Символьные типы 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.

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 System i.

- 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 сети в 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.

Глава 20. Настройка производительности прикладных программ CLI/ODBC

CLI/ODBC - это интерфейс прикладного программирования SQL, который можно вызывать из ваших прикладных программ баз данных. Функции CLI вызывают хранимые процедуры DB2, которые, в свою очередь, обращаются к таблицам системного каталога.

Некоторые прикладные программы используют API ODBC для сбора метаданных, которые используются при дальнейшей обработке. Возможны десять вызовов API метаданных:

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns

Некоторые прикладные программы CLI/ODBC, которые используют вышеперечисленные API метаданных, могут запрашивать любые объекты базы данных. Например, вызов SQLTables запрашивает метаданные для всех таблиц базы данных. В большой системе такие запросы могут сильно увеличить сетевой трафик, занять долгое время и поглотить большое количество ресурсов сервера.

Некоторые параметры инициализации CLI/ODBC 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
- SysSchema
- GrantorList
- GranteeList

Часть 5. Устранение неисправностей

Глава 21. Устранение неисправностей

Среда DB2 Connect включает много программных и аппаратных средств и средств связи. При устранении неисправностей разумно исключать гипотезы, уточняя доступные данные, пока не будет получено заключение (то есть найден источник ошибки).

Собрав информацию и выбрав тему, переходите к соответствующему разделу.

Сбор связанной информации

Устранение неисправностей включает в себя 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 одним или несколькими клиентами и сервером (шлюзом) DB2 Connect, между шлюзом DB2 Connect и сервером баз данных хоста или System i или между DB2 Connect Personal Edition и сервером баз данных хоста или System i?
- Что можно определить по содержанию сообщения и по возвращенным в нем элементам?
- Есть ли смысл применять сейчас такие диагностические программы, как db2trc, db2pd или db2support?
- 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 операционных систем 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 операционных систем UNIX: внесли ли вы в файл блокировки узла пароль, предоставленный IBM?

4. *Правильно ли были сконфигурированы сервер баз данных хоста или System i программы связи на рабочей станции?*
 - Обратите внимание на три конфигурации:
 - a. Конфигурация сервера баз данных хоста или System i идентифицирует реквестер прикладных программ для сервера. На СУБД сервера хоста или System i 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 System i для клиента.
 - c. В конфигурации клиентской рабочей станции должно быть указано имя рабочей станции и протокол связи.
 - Исследуя ошибку начального соединения, надо убедиться, что все имена 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 System i, и у администратора сети есть утилиты диагностики.
5. *Есть ли у вас уровень полномочий, требуемый в системе управления базами данных на сервере хоста или System i для использования базы данных сервера хоста или System i?*
 - 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 System i с помощью процессора командной строки?*
 - Выполнена ли процедура связывания процессора командной строки с сервером баз данных хоста или System i?

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 был ли недавно установлен пакет Fix Pack? 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 Fix Pack IBM и загрузите его *после* установки функции.
2. *Возникла 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?*

- Для 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.
 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 TCP/IP попробуйте сделать PING для удаленного хоста.*

Средства диагностики

При ошибках можно использовать следующие средства:

- Все данные диагностики, включая файлы дампа, файлы перехвата, журналы ошибок, файлы уведомлений и журналы оповещений находятся в пути, задаваемом в параметре конфигурации менеджера баз данных **diagpath** - пути каталога данных диагностики:

В случае пустого значения этого параметра конфигурации данные диагностики записываются в один из следующих каталогов (папок):

- Для сред Linux и UNIX: `INSTHOME/sqllib/db2dump`, где *INSTHOME* - домашний каталог экземпляра.
- Для поддерживаемых сред Windows:
 - Если переменная среды **DB2INSTPROF** не задана, используется `x:\SQLLIB\DB2INSTANCE`, где `x:\SQLLIB` - буква диска и каталог, заданный в переменной реестра **DB2PATH**, а значение **DB2INSTANCE** - имя экземпляра.

Примечание: Каталог не обязан называться SQLLIB.

- Если переменная среды **DB2INSTPROF** задана, используется `x:\DB2INSTPROF\DB2INSTANCE`, где **DB2INSTPROF** - имя каталога профиля экземпляра, а **DB2INSTANCE** - имя экземпляра (по умолчанию - значение **DB2INSTDEF** в 32-битных операционных системах Windows).
- В операционных системах Windows для просмотра журнала применяется Окно просмотра событий.
- Для использования доступны следующие средства диагностики: **db2trc**, **db2pd** и **db2support**.
- Для операционных систем Linux и UNIX - команда **ps**, которая выводит в стандартный выходной поток информацию о состоянии процесса по всем активным процессам.

- В операционных системах UNIX при серьезных ошибках в текущем каталоге создается файл ядра. Он содержит образ памяти прерванного процесса и позволяет определить, какая функция вызвала ошибку.

Глава 22. Трассировки DB2 в DB2 Connect

При устранении неисправности выполняемые в вашей среде действия и операции трассировки позволяют получить полезную информацию. Данные трассировки, выполненной в продукте сервера баз данных DB2, можно получить, выгрузить и сформатировать. Утилита трассировки поставляется как часть продукта сервера баз данных DB2.

Получение трассировки DB2 при помощи db2trc

Команда **db2trc** управляет утилитой трассировки, поставляемой с DB2. Утилита трассировки записывает информацию об операциях и преобразует ее в удобный для чтения формат.

Учтите, что выполнение трассировки предполагает дополнительные затраты, поэтому включение утилиты трассировки может уменьшить производительность системы.

Обычно средства трассировки DB2 используются службой поддержки DB2 и коллективами разработчиков для устранения неисправностей. Выполнив трассировку, можно получить информацию об ошибках, однако так как вы не знаете исходный код DB2, использование трассировки для вас ограничено.

Тем не менее, важно знать, как правильно включить трассировку и как записать файлы трассировки, если вас попросят их предоставить.

Примечание: Для использования db2trc потребуются полномочия SYSADM, SYSCTRL или SYSMAINT.

Чтобы получить общее представление о доступных опциях, введите команду db2trc без параметров:

```
C:\>db2trc
```

```
Использование: db2trc (chg|clr|dmp|flw|fmt|inf|off|on) опции
```

Для вывода дополнительной информации о конкретных параметрах команды db2trc используйте опцию -u. Например, чтобы просмотреть дополнительную информацию о включении трассировки, введите команду:

```
db2trc on -u
```

Она выведет информацию обо всех дополнительных опциях, помеченных как "facilities" (возможности), которые можно задать при включении трассировки DB2.

При включенной трассировке наиболее важна опция -L. Она задает размер буфера памяти, который будет использоваться для хранения трассируемой информации. Размер буфера можно задать в байтах или мегабайтах. (Чтобы задать значения в мегабайтах, добавьте после числа "M" или "m"). Размер буфера трассировки должен быть равен двум мегабайтам в степени N. Если задать значение, не отвечающее этому требованию, размер буфера трассировки автоматически будет округлен до ближайшего значения, кратного двум в степени N.

При слишком малом буфере трассировки можно потерять информацию. По умолчанию при заполнении буфера сохраняется только наиболее свежая информация трассировки. Слишком большой буфер трассировки может затруднить отправку файла в службу поддержки DB2.

Для трассировки относительно коротких операций (таких как соединение с базой данных) обычно достаточен размер около 8 Мбайт:

```
C:\> db2trc on -l 8M
Trace is turned on
(Трассировка включена)
```

Однако при трассировке более длинных операций или одновременном выполнении работы большого объема может потребоваться буфер трассировки более крупного размера.

На большинстве платформ трассировку можно включить в любое время, и она будет выполняться, как описано выше. Однако об определенных ситуациях следует знать следующее:

1. В системах многораздельных баз данных трассировку нужно запускать для каждого физического (а не логического) раздела базы данных.
2. На платформах HP-UX, Linux и Solaris, если отключить трассировку после запуска экземпляра, то при следующем запуске трассировки будет использоваться очень маленький буфер, независимо от заданного размера. Например, вчера вы включили трассировку при помощи команды `db2trc on -l 8m`, после чего собрали данные и отключили трассировку (`db2trc off`). Сегодня вы хотите запустить трассировку с заданным для буфера памяти размером 32 мегабайта (`db2trc on -l 32m`) без остановки и перезапуска экземпляра. Вы увидите, что в этом случае для трассировки будет выделен маленький буфер. Для эффективного выполнения трассировки на этих платформах включайте трассировку перед запуском экземпляра с необходимым вам размером буфера, после чего “очищайте” буфер по мере необходимости.

Дамп файла трассировки DB2

После включения трассировки при помощи опции `on` будет трассироваться вся последующая работа, выполняемая экземпляром.

Пока выполняется трассировка, можно использовать опцию `clr` для очистки буфера трассировки. Вся существующая информация в этом буфере трассировки будет удалена.

```
C:\>db2trc clr
Trace has been cleared
(Трассировка очищена)
```

По завершении трассируемой операции примените опцию `dmp`, указав после нее имя файла трассировки, чтобы выгрузить буфер памяти на диск. Например:

```
C:\>db2trc dmp trace.dmp
Trace has been dumped to file
(Трассировка выгружена в файл)
```

После выгрузки буфера трассировки на диск утилита трассировки продолжит свою работу. Для отключения трассировки используйте опцию `off`:

```
C:\>db2trc off
Trace is turned off
(Трассировка выключена)
```

Форматирование файла трассировки DB2

Команда `db2trc dmp` создает файл дампа в двоичном формате, и его нельзя прочитать.

Для проверки читаемости двоичного файла трассировки его можно отформатировать, чтобы вывести управление потоком, и отправить отформатированный вывод в устройство Null. Команда, выполняющая эту задачу, показана в следующем примере:

```
db2trc flw example.trc nul
```

где `example.trc` - двоичный файл, сгенерированный при помощи опции `dmp`.

Вывод этой команды явно покажет, есть ли проблемы при чтении этого файла и была ли перенесена информация трассировки.

После проверки файл дампа можно отправить в службу поддержки DB2. Специалисты службы отформатируют его на основе вашего уровня обслуживания DB2. Однако иногда вас могут попросить преобразовать файл дампа в формат ASCII перед его отправкой. Для этого используются опции `flw` и `fmt`. Нужно указать имя двоичного файла дампа и имя файла ASCII, который вы хотите создать:

```
C:\>db2trc flw trace.dmp trace.flw
C:\Temp>db2trc flw trace.dmp trace.flw
Total number of trace records      : 18854
(Общее число записей трассировки)
Trace truncated                    : NO
(Трассировка усечена)
Trace wrapped                       : NO
(Трассировка перенесена)
Number of trace records formatted  : 1513 (pid: 2196 tid 2148 node: -1)
(Число отформатированных записей трассировки)
Number of trace records formatted  : 100 (pid: 1568 tid 1304 node: 0)
(Число отформатированных записей трассировки)
...
```

```
C:\>db2trc fmt trace.dmp trace.fmt
C:\Temp>db2trc fmt trace.dmp trace.fmt
Trace truncated                    : NO
(Трассировка усечена)
Trace wrapped                       : NO
(Трассировка перенесена)
Total number of trace records      : 18854
(Общее число записей трассировки)
Number of trace records formatted  : 18854
(Число отформатированных записей трассировки)
```

Если в этом выводе для "Trace wrapped" указано "YES", это означает, что буфер трассировки слишком мал, чтобы вместить всю собранную за время трассировки информацию. В зависимости от ситуации трассировка с переносом может быть и успешной. Если вас интересует только самая последняя информация (эта часть информации сохраняется по умолчанию, если только не задана опция `-i`), то содержимого файла трассировки может оказаться достаточно. Но если нужна информация о событиях в начале периода трассировки, может потребоваться повторить операцию с использованием буфера трассировки большего размера.

При форматировании двоичного файла в читаемый текстовый файл доступны некоторые опции. Например, команда `db2trc fmt -xml trace.dmp trace.fmt` позволяет преобразовать двоичные данные и вывести результат в доступном для

синтаксического анализа формате xml. Дополнительные опции приведены в подробном описании команды трассировки db2trc.

Кроме того, следует знать, что в операционных системах Linux и UNIX DB2 выгружает буфер трассировки на диск автоматически при закрытии экземпляра из-за серьезной ошибки. Таким образом, если в момент ненормального завершения работы экземпляра трассировка была включена, в каталоге диагностики создается файл с именем db2trdmp.###, где ### - номер раздела базы данных. На платформах Windows этого не происходит. В этом случае надо выгружать информацию трассировки вручную.

Вот пример обычной последовательности команд db2trc:

```
db2trc on -l 8M
db2trc clr
<Выполняем команды восстановления ошибки>
db2trc dump db2trc.dmp
db2trc off
db2trc flw db2trc.dmp <имя_файла>.flw
db2trc fmt db2trc.dmp <имя_файла>.fmt
db2trc fmt -c db2trc.dmp <имя_файла>.fmtc
```

Глава 23. Файлы трассировки DRDA

Перед анализом трассировок DRDA следует понимать, что DRDA - открытый стандарт для определения структур данных и связи. Например, в DRDA входит набор правил организации данных для передачи и правил передачи этой информации. Эти правила изложены в следующих справочных руководствах:

- DRDA Версии 3, том 1: Distributed Relational Database Architecture
- DRDA Версии 3, том 2: Formatted Data Object Content Architecture
- DRDA Версии 3, том 3: Distributed Data Management Architecture

Версии PDF этих руководств доступны на сайте www.opengroup.org.

Утилита **db2drdat** предназначена для записи данных, передаваемых между реквестером программ DRDA (Application Requestor, AR) и сервером прикладных программ DRDA Application Server, AS) DB2 (например, между DB2 Connect и сервером баз данных хоста или Series i.)

Утилита трассировки

Утилита db2drdat предназначена для записи данных, передаваемых между сервером DB2 Connect (который обслуживает IBM data server) и сервером баз данных хоста или System i.

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 System i, но при ее выполнении происходит ошибка и вы получаете код возврата. Если вы знаете, какая информация передается системе управления сервером баз данных хоста или System i, можно определить причину ошибки, даже если код возврата содержит только самую общую информацию. Многие проблемы вызываются просто пользовательскими ошибками.

В выходной информации db2drdat перечисляются потоки данных, которыми обменивались рабочая станция DB2 Connect и система управления сервером баз данных хоста или System i. Данные, посланные на сервер баз данных хоста или System i, помечаются SEND BUFFER, а данные, полученные от сервера баз данных хоста или System i, 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 System i. 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

Вывод трассировки

Утилита db2drdat записывает в *файл трассировки* следующую информацию:

- -r
 - Тип объекта/ответа DRDA
 - Буфер приема
- -s
 - Тип требования DRDA
 - Буфер отправки
- -c
 - SQLCA
- Информация об ошибках TCP/IP
 - Код возврата функции приема
 - Уровень серьезности
 - Используемый протокол
 - Используемый API
 - Функция
 - Номер ошибки

Примечание:

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.
3. Возвращаемые поля могут быть разными в зависимости от платформы, на которой выполняется DB2 Connect, даже для одного и того же API.
4. Если db2drdat направляет вывод в уже существующий файл, старое содержимое файла будет стерто, если разрешения для файла позволяют это сделать.

Анализ файла трассировки

В данные трассировки db2drdat попала следующая информация:

- ID процесса (PID) программы клиента
- RDB_NAME, занесенное в каталог DCS
- CCSID DB2 Connect
- CCSID сервера баз данных хоста или System i
- Система управления сервером баз данных хоста или System i, 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 System i. Эти команды отправляются, когда вводится команда базы данных CONNECT TO.

Следующий буфер содержит ответ, который DB2 Connect получает от системы управления сервером баз данных хоста или System i. Он содержит ответные данные Exchange Server Attributes (EXCSATRD) и ответное сообщение Access RDB (ACCRDBRM).

EXCSAT

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'9481A292' (в кодировке CCSID 500) в символьном виде, если отбросить X'116D', дает *mask*.

Команда EXCSAT также содержит объект EXTNAM (External Name), который часто записывается в диагностическую информацию системы управления сервером баз данных хоста или System i. 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 db2br, дополненное пробелами, за которым следует 000C50CC. Для клиентов IBM Data Server в Linux или UNIX это значение соответствует значению в команде ps, которая записывает в стандартный вывод информацию о состоянии активных процессов.

ACCRDB

Команда ACCRDB содержит RDB_NAME в объекте RDBNAM, код которого - X'2110'. Команда ACCRDB следует за командой EXCSAT в первом буфере. Величина X'E2E3D3C5C3F1' в команде ACCRDB в символьном виде, если отбросить X'2110' - это STLEC1. Это соответствует полю имени базы данных назначения в каталоге DCS.

Код учетной строки - X'2104'.

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'0333', то есть 819.

Дополнительные объекты CCSIDDBC (CCSID для двухбайтных символов) и CCSIDMBC (CCSID для символов смешанной длины), с кодами соответственно X'119D' и X'119E', также присутствуют в команде ACCRDB. В нашем примере CCSIDSBC - это X'04B0', то есть 1200, а CCSIDMBC - X'0333', то есть, соответственно, 819.

EXCSATRD и ACCRDBRM

Величины CCSID возвращаются также с сервера баз данных хоста или System i в ответном сообщении Access RDB (ACCRDBRM) во втором буфере. Этот буфер содержит EXCSATRD и следующий за ним ACCRDBRM. Выходной файл примера содержит два значения CCSID для системы сервера баз данных хоста или System i. Это значения 1208 (для однобайтных и смешанных символов) и 1200 (для двухбайтных символов).

Если DB2 Connect не распознает кодовую страницу, пришедшую с сервера баз данных хоста или System i, пользователю вместе с кодовыми страницами источника и назначения будет возвращен SQLCODE -332. Если сервер баз данных хоста или System i 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 -332.

ACCRDBRM также содержит параметр PRDID (идентификатор продукта, с кодом X'112E'). Значение равно X'C4E2D5F0F8F0F1F5', то есть DSN08015 в EBCDIC. Согласно стандартам, DSN - это DB2 Universal Database for z/OS and OS/390. Указывается также номер версии. ARI - это сервер DB2 Server for VSE & VM, SQL - база данных DB2 или DB2 Connect, а QSQ - это DB2 for i5/OS.

Примеры файлов вывода трассировки

Ниже приведен пример вывода, где показаны некоторые потоки данных DRDA между рабочими станциями DB2 Connect и сервером баз данных хоста или System i. С точки зрения пользователя будет введена команда базы данных CONNECT TO с использованием процессора командной строки (CLP).

рис. 13 на стр. 143 использует соединение DB2 Connect Enterprise Edition Версии 9.1 и DB2 Universal Database (UDB) for z/OS Версии 8 через TCP/IP.

1 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 0 probe 100
 bytes 16

Data1 (PD_TYPE_UINT,8) unsigned integer:
 233

2 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 19532 probe 1177
 bytes 250

SEND BUFFER(AR):

	EXCSAT RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	00C3D041000100BD 1041007F115E8482	...A.....A...^..	.C}.....".;db
0010	F282974040404040 4040404040404040	...@@@@@@@@@@@@	2bp
0020	4040F0F0F0C3F5F0 C3C3F0F0F0000000	@@.....	000C50CC000...
0030	0000000000000000 0000000000000000
0040	0000000000000000 000000000060F0F0-00
0050	F0F1A2A495404040 4040404040404040@@@@@@@@@@	01sun
0060	4040404040404040 4040404040404040	@@@@@@@@@@@@@@	
0070	C4C5C3E5F8404040 F0A2A49540404040@@@...@@@	DECV8 0sun
0080	4040404040404040 4000181404140300	@@@@@@@@@.....
0090	0724070008147400 05240F0008144000	.\$...t.\$...@.
00A0	08000E1147D8C4C2 F261C1C9E7F6F400	...G...a.....QDB2/AIX64.
00B0	08116D9481A29200 0C115AE2D8D3F0F9	..m.....Z.....	.._mask...]SQL09
00C0	F0F0F0	...	000

	ACCSECRQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0026D00100020020 106D000611A20003	..&....m.....	..}.....s...
0010	00162110E2E3D3C5 C3F1404040404040	..!.....@@@@@	...STLECI
0020	40404040404040	@@@@@	

3 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110546200 probe 100
 bytes 12

Data1 (PD_TYPE_UINT,4) unsigned integer:
 105

4 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110549755 probe 1178
 bytes 122

RECEIVE BUFFER(AR):

	EXCSATRD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0059D04300010053 1443000F115EE5F8	.Y.C...S.C...^..	..}.....;V8
0010	F1C14BE2E3D3C5C3 F100181404140300	..K.....	1A.STLECI.....
0020	0724070007147400 05240F0007144000	.\$...t.\$...@.
0030	0700081147D8C4C2 F20014116DE2E3D3	...G.....m...QDB2..._STL
0040	C5C3F14040404040 4040404040000C11	...@@@@@@@@@...	EC1 ...
0050	5AC4E2D5F0F8F0F1 F5	Z.....]DSN08015

	ACCSECRD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0010D0030002000A 14AC000611A20003}.....s...

5 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110656806 probe 100
 bytes 16

Data1 (PD_TYPE_UINT,8) unsigned integer:
 233

Рисунок 13. Пример вывода трассировки (соединение типа TCP/IP)

6 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110659711 probe 1177
 bytes 250

SEND BUFFER(AR):

	SECCHK RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	003CD04100010036	106E000611A20003	.<.A...6.n.....
0010	00162110E2E3D3C5	C3F1404040404040	..!.....@@@
0020	40404040404000C	11A1D9858799F485	@@@.....
0030	A599000A11A09585	A6A39695vr....newton

	ACCRDB RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	00ADD001000200A7	20010006210F2407!.\$.
0010	00172135C7F9F1C1	F0C4F3C14BD7C1F8	..!5.....K...
0020	F806030221064600	162110E2E3D3C5C3	...!.F.!.....
0030	F140404040404040	4040404040000C11	.@@@@@@@@@... 1
0040	2EE2D8D3F0F9F0F0	F000D002FD8E3C4/... .SQL09000....QTD
0050	E2D8D3C1E2C30016	00350006119C03335.....3 SQLASC.....
0060	0006119D04B00006	119E0333003C21043.

7 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 259908001 probe 100
 bytes 12

Data1 (PD_TYPE_UINT,4) unsigned integer:
 176

8 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 259911584 probe 1178
 bytes 193

RECEIVE BUFFER(AR):

	SECCHKRM RPYDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0015D0420001000F	1219000611490000	...B.....I..
0010	000511A400	u.

	ACCRDBRM RPYDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	009BD00200020095	2201000611490000"....I..
0010	000D002FD8E3C4E2	D8D3F3F7F0000C11	../. ..QTDSQL370...
0020	2EC4E2D5F0F8F0F1	F5001600350006115... .DSN08015.....
0030	9C04B80006119E04	B80006119D04B000
0040	0C11A0D5C5E6E3D6	D540400006212524@...!%\$...NEWTON
0050	34001E244E000624	4C00010014244D00	4..\$N..\$L...\$M.+...<.....(.
0060	06244FFFFF000A11	E8091E768301BE00	.\$0.....v.... ..!.....Y...c...
0070	2221030000000005	68B3B8C7F9F1C1F0	"!.....h..... ..G91A0
0080	C4F3C1D7C1F8F840	4040400603022106@@@...!. D3APA88
0090	46000A11E8091E76	831389	F.....v....Y...c.i

9 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 364420503 probe 100
 bytes 16

Data1 (PD_TYPE_UINT,8) unsigned integer:
 10

Рисунок 14. Пример вывода трассировки (соединение типа TCP/IP), продолжение

10 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 364440751 probe 1177
 bytes 27

SEND BUFFER(AR):

	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}.....

11 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 475009631 probe 100
 bytes 12

Data1 (PD_TYPE_UINT,4) unsigned integer:
 54

12 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 475014579 probe 1178
 bytes 71

RECEIVE BUFFER(AR):

	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	00162110E2E3D3C5 C3F1404040404040	..!.....@#@#@#STLEC1
0020	4040404040400005 211501	@#@#@#...!

	SQLCARD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	000BD00300010005 2408FF\$. .	..}.....

13 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 721710319 probe 100
 bytes 16

Data1 (PD_TYPE_UINT,8) unsigned integer:
 126

14 data DB2 UDB DRDA Communication Manager sqljcsend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 721727276 probe 1177
 bytes 143

SEND BUFFER(AR):

	EXCSQLIMM RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0053D0510001004D 200A00442113E2E3	..S.Q...M ..D!...	..}....(.....ST
0010	D3C5C3F140404040 4040404040404040@#@#@#@#@#@#	LEC1
0020	D5E4D3D3C9C44040 4040404040404040@#@#@#@#@#	NULLID
0030	4040E2D8D3C3F2C6 F0C1404040404040	@@.....@#@#@#	SQLC2F0A
0040	4040404041414141 41484C5600CB0005	@#@@AAAAAHLV....<.....
0050	2105F1	!..	..1

	SQLSTT OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	002BD00300010025 2414000000001B64	..+....%\$.d	..}.....
0010	656C657465206672 6F6D206464637375	elete from ddcsu	..%.....?_.....
0020	73312E6D79746162 6C65FF	s1.mytable.	..._`./.%..

15 data DB2 UDB DRDA Communication Manager sqljcreceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 832901261 probe 100
 bytes 12

Data1 (PD_TYPE_UINT,4) unsigned integer:
 102

Рисунок 15. Пример вывода трассировки (соединение типа TCP/IP), продолжение

16 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 832906528 probe 1178
 bytes 119

RECEIVE BUFFER(AR):

	SQLCARD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0066D00300010060 240800FFFFFF3434	.f.....`\$.44	..}....-.....
0010	3237303444534E58 4F544C2000FFFFFFE	2704DSNXOTL+!.<.....
0020	0C00000000000000 00FFFFFFF000000
0030	00000000000572020 2057202020202020W W
0040	001053544C454331 2020202020202020	..STLEC1<.....
0050	2020000F44444353 5553312E4D595441	..DDCSUS1.MYTA(...
0060	424C450000FF	BLE...<.....

17 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 833156953 probe 100
 bytes 16

Data1 (PD_TYPE_UINT,8) unsigned integer:
 10

18 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 833159843 probe 1177
 bytes 27

SEND BUFFER(AR):

	RDBRLLBCK RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	000AD00100010004 200F}.....

19 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 943302832 probe 100
 bytes 12

Data1 (PD_TYPE_UINT,4) unsigned integer:
 54

20 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 943306288 probe 1178
 bytes 71

RECEIVE BUFFER(AR):

	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	00162110E2E3D3C5 C3F1404040404040	..!.....@#@#@@STLEC1
0020	4040404040400005 211502	@#@#@@...!..

	SQLCARD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	000BD00300010005 2408FF\$. .	..}.....

Рисунок 16. Пример вывода трассировки (соединение типа TCP/IP), продолжение

Информация последующего буфера для трассировки DRDA

Для получения дополнительной информации можно проанализировать последующие буферы. Следующее требование содержит информацию о принятии. Команда commit указывает системе управления сервером баз данных хоста или System i выполнить принятие для текущей единицы работы. Четвертый буфер принимается от системы управления сервером баз данных хоста или System i в результате выполнения

принятия или отката. Он содержит ответное сообщение Конец единицы работы (ENDUOWRM), указывающее на завершение текущей единицы работы.

В нашем примере запись трассировки 12 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).

На рис. 13 на стр. 143 показан пример буфера приема со значением SQLCA ошибки в записи трассировки 16.

Часть 6. Сообщения

Глава 24. 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.

SQL0965 или SQL0969

Признак

Сообщения SQL0965 и SQL0969 могут посылааться с множеством различных кодов возврата от DB2 for i5/OS, DB2 for z/OS, and 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 или i5/OS, не поддается преобразованию. Исправьте ошибку, исходя из ее кода, и повторите невыполненную команду.

SQL5043N

Признак

Не были успешно запущены системы поддержки одного или нескольких протоколов связи. Однако ядро менеджера баз данных запущено успешно.

Возможно, на сервере 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:

```
2001-05-30-14.09.55.321092 Экземпляр:svtdbm5 Узел: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 сервер для удаленных клиентов, испытывает затруднения при обработке одного или нескольких клиентских протоколов связи. Это могут быть протоколы TCP/IP 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, svcename или pname).

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 с сервером DB2 Connect, все еще есть хэндл на порту TCP/IP. На каждом клиентском компьютере, который соединен с сервером DB2 Connect, введите команды:

```
db2 terminate
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. Выполните команду db2support до того, как обращаться к службе поддержки.

SQL30060

Признак

SQL30060N У "*ID_авторизации*" нет привилегии на выполнение операции "*операция*".

Решение

При соединении с DB2 для OS/390 и z/OS таблицы базы данных связей (Communications Database, CDB) не были правильно изменены.

SQL30061

Признак

Произошло соединение с неверным хостом или сервером баз данных System i - 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 z/OS and 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.

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_данных>  
db2 catalog database <локальное_имя> as <алиас> at node <имя_узла>  
authentication server
```

После этого, чтобы связаться с базой данных, используется команда:
db2 connect to <алиас> user <имя_пользователя> using <пароль>

SQL30081N с кодом возврата 79

Признак

SQL30081N Ошибка связи.
Используемый протокол
связи: "TCP/IP". Используемый 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 с сервером DB2 Connect. Она может также произойти при соединении с хостом или сервером баз данных System i с сервера DB2 Connect.

1. Возможно, на сервере DB2 Connect неверно задана переменная профиля DB2COMM. Проверьте, так ли это. Например, если DB2 Enterprise Server Edition работает в AIX, в профиле sql1lib/db2profile должна быть команда db2set db2comm=tcpip.
2. Возможно, имя службы TCP/IP и номер порта на клиенте IBM Data Server не соответствуют заданным на сервере DB2 Connect. Проверьте записи в файлах services TCP/IP на обоих компьютерах.
3. Проверьте, запущена ли DB2 на сервере 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:

```
2001-02-03-12.41.04.861119 Экземпляр:svtdbm2 Узел: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 API связи: "SOCKETS".
Положение
где обнаружена ошибка: "9.21.85.159".
Функция связи, обнаружившая ошибку: "send".
Коды ошибки протокола связи: "10032",
"*, "*".
SQLSTATE=08001

Решение

Это сообщение об ошибке может быть получено при попытке рассоединения

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.

SQL30082 RC=24 при выполнении оператора CONNECT

Признак

SQLCODE -30082 Неправильное имя пользователя или пароль.

Решение

Убедитесь, что в операторе CONNECT пароль указан верно. Пароль нельзя отправить базе данных сервера назначения. IBM data server должен отправить пароль базе данных сервера назначения. На некоторых платформах, например, на AIX, пароль можно использовать, только если он указан в операторе CONNECT.

Часть 7. Приложения

Приложение А. Обзор технической информации DB2

Для получения технической информации по DB2 служат следующие инструменты и методы:

- Информационный центр DB2
 - Темы (Задачи, принципы работы и справочные темы)
 - Справка по инструментам DB2
 - Примеры программ
 - Обучающие программы
- Книги по DB2
 - Файлы PDF (загружаемые)
 - Файлы PDF (из DB2 PDF DVD)
 - Печатные книги
- Справка командной строки
 - Справка по командам
 - Справка по сообщениям

Примечание: Темы Информационного центра DB2 обновляются чаще, чем книги в формате PDF или бумажные книги. Чтобы получить наиболее свежую информацию по DB2, устанавливайте обновления сразу после их появления или используйте Информационный центр DB2 Information Center на сайте ibm.com.

Дополнительную техническую информацию о DB2, такую как технические замечания (technotes), оригинальные публикации (white papers) и технические руководства (IBM Redbooks) можно найти на сайте ibm.com. Обратитесь к библиотеке на сайте DB2 Information Management по адресу <http://www.ibm.com/software/data/sw-library/>.

Отзывы о документации

Мы ценим ваши отзывы о документации по DB2. Если у вас есть предложения по совершенствованию документации по DB2, присылайте электронные сообщения на адрес db2docs@ca.ibm.com. Коллектив разработчиков документации DB2 читает все ваши отзывы, но не может отвечать напрямую. Чтобы мы могли лучше понять ваши трудности, по возможности представьте конкретные примеры. Если ваш отзыв касается конкретной темы или файла справки, приложите заголовок справки и URL.

Не обращайтесь по этому адресу с вопросами, требующими службы поддержки заказчиков DB2. Если у вас есть технический вопрос по DB2, который не удастся разрешить по документации, обратитесь за помощью в ваш местный центр обслуживания IBM.

Техническая библиотека DB2 в печатном виде и в формате PDF

Ниже в таблице описан состав библиотеки DB2, доступной в Центре публикаций IBM по адресу www.ibm.com/shop/publications/order. Английские руководства по DB2 Версия 9.5 в формате PDF и переведенные версии можно загрузить с сайта www.ibm.com/support/docview.wss?rs=71&uid=swg2700947.

В таблицах описаны книги, доступные в печатном виде, тем не менее, в вашей стране или регионе они могут не быть доступны в печатном виде.

Таблица 18. Техническая информация по DB2

Название	Номер формы	Доступна в печатном виде
<i>Administrative API Reference</i>	SC23-5842-00	Да
<i>Administrative Routines and Views</i>	SC23-5843-00	Нет
<i>Call Level Interface Guide and Reference, Volume 1</i>	SC23-5844-00	Да
<i>Call Level Interface Guide and Reference, Volume 2</i>	SC23-5845-00	Да
<i>Command Reference</i>	SC23-5846-00	Да
<i>Data Movement Utilities Guide and Reference</i>	SC23-5847-00	Да
<i>Data Recovery and High Availability Guide and Reference</i>	SC23-5848-00	Да
<i>Data Servers, Databases, and Database Objects Guide</i>	SC23-5849-00	Да
<i>Database Security Guide</i>	SC23-5850-00	Да
<i>Developing ADO.NET and OLE DB Applications</i>	SC23-5851-00	Да
<i>Developing Embedded SQL Applications</i>	SC23-5852-00	Да
<i>Developing Java Applications</i>	SC23-5853-00	Да
<i>Developing Perl and PHP Applications</i>	SC23-5854-00	Нет
<i>Developing User-defined Routines (SQL and External)</i>	SC23-5855-00	Да
<i>Getting Started with Database Application Development</i>	GC23-5856-00	Да
<i>Начинаем работу по установке и управлению DB2 в Linux и Windows</i>	GC89-1002-00	Да
<i>Internationalization Guide</i>	SC23-5858-00	Да
<i>Message Reference, Volume 1</i>	GI11-7855-00	Нет
<i>Message Reference, Volume 2</i>	GI11-7856-00	Нет
<i>Руководство по перенастройке</i>	GC89-1001-00	Да
<i>Net Search Extender Administration and User's Guide</i>	SC23-8509-00	Да
Примечание: Содержимое этого документа не входит в Информационный центр DB2.		
<i>Partitioning and Clustering Guide</i>	SC23-5860-00	Да
<i>Query Patroller Administration and User's Guide</i>	SC23-8507-00	Да
<i>Быстрый старт для клиентов IBM Data Server</i>	GC89-1004-00	Нет
<i>Быстрый старт для серверов DB2</i>	GC89-1003-00	Да

Таблица 18. Техническая информация по DB2 (продолжение)

Название	Номер формы	Доступна в печатном виде
<i>Spatial Extender and Geodetic Data Management Feature User's Guide and Reference</i>	SC23-8508-00	Да
<i>SQL Reference, Том 1</i>	SC23-5861-00	Да
<i>SQL Reference, Том 2</i>	SC23-5862-00	Да
<i>System Monitor Guide and Reference</i>	SC23-5865-00	Да
<i>Text Search Guide</i>	SC23-5866-00	Да
<i>Troubleshooting Guide</i>	GI11-7857-00	Нет
<i>Tuning Database Performance</i>	SC23-5867-00	Да
<i>Учебник по Наглядному объяснению</i>	SC89-1008-00	Нет
<i>What's New</i>	SC23-5869-00	Да
<i>Workload Manager Guide and Reference</i>	SC23-5870-00	Да
<i>pureXML Guide</i>	SC23-5871-00	Да
<i>XQuery Reference</i>	SC23-5872-00	Нет

Таблица 19. Техническая информация о DB2 Connect

Название	Номер формы	Доступна в печатном виде
<i>Быстрый старт для DB2 Connect Personal Edition</i>	GC89-1006-00	Да
<i>Быстрый старт для серверов DB2 Connect</i>	GC89-1007-00	Да
<i>DB2 Connect. Руководство пользователя</i>	SC89-1005-00	Да

Таблица 20. Техническая информация об Information Integration

Название	Номер формы	Доступна в печатном виде
<i>Information Integration: Administration Guide for Federated Systems</i>	SC19-1020-01	Да
<i>Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i>	SC19-1018-02	Да
<i>Information Integration: Configuration Guide for Federated Data Sources</i>	SC19-1034-01	Нет
<i>Information Integration: SQL Replication Guide and Reference</i>	SC19-1030-01	Да
<i>Information Integration: Introduction to Replication and Event Publishing</i>	SC19-1028-01	Да

Заказ печатных копий книг DB2

Если вам нужны печатные книги по DB2, то вы можете купить их через Интернет во многих (но не во всех) странах и территориях. Вы всегда можете заказать печатные книги по DB2 у своего местного представителя IBM. Имейте в виду, что некоторые книги, которые есть на диске DVD *Документация по DB2 в формате PDF*, нельзя заказать в печатном виде. В частности, оба тома *Справочника по сообщениям DB2* недоступны в печатном виде.

Печатные версии многих публикаций по DB2, находящихся на диске DVD *Документация по DB2 в формате PDF*, можно заказать за плату в IBM. В зависимости от того, где вы находитесь, вы, возможно, сможете заказать эти книги через Интернет, в центре публикаций IBM Publications Center. Если в вашей стране или территории заказ через Интернет не удастся сделать, вы всегда можете заказать печатные книги по DB2 у вашего местного представителя IBM. Заметим, что некоторые книги, которые есть на диске DVD *Документация по DB2 в формате PDF*, нельзя заказать в печатном виде.

Примечание: Наиболее свежая и полная информация по DB2 находится в Информационном центре DB2 по адресу <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5>.

Чтобы заказать печатные книги DB2:

- Чтобы узнать, можете ли вы в свое стране или территории заказать печатные книги по DB2 в Интернете, проверьте Центр публикаций IBM Publications Center по адресу: <http://www.ibm.com/shop/publications/order>. Чтобы перейти к информации о заказе, вам надо будет выбрать свою страну, область или язык, а затем следовать инструкциям для вашего местоположения.
- Чтобы заказать печатные книги по DB2 у своего местного представителя IBM:
 1. На одном из следующих Web-сайтов найдите, как к вашему местному представителю:
 - Всемирный указатель IBM по контактам: www.ibm.com/planetwide
 - Сайт Центра публикаций IBM <http://www.ibm.com/shop/publications/order>. Чтобы перейти к странице, соответствующей вашему местоположению, вам надо будет выбрать свою страну, область или язык. На этой странице выберите ссылку "Об этом сайте".
 2. Если будете звонить, скажите, что хотите заказать публикацию по DB2.
 3. Сообщите представителю заголовки и форм-номера книг, которые хотите заказать. Эти заголовки и форм-номера можно найти в разделе "Техническая библиотека DB2 в печатном виде и в формате PDF" на стр. 157.

Вызов справки по SQLSTATE из командной строки

Система DB2 возвращает SQLSTATE - значения, описывающие ошибки, которые могут возникнуть при выполнении оператора SQL. Справка по SQLSTATE объясняет значение состояний SQL и коды классов состояний SQL.

Для вызова справки по SQLSTATE откройте процессор командной строки и введите:

```
? sqlstate или ? код класса
```

где *sqlstate* - допустимый пятизначный код SQL, а *код класса* - первые две цифры sqlstate.

Например, ? 08003 выводит справку по состоянию SQL 08003, а ? 08 выводит справку по коду класса 08.

Доступ к различным версиям Информационного центра DB2

Для тем DB2 Версии 9.5 используйте следующий URL Информационного центра DB2: <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>

Для тем DB2 Версии 9 используйте следующий URL Информационного центра DB2: <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>

Темы DB2 Версии 8 можно найти в Информационном центре Версии 8 по адресу: <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>.

Вывод тем Информационного центра DB2 на предпочитаемом вами языке

Информационный центр DB2 по возможности выводит темы на языке, заданном в предпочтениях браузера. Если тема еще не переведена на предпочитаемый вами язык, Информационный центр DB2 выводит ее по-английски.

- Для вывода тем на предпочитаемом вами языке в браузере Internet Explorer:
 1. В Internet Explorer выберите **Сервис** —> **Свойства обозревателя** —> **Языки...** Откроется окно Выбор языка.
 2. Убедитесь, что предпочитаемый вами язык указан в первой строке списка языков.
 - Чтобы добавить в список новый язык, нажмите кнопку **Добавить...**
 - Примечание:** При добавлении языка не гарантируется, что на компьютере есть шрифты, требуемые для вывода тем на предпочитаемом языке.
 - Чтобы переместить язык в начало списка, выделите язык и нажимайте кнопку **Выше**, пока он не окажется первым в списке языков.
 3. Очистите кэш браузера, а затем обновите страницу, чтобы вывести Информационный центр DB2 на предпочитаемом вами языке.
- Для вывода тем на предпочитаемом вами языке в браузере Firefox или Mozilla:
 1. Выберите кнопку в разделе **Языки** диалогового окна **Инструменты** —> **Опции** —> **Дополнительно**. В окне Настройки появится панель Языки.
 2. Убедитесь, что предпочитаемый вами язык указан в первой строке списка языков.
 - Чтобы добавить в список новый язык, нажмите кнопку **Добавить...** и выберите нужный язык в окне Добавление языков.
 - Чтобы переместить язык в начало списка, выделите язык и нажимайте кнопку **Выше**, пока он не окажется первым в списке языков.
 3. Очистите кэш браузера, а затем обновите страницу, чтобы вывести Информационный центр DB2 на предпочитаемом вами языке.

При некоторых сочетаниях браузер-операционная система вам, возможно, придется к тому же изменить региональные настройки операционной системы на нужные вам локаль и язык.

Обновление Информационного центра DB2, установленного на вашем компьютере или на сервере интранета

Если вы установили Информационный центр DB2 локально, можно загружать и устанавливать обновления, которые IBM сделала доступными.

Для обновления вашего локально установленного Информационного центра DB2 Information Center:

1. Остановите Информационный центр DB2 на вашем компьютере и перезапустите его в автономном режиме. Запуск Информационного центра в автономном режиме предотвращает доступ к нему других пользователей по сети и дает возможность загрузить и применить исправления.
2. Чтобы посмотреть доступные обновления, воспользуйтесь возможностью Изменить. При помощи этой возможности можно загрузить и установить интересующие вас обновления.

Примечание: Если в вашей среде необходимо установить обновления Информационного центра DB2 на компьютере, не подключенном к интернету, необходимо создать зеркальную копию сайта обновлений в локальной файловой системе, используя компьютер, который подключен к интернету и на котором установлен Информационный центр DB2. Кроме того, создание зеркальной копии и прокси-сервера сайта обновлений полезно, если в вашей сети многие пользователи будут устанавливать обновления документации, и вы хотите сократить время, затраченное на отдельные обновления.

Если пакеты доступны, используйте эту возможность для загрузки пакетов. Но возможность обновления доступна только в автономном режиме.

3. Остановите Информационный центр в автономном режиме и перезапустите Информационный центр DB2 на вашем компьютере.

Примечание: В Windows Vista приведенные ниже команды должен запускать администратор. Чтобы вывести окно командной строки или графическое средство со всеми привилегиями администратора, щелкните по ярлыку правой кнопкой и выберите **Выполнить от имени администратора**.

Чтобы обновить DB2, установленный на вашем компьютере или на сервере интранета:

1. Остановите Информационный центр DB2.
 - В Windows выберите **Пуск → Панель управления → Администрирование → Службы**. Затем щелкните правой кнопкой мыши по службе **Информационный центр DB2** и выберите **Стоп**.
 - В Linux введите команду:
`/etc/init.d/db2icdv95 stop`
2. Запустите Информационный центр в автономном режиме.
 - В Windows:
 - a. Откройте командное окно.
 - b. Перейдите в каталог, где установлен Информационный центр. По умолчанию Информационный центр DB2 установлен в каталоге <Program Files>\IBM\DB2 Information Center\Version 9.5, где <Program Files> соответствует положению каталога Program Files.
 - c. Перейдите из каталога установки в каталог doc\bin.
 - d. Запустите файл help_start.bat:
`help_start.bat`

- В Linux:
 - a. Перейдите в каталог, где установлен Информационный центр. По умолчанию Информационный центр DB2 установлен в каталоге `/opt/ibm/db2ic/V9.5`.
 - b. Перейдите из каталога установки в каталог `doc/bin`.
 - c. Запустите сценарий `help_start`:


```
help_start
```

В системном браузере по умолчанию откроется Информационный центр в автономном режиме.

3. Нажмите кнопку Изменить (🔧). На правой панели Информационного центра выберите Найти изменения. Будет выведен список изменений для существующей документации.
4. Чтобы начать процесс загрузки, пометьте документы, которые вы хотите загрузить, затем нажмите кнопку Установить изменения.
5. Когда процесс загрузки и установки будет завершен, нажмите кнопку Готово.
6. Остановите Информационный центр в автономном режиме.
 - В Windows перейдите в каталоге установки в подкаталог `doc\bin` и запустите файл `help_end.bat`:


```
help_end.bat
```

Примечание: Пакетный файл `help_end` содержит команды, необходимые для корректного прерывания процессов, запущенных пакетным файлом `help_start`. Не используйте сочетание клавиш `Ctrl-C` или другие методы для прерывания работы `help_start.bat`.

- В Linux перейдите в каталоге установки в подкаталог `doc/bin` и запустите сценарий `help_end`:


```
help_end
```

Примечание: Сценарий `help_end` содержит команды, необходимые для корректного прерывания процессов, запущенных сценарием `help_start`. Не используйте другие методы для прерывания работы сценария `help_start`.

7. Перезапустите Информационный центр DB2.
 - В Windows выберите **Пуск → Панель управления → Администрирование → Службы**. Затем щелкните правой кнопкой мыши по службе **Информационный центр DB2** и выберите **Пуск**.
 - В Linux введите команду:


```
/etc/init.d/db2icdv95 start
```

В обновленном Информационном центре DB2 будут показаны новые и измененные темы.

Учебные материалы по DB2

Обучающие программы DB2 помогают познакомиться с различными аспектами продуктов DB2. В уроках содержатся пошаговые инструкции.

Прежде, чем вы начнете

XHTML-версию обучающей программы можно просмотреть в Информационном центре на сайте <http://publib.boulder.ibm.com/infocenter/db2help/>.

В некоторых уроках используются примеры данных или кодов программ. Описание необходимых условий для выполнения задач обучающей программы смотрите в самой обучающей программе.

Учебные материалы по DB2

Для просмотра Обучающей программы щелкните по заголовку.

“pureXML” в книге *pureXML Guide*

Конфигурирование базы данных DB2 для хранения данных XML и выполнение основных операций со складом собственных данных XML.

“Учебник по Наглядному объяснению” в книге *Учебник по Наглядному объяснению*

Анализ, оптимизация и настройка операторов SQL для улучшения производительности при помощи Наглядного объяснения.

Информация об устранении неисправностей DB2

Для продуктов DB2 доступна разнообразная информация по устранению неисправностей и диагностике ошибок.

Документация DB2

Информацию по устранению неисправностей можно найти в Руководстве по устранению неисправностей DB2 или в разделе Поддержка и устранение неисправностей Информационного центра DB2. Здесь вы найдете информацию о том, как определять и идентифицировать проблемы с помощью диагностических инструментов и средств DB2, а также решения наиболее часто возникающих проблем и другие советы как решать проблемы, с которыми вы можете встретиться при работе с вашими продуктами DB2.

Сайт технической поддержки DB2

Если вы сталкиваетесь с проблемами и вам нужна помощь в поиске причин и решений, обратитесь на сайт технической поддержки DB2 (DB2 Technical Support). На сайте технической поддержки есть ссылки на самые свежие публикации по DB2, TechNotes, исправления программ (APAR), пакеты корректировки и прочие ресурсы. Для решения ваших проблем можно воспользоваться поиском по этой базе знаний.

Web-сайт технической поддержки DB2 находится по адресу
<http://www.ibm.com/software/data/db2/udb/support.html>

Положения и условия

Разрешения на использование этих публикаций предоставляются в соответствии со следующими положениями и условиями.

Личное использование: вам предоставляется право воспроизводить эти публикации в личных некоммерческих целях при условии, что будут воспроизведены все замечания об авторских правах. Вам запрещается распространять эти публикации, полностью или по частям, демонстрировать их или создавать из них производные продукты без явного на то согласия от IBM.

Коммерческое использование: вам предоставляется право воспроизводить эти публикации исключительно в пределах своего предприятия при условии, что будут воспроизведены все замечания об авторских правах. За пределами вашего

предприятия вам запрещается распространять эти публикации, полностью или по частям, демонстрировать их или создавать из них производные продукты без явного на то согласия от IBM.

На публикации или любую информацию, данные, программные средства или иную интеллектуальную собственность, содержащуюся в публикациях, предоставляются только права, явным образом предоставленные настоящим документом; никакие иные разрешения, лицензии и права не предоставляются.

IBM оставляет за собой право отозвать разрешения, предоставленные этим документом, если, по мнению IBM, использование публикаций наносит ущерб IBM или, как это установлено IBM, вышеприведенные инструкции не соблюдаются должным образом.

Вы имеете право загружать, экспортировать или реэкспортировать эту информацию только при условии соблюдения всех применимых законов и нормативных актов, включая все законы и нормативные акты США, касающиеся экспорта.

IBM НЕ ДАЕТ НИКАКИХ ГАРАНТИЙ ОТНОСИТЕЛЬНО СОДЕРЖАНИЯ ЭТИХ ПУБЛИКАЦИЙ. ДАННЫЕ ПУБЛИКАЦИИ ПРЕДОСТАВЛЕНЫ "КАК ЕСТЬ", БЕЗ КАКИХ-ЛИБО ГАРАНТИЙ, ЯВНЫХ ИЛИ ПОДРАЗУМЕВАЕМЫХ, ВКЛЮЧАЯ (НО НЕ ОГРАНИЧИВАЯСЬ ТАКОВЫМИ) ПРЕДПОЛАГАЕМЫЕ ГАРАНТИИ СОБЛЮДЕНИЯ АВТОРСКИХ ПРАВ, РЫНОЧНОЙ ПРИГОДНОСТИ ИЛИ СООТВЕТСТВИЯ ОПРЕДЕЛЕННОЙ ЦЕЛИ.

Приложение В. Замечания

Эта информация описывает продукты и услуги, предлагаемые в США.

IBM может не предлагать описанные продукты, услуги и возможности в других странах. Сведения о продуктах и услугах, доступных в настоящее время в вашей стране, можно получить в местном представительстве IBM. Любые ссылки на продукты, программы или услуги IBM не означают явным или неявным образом, что можно использовать только продукты, программы или услуги IBM. Разрешается использовать любые функционально эквивалентные продукты, программы или услуги, если при этом не нарушаются права IBM на интеллектуальную собственность. Однако ответственность за оценку и проверку работы любых продуктов, программ и услуг других фирм лежит на пользователе.

Фирма IBM может располагать патентами или рассматриваемыми заявками на патенты, относящимися к предмету данного документа. Получение этого документа не означает предоставления каких-либо лицензий на эти патенты. Запросы по поводу лицензий следует направлять в письменной форме по адресу:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

По поводу лицензий, связанных с использованием наборов двухбайтных символов (DBCS), обращайтесь в отдел интеллектуальной собственности IBM в вашей стране/регионе 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/регионе, где подобные заявления противоречат местным законам: КОРПОРАЦИЯ INTERNATIONAL BUSINESS MACHINES ПРЕДСТАВЛЯЕТ ДАННУЮ ПУБЛИКАЦИЮ “КАК ЕСТЬ” БЕЗ КАКИХ-ЛИБО ГАРАНТИЙ, ЯВНЫХ ИЛИ ПОДРАЗУМЕВАЕМЫХ, ВКЛЮЧАЯ ПРЕДПОЛАГАЕМЫЕ ГАРАНТИИ СОВМЕСТИМОСТИ, РЫНОЧНОЙ ПРИГОДНОСТИ И СООТВЕТСТВИЯ ОПРЕДЕЛЕННОЙ ЦЕЛИ, НО НЕ ОГРАНИЧИВАЯСЬ ИМИ. В некоторых странах для определенных сделок подобные оговорки не допускаются; таким образом, это утверждение может не относиться к вам.

Данная информация может содержать технические неточности и типографские опечатки. Периодически в информацию вносятся изменения, они будут включены в новые издания этой публикации. Фирма IBM может в любое время без уведомления вносить изменения и усовершенствования в продукты и программы, описанные в этой публикации.

Этот документ может содержать ссылки на внешние по отношению к IBM сайты и ресурсы. IBM не делает никаких заявлений и не дает никаких гарантий относительно содержания таких внешних сайтов или ресурсов других разработчиков, доступных по

ссылками из данного документа. Ссылка на внешний сайт не означает, что IBM рекомендует или одобряет использование этого сайта или его владельца. Кроме того, IBM не участвует и не несет ответственности по каким-либо сделкам заключенным между вами и независимыми сторонами, даже если вы узнали об этих сторонах на сайте IBM. Соответственно, вы соглашаетесь и подтверждаете, что IBM не несет ответственности за такие внешние сайты и ресурсы и не отвечает за их содержимое, услуги, продукты или прочие материалы, доступные в связи с этими сайтами или ресурсами. Использование любого программного обеспечения независимых разработчиков регулируется условиями лицензии, прилагаемой к этому программному обеспечению.

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
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

Такая информация может быть предоставлена на определенных условиях (в некоторых случаях к таким условиям может относиться оплата).

Лицензированная программа, описанная в данном документе, и все лицензированные материалы, доступные с ней, предоставляются IBM на условиях IBM Customer Agreement (Соглашения IBM с заказчиком), Международного соглашения о лицензиях на программы IBM или эквивалентного соглашения.

Приведенные данные о производительности измерены в контролируемой среде. Таким образом, результаты, полученные в других операционных средах, могут существенно отличаться от них. Некоторые показатели измерены получены в системах разработки и нет никаких гарантий, что в общедоступных системах эти показатели будут теми же. Более того, некоторые результаты могут быть получены путем экстраполяции. Реальные результаты могут отличаться от них. Пользователи должны проверить данные для своих конкретных сред.

Информация о продуктах других фирм получена от поставщиков этих продуктов, из их опубликованных объявлений или из других общедоступных источников. Фирма IBM не проверяла эти продукты и не может подтвердить точность измерений, совместимость или прочие утверждения о продуктах других фирм. Вопросы о возможностях продуктов других фирм следует направлять поставщикам этих продуктов.

Все утверждения о будущих планах и намерениях IBM могут быть изменены или отменены без уведомлений, и описывают исключительно цели фирмы.

Эта информация может содержать примеры данных и отчетов, иллюстрирующие типичные деловые операции. Чтобы эти примеры были правдоподобны, в них

включены имена лиц, названия компаний и товаров. Все эти имена и названия вымышлены и любое их сходство с реальными именами и адресами полностью случайно.

ЛИЦЕНЗИЯ НА КОПИРОВАНИЕ:

Эта информация может содержать примеры прикладных программ на языках программирования, иллюстрирующих приемы программирования для различных операционных платформ. Разрешается копировать, изменять и распространять эти примеры программ в любой форме без оплаты фирме IBM для целей разработки, использования, сбыта или распространения прикладных программ, соответствующих интерфейсу прикладного программирования операционных платформ, для которых эти примера программ написаны. Эти примеры не были всесторонне проверены во всех возможных условиях. Поэтому IBM не может гарантировать их надежность, пригодность и функционирование.

Каждая копия программ примеров или программ, созданных на их основе, должна содержать следующее замечание об авторских правах:

© (название вашей фирмы) (год). Части этого кода построены на основе примеров программ IBM Corp. © Copyright IBM Corp. *_вставьте год или годы_*. Все права защищены.

Товарные знаки

Названия компаний, продуктов и услуг в документах библиотеки DB2 Версия 9.5 могут быть товарными знаками или марками сервиса International Business Machines Corporation или других фирм. Информацию о товарных знаках IBM Corporation в Соединенных Штатах и/или других странах смотрите по адресу <http://www.ibm.com/legal/copytrade.shtml>.

Следующие термины, используемые по крайней мере в одном из документов библиотеки документации DB2, являются товарными знаками или зарегистрированными товарными знаками других компаний:

Microsoft, Windows, Windows NT и логотип Windows - товарные знаки корпорации Microsoft в США и/или других странах.

Intel, Intel logo, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium и Pentium - товарные знаки Intel Corporation в Соединенных Штатах и/или других странах.

Java и все товарные знаки на основе Java - товарные знаки Sun Microsystems, Inc. в Соединенных Штатах и/или в других странах.

UNIX - зарегистрированный товарный знак The Open Group в Соединенных Штатах и в других странах.

Linux - зарегистрированный товарный знак Linus Torvalds в Соединенных Штатах и/или других странах.

Adobe, логотип Adobe, PostScript и логотип PostScript - зарегистрированные товарные знаки или товарные знаки Adobe Systems Incorporated в Соединенных Штатах и/или других странах.

Названия других компаний, продуктов и услуг могут быть товарными знаками или марками сервиса других фирм.

Индекс

Спец. символы

, (запятая) в строке параметров 33
,, (запятая запятая) в строке параметров 33

A

AS имя базы данных назначения 33

C

CCSID (идентификатор кодового набора символов)
поддержка двух направлений письма
описание 33

CDRA (character data representation architecture - архитектура
представления символьных данных) 10

CLI (call level interface, интерфейс уровня вызовов)
доверенные соединения 41
обзор 127
прикладные программы
CURRENTPACKAGESET 48

D

DB2 Connect
DCE
защита 48
защита 41
концентратор соединений 112
обзор 3
перемещение данных с помощью 59
поддержка Sysplex 113
понятия 8
сценарии
мониторы транзакций 15
усовершенствования по сравнению с предыдущими
версиями 3

DB2 Connect Enterprise Edition
API 21
JDBC 21
SQLJ 21
tuxedo 26
Web-серверы 22
менеджер транзакций стандарта XA 57
мониторы транзакций 26
программы Web 19
сервер соединений 18
сценарии серверов соединений 15

DB2 Connect Personal Edition
описание продукта 3

DB2 Universal Database for OS/390 and z/OS 31
DOMAIN 31
DYNAMICRULES(BIND) 48
RESPORT 31
TCPPORT 31
набор данных начальной загрузки 31
параметры BSDS 31
усовершенствования системы защиты
защита прикладных программ ODBC и Java в настольных
системах 48

DB2 Universal Database for OS/390 and z/OS *(продолжение)*
усовершенствования системы защиты *(продолжение)*
поддержка смены пароля 48
Поле TCP/IP security already verified 48
расширенные коды защиты 48

DB2_RESOLVE_CALL_CONFLICT
переменная реестра 69, 105, 113
db2trc (утилита трассировки DB2)
запуск, обзор 135
форматирование вывода трассировки 137

DCE
предварительные требования 48

DSS (распределенный подраздел)
тип, трассировка 139

F

FDOCA (Formatted Data Object Content Architecture - Архитектура
содержимого объекта форматированных данных) 10

I

IBM SQL 6
IBM WebSphere 21
ID авторизации 72
ID кодовой страницы 72
ID программы хоста 72
ID продукта клиента 72
ID продукта хоста 72

J

Java
серверы прикладных программ
API 21
DB2 Connect EE 21
JDBC 21
SQLJ 21

M

Microsoft Windows
прикладные программы 15

N

NULLID
OS/400 51
NUMDB
параметр 117

O

ODBC
интерфейс 15
обзор 127
оптимизация доступа 99

ODBC (*продолжение*)
прикладные программы
CURRENTPACKAGESET 48
OS/390
DRDA 10
OS/400
DRDA 10

R

Relational Connect
описание продукта 8

S

SOCKS
узел
обязательные переменные среды 31
SQL (Structured Query Language, язык структурированных запросов)
динамический 101
статический 101
SQL/DS
DRDA 10
SQL_ATTR_
TRUSTED_CONTEXT_PASSWORD
использование 44
TRUSTED_CONTEXT_USERID
использование 44
USE_TRUSTED_CONTEXT
использование 42
SQLCA (область связи SQL)
буферы данных 139
поле SQLCODE 139
SQLCODE
отображение 63
поле в SQLCA 139
файл отображения 63
SQLDA (область дескриптора SQL)
размер выделяемой 101
SQLSTATE
коды классов 63
Sysplex
балансировка нагрузки 114
использование 114
особенности zSeries 113
отказоустойчивость 114
параметр 33
поддержка в DB2 Connect 113
приоритет 114
требования конфигурации 115
System i
DRDA 10

T

TCP/IP
DOMAIN 31
RESPOPT 31
TCPPORT 31
защита
проверенная 48
сценарии 49
имена служб 31
имена удаленных хостов 31, 37
имена хостов 37

TCP/IP (*продолжение*)
команда ACCSEC 140
команда SECCHK 140
конфигурирование
соединения хоста 17
номера портов 37
порт повторной синхронизации 31
расширения RFC-1323
изменение размера окна 123
TCP_KEEPALIVE
параметр конфигурации операционной системы 91
Tuxedo
DB2 Connect Enterprise Edition 26

V

VM
DRDA
и DB2 Connect 10
VSE
DRDA 10

W

Web-серверы
DB2 Connect Enterprise Edition 22
WebSphere
Advanced Edition 21
Enterprise Edition 21
Standard Edition 21
возможности 21
обзор 21

X

XA
доверенные соединения 41
менеджеры ресурсов 26
менеджеры транзакций
концентраторы соединений 107
описание 26
примеры концентратора 107

Z

z/OS
DRDA 10

A

автоматическое перенаправление клиента
описание 89
отказы соединений 91
установка 89
алиас клиента базы данных 72
амперсанд (двойные ())
файл отображения SQLCODE 63
аппаратное обеспечение
производительность сети 124
Архитектура распределенных реляционных баз данных (DRDA)
CDRA (character data representation architecture - архитектура представления символьных данных) 10
FDOCA (Formatted Data Object Content Architecture - Архитектура содержимого объекта форматированных данных) 10

Архитектура распределенных реляционных баз данных (DRDA)

(продолжение)

MSA Management Services Architecture - Архитектура служб управления) 10

TCP/IP 10

архитектуры 10

доступ к данным 9

обзор 9

понятия 9

поток данных 10

распределенное управление данными (DDM) 10

реквестер прикладных программ 10

сервер прикладных программ 10

аутентификация 37

обзор 46

проверка 46

типы

CLIENT 46, 48

KERBEROS 46

SERVER 46

SERVER_ENCRYPT 46

по умолчанию 46

Б

базы данных

алиас 31, 37

группировка требований 101

имя 31, 33, 37

объект RDBNAM 140

настройка 119

понятия

MVS 6

OS/390 6

OS/400 6

VM 6

VSE 6

z/OS 6

средства производительности 95

базы данных назначения

имя 33, 37

базы данных объединения

распределенное требование 12

базы данных хоста

элемент мониторинга имени 72

базы данных хостов

соединения

балансировка нагрузки 87

высокая доступность 87

блоки запросов

повышение скорости передачи данных DB2 Connect 121

блокирование данных 101

буфер отправки

данные трассировки 139

буфер приема 139

В

в файле отображения SQLCODE 63

виртуальный телекоммуникационный метод доступа (virtual telecommunications access method, VTAM) 49

возможность трассировки 135

трассировки DB2 137

Трассировки DB2 135, 136

трассировки DRDA 142, 146

восстановление связи

клиентские программы 89

время ответа 95

высокая доступность

DB2 Connect 85

Д

данные

блокирование 101

источники

распределенное требование 12

передача

между хостом и рабочей станцией 59

потoki 10

производительность 95

преобразования

хост 124

производительность передачи 124

скорость передачи 95, 124

даты

поддержка часовых поясов 33

двухфазное принятие

активизация 55

порт повторной синхронизации для соединений TCP/IP 31

диагностика ошибок

оперативная информация 164

ошибки после соединения 132

проблемы соединения 131

средства диагностики 133

учебные материалы 164

динамический SQL

CURRENTPACKAGESET 48

вопросы производительности 101

эффекты обработки 6

доверенные соединения 41

переключение пользователей при помощи CLI/ODBC 44

через CLI/ODBC 42

доверенный контекст

поддержка в DB2 Connect 41

через CLI/ODBC 42

документация

PDF или печатная 157

положения и условия использования 164

дополнительный блок запроса

CLI/ODBC 122

JDBC 122

встроенный SQL 122

Е

единицы работы (units of work, UOW)

определение 9

распределенные 55

удаленная 11

Ж

журнал оценки для политики

Монитор работоспособности DB2 for z/OS 76

журналы

оценка политики 76

3

- заказ книг DB2 160
- замечания 167
- запуск
 - Монитор работоспособности DB2 for z/OS 77
- защита
 - DB2 Connect
 - особенности 48
 - поддержка 49
 - Kerberos 47
 - TCP/IP 49
 - введение 41
 - значения каталога узла 31
 - оператор GRANT 49
 - оператор REVOKE 29
 - расширенные коды
 - OS/390 и z/OS 48
 - советы 48
 - типы 37
- значение аутентификации 31
- зонный десятичный тип данных 124

И

- изменение размера окна
 - расширения RFC-1323 123
- изменения
 - Информационный центр 162
 - Информационный центр DB2 162
 - каталоги баз данных 31
- измерение
 - производительность 95
- имя клиента NNAME 72
- Информационный центр
 - версии 161
 - изменение 162
 - просмотр на различных языках 161
- Информационный центр DB2
 - версии 161
 - изменение 162
 - просмотр на различных языках 161

К

- каталог Database Connection Services (DCS)
 - изменение записей 31
- каталог DCS
 - AS имя базы данных назначения 33
 - задание строки параметров 38
 - имя базы данных 33
 - имя базы данных назначения 33
 - параметр BIDI 33
 - параметр LOCALDATE 33
 - параметр SYSPLEX 33
 - содержимое 33
- каталоги
 - настройка
 - рабочие листы 37
- каталоги баз данных
 - DCS (database connection services - службы соединений с базами данных) 31
 - изменение 31
 - несколько записей 38
 - системная база данных 31
 - узел 31

- клиентские программы
 - восстановление связи 89
- ключевое слово CURRENTPACKAGESET CLI/ODBC 48
- код ошибки SQL0965 151
- код ошибки SQL0969 151
- код ошибки SQL1338 31, 151
- код ошибки SQL30020 151
- код ошибки SQL30060 151
- код ошибки SQL30061 151
- код ошибки SQL30073 151
- код ошибки SQL30081N 151
- код ошибки SQL30082 151
- код ошибки SQL5043N 151
- команда ACCRDB 140
- команда ACCRDBRM 140
- команда ACCSEC 140
- команда commit
 - выходные буферы трассировки 140
- команда Exchange Server Attributes 140
- команда EXCSAT 140
- команда EXCSATRD 140
- команда FORCE
 - ID агента для 72
- Команда GET SNAPSHOT 70
- команда LIST DCS APPLICATIONS 72
- команда SECCHK 140
- команда системы MVS START 76
- команда системы MVS STOP 76
- команды
 - ACCRDB 140
 - ACCRDBRM 140
 - ACCSEC 140
 - commit 140
 - EXCSAT 140
 - EXCSATRD 140
 - GET SNAPSHOT 70
 - SECCHK 140
- комплект поставки продукта 3
- конфигурирование
 - особенности, изменение пароля 48
 - соединения хоста 15
- конфликт
 - системные ресурсы 120
- концентратор соединений 105
 - DB2 Connect 112
- концентраторы соединений
 - диспетчер 107
 - затраты 107
 - логические агенты 107
 - обзор 105
 - ограничения 107
 - параметр конфигурации MAX_COORDAGENTS 107
 - параметр конфигурации MAXAGENTS 107
 - параметр конфигурации NUM_INITAGENTS 107
 - параметр конфигурации NUM_POOLAGENTS 107
 - параметры конфигурации 107
 - Поддержка транзакций XA 107
 - примеры 107
 - рабочие агенты 107
 - реализация 107
 - создание пулов 107
 - сравнение с пулами соединений 112
 - кэшированный список адресов 114

M

- маркеры
 - SQLCODE 63
- Мастер по конфигурированию обновления нескольких баз данных 56
- мастера
 - Multisite Update 56
- менеджер точек синхронизации (SPM)
 - параметры по умолчанию 57
 - сценарии 57
- менеджер транзакций WebSphere MQ
 - DB2 Connect 112
- многоузловое изменение
 - активизация 55
 - менеджер точек синхронизации 57
 - проверка 56
 - распределенная единица работы (distributed unit of work, DUOW) 55
 - Центр управления 56
- модель распределенной обработки транзакций (DTP)
 - X/Open 26
- монитор
 - производительность 69
 - соединения 69
- монитор работоспособности
 - DB2 for z/OS 76
- Монитор работоспособности DB2 for z/OS
 - запуск, остановка, обновление 77
 - обзор 76
 - просмотр объектов оповещения 83
 - просмотр сводок оповещений 81
 - просмотр, передача на выполнение, сохранение рекомендованных действий 79
- мониторинг
 - монитор производительности Windows 69
 - соединения
 - сервер DB2 Connect 69
- мониторы транзакций
 - OLTP 26
 - Tuxedo 26
 - многоузловое изменение 55
 - примеры 26
 - транзакции 26
 - характеристики использования 26

N

- Наглядное объяснение
 - учебный материал 163
- настройка
 - DB2 Connect 93
 - DB2 for OS/390 and z/OS 121
 - каталоги, рабочие листы для каталогов 37
 - параметр DIRCACHE 117
 - параметр MAXAGENTS 117
 - параметр MAXDARI 117
 - параметр NUMDB 117
 - параметр RQRIOBLK 117
 - производительность
 - база данных 119
 - сеть 119
- настройка производительности
 - DB2 Connect 93
- номер внешней последовательности 72
- номер последовательности клиента 72

O

- об этой книге v
 - обзор документации 157
 - обновление
 - Монитор работоспособности DB2 for z/OS 77
 - объект EXTNAM 140
 - объект SRVNAM 140
 - объекты оповещения
 - просмотр 83
 - ограничения
 - концентратор соединений 107
 - оператор COMMIT
 - статически связанный 101
 - оператор DESCRIBE 101
 - оператор EXECUTE IMMEDIATE
 - разработка программ 101
 - оператор GRANT
 - защита 49
 - оператор PREPARE
 - влияние на производительность 101
 - при разработке программ 101
 - оператор REVOKE
 - защита 49
 - оператор ROLLBACK
 - статически связанный 101
 - оператор SELECT
 - FOR FETCH ONLY 101
 - изменяемый 101
 - при разработке программ 101
 - оператор SET CURRENT PACKAGESET 48
 - операторы
 - COMMIT 101
 - DESCRIBE 101
 - EXECUTE IMMEDIATE 101
 - FOR FETCH ONLY 101
 - PREPARE 101
 - ROLLBACK
 - разработка программ 101
 - SELECT 101
 - Операторы SQL
 - вывод справки 160
 - операционные системы Windows
 - Монитор производительности 69
 - оповещения монитора работоспособности 76
 - опция мониторинга SHOW DETAIL 72
 - остановка
 - Монитор работоспособности DB2 for z/OS 77
 - отказы соединений
 - автоматическое перенаправление клиента 91
 - отношения доверия
 - доверенные контексты и доверенные соединения 41
 - отображение
 - SQLCODE 63
 - оптимизация 63
 - параметр NOMAP 63
 - ошибки
 - устранение неисправностей 131
- ## P
- пакеты
 - созданные на сервере баз данных хоста или System i 51
 - параметр AGENTPRI 117
 - параметр D (disconnect) 33
 - параметр DIRCACHE 117
 - параметр EXTRA BLOCKS SRV 122

- параметр INTERRUPT_ENABLED (disconnect) 33
- параметр LOCALDATE 33
- параметр MAX_COORDAGENTS 105, 107
- параметр MAXAGENTS 107, 117
- параметр NOMAP 33, 63
- параметр NUM_INITAGENTS 105, 107
- параметр NUM_POOLAGENTS 105, 107
- параметр PRDID 140
- параметр RQRIOBLK
 - настройка 117
- параметр конфигурации MAXDARI 117
- параметр конфигурации поддержка кэша каталогов
 - Настройка DB2 Connect 117
- параметры
 - AGENTPRI 117
 - BIDI 33
 - D (disconnect) 33
 - DIRCACHE 117
 - EXTRA BLOCKS SRV 122
 - INTERRUPT_ENABLED (disconnect) 33
 - LOCALDATE 33
 - MAX_COORDAGENTS 107
 - MAXAGENTS 107, 117
 - MAXDARI 117
 - NOMAP 33
 - NUM_INITAGENTS 107
 - NUM_POOLAGENTS 107
 - NUMDB 117
 - PRDID 140
 - RQRIOBLK 117
 - SYSPLEX 33
 - запятая в строке 33
 - параметры каталога 37
 - строки 38
- параметры BSDS (набора данных загрузки)
 - z/OS и OS/390 31
- параметры конфигурации
 - MAX_COORDAGENTS 105
 - NUM_INITAGENTS 105
 - NUM_POOLAGENTS 105
 - TCP_KEEPALIVE 91
- параметры набора данных загрузки (BSDS)
 - Z/OS и OS/390 31
- пароли
 - поддержка изменения (OS/390 и z/OS) 48
- перемещение данных
 - использование DB2 Connect 59
- печатные книги
 - заказ 160
- поддержка CCSID с двумя направлениями письма
 - параметр BIDI 33
- полномочия
 - связывание 51
- полномочия CREATE IN COLLECTION NULLID 51
- положения и условия
 - использование публикаций 164
- предикаты
 - производительность алгоритма 101
- преобразования
 - данные хоста 124
- привилегия BINDADD
 - полномочия для связывания 51
- прикладные программы
 - Web
 - использование DB2 Connect 19
 - производительность 101
 - связывание 51
- прикладные программы (продолжение)
 - составной SQL 101
 - хранимые процедуры 101
- приложение клиента
 - ID 72
- примеры
 - концентраторы XA 107
 - концентраторы соединений 107
- проверка
 - многоузловое изменение 56
- программирование CGI (Common Gateway Interface)
 - ограничения 19
 - преимущества 19
- программирование Common Gateway Interface (CGI)
 - ограничения 19
 - преимущества 19
- программы Web
 - DB2 Connect Enterprise Edition 19
 - хранимые процедуры 22
- продукты сервера DB2 Connect
 - описание продукта 3
- проектирование программ 101
- производительность
 - DB2 for OS/390 and z/OS 121
 - измерение 95
 - концентратор соединений 112
 - настройка 121
 - оптимизация доступа ODBC 99
 - особенности SQL 101
 - повышение скорости передачи данных 121
 - показатели 95
 - понятия 95
 - поток данных 95
 - прикладные программы
 - блокирование данных 101
 - группировка требований 101
 - логика предикатов 101
 - проектирование 101
 - составной SQL 101
 - хранимые процедуры 101
 - процессор командной строки 101
 - пулы соединений 112
 - сетевое аппаратное обеспечение 124
 - сетевые средства 95
 - системные ресурсы 120
 - средства 95
 - узкие места 95
 - устранение неисправностей 121
- пропускная способность
 - транзакции 95
- протокол аутентификации Kerberos
 - DB2 Connect 46
 - OS/390 и z/OS 47
- процессор
 - средства использования 95
- процессор командной строки (CLP)
 - Операторы SQL 7
 - производительность 101
- прямой доступ к базам данных
 - DB2 Connect PE 15
- пулы соединений 105
 - обзор 105
 - сравнение с концентратором соединений 112

Р

- рабочие листы
 - настройка каталога 37
- размер блока 117
- размер блока подкачки 117
- разработка прикладных программ 101
 - ODBC 15
 - клиент DB2 AD 15
- распределенная единица работы
 - двухфазное принятие 55
 - многоузловое изменение 55
 - поддерживаемые серверы 55
 - характеристики 9
- распределенное требование
 - базы данных объединения 12
 - компенсация 12
 - определение 12
 - поддержка 12
 - прозрачность положений 12
- распределенное управление данными (DDM) 10, 139
- реквестеры прикладных программ
 - определение DRDA 10
 - параметры 37
- рекомендуемые действия
 - просмотр, передача на выполнение, сохранение 79

С

- с плавающей точкой
 - тип данных 124
- сводка оповещений
 - просмотр 81
- связывание
 - имена пакетов 51
 - пакеты 51
 - полномочия
 - маркеры параметров со смещением 51
 - утилиты и программы 51
- серверы
 - прикладных программ
 - DB2 Connect EE 23
- серверы прикладных программ
 - 2-слойная модель 23
 - 3-слойная модель 23
 - DB2 Connect ESE 23
 - внедрение 23
 - конфигурирование 23
 - обзор 23
 - определение DRDA 10
 - поддержка в DB2 Connect 23
 - расширенные клиенты 23
- сеть
 - настройка 119
 - скорость передачи данных 124
 - средства производительности 95
- символическое имя назначения 37
 - учет регистра 31
- системные ресурсы
 - конфликт 120
- системный каталог базы данных
 - алиас базы данных 31
 - аутентификация 31
 - значения 31
 - имя базы данных 31
 - имя узла 31
 - перед изменением 31

- системный монитор базы данных
 - описание 7
 - удаленные клиенты 69
- служебный журнал первого отказа 133
- соединения
 - DB2 Connect Enterprise Edition 18
 - концентраторы, смотрите концентраторы соединений 107
 - направление на хост 15
 - напрямую с базой данных хоста или System i 17
 - повторное установление
 - DB2 Connect Enterprise Edition 18
 - направление на хост 15
 - серверы, DB2 Connect Enterprise Edition 18
 - создание пулов
 - концентраторы соединений 107
 - обзор 105
 - преимущества 107
- сообщение-ответ об окончании единицы работы (ENDUOWRM) 140
- сообщения об ошибках
 - DB2 Connect 151
- составной SQL
 - NOT ATOMIC 101
- составной оператор SQL ATOMIC
 - не поддерживается в DB2 Connect 101
- составной оператор SQL NOT ATOMIC
 - разработка программ 101
- состояние системы
 - Команда GET SNAPSHOT 70
- список связывания 51
- справка
 - вывод 161
 - для операторов SQL 160
- средства
 - диагностика 133
 - использование памяти 95
 - использование процессора 95
 - производительность 95
- средства диагностики
 - диагностика ошибок 133
- средства использования памяти 95
- ссылки
 - задание нескольких записей базы данных 38
- статический SQL
 - производительность 101
 - эффекты обработки 6
- сценарии
 - защита TCP/IP 49

Т

- тип аутентификации CLIENT
 - Информация о DB2 Connect 46
- тип аутентификации SERVER 46
- тип аутентификации SERVER_ENCRYPT 46
- тип данных CHAR
 - описание 124
- тип данных INTEGER 124
- тип данных VARCHAR
 - описание 124
- тип защиты PROGRAM 49
- тип защиты SAME 49
- типы данных
 - CHAR 124
 - INTEGER 124
 - VARCHAR 124
 - зонный десятичный 124

- типы данных *(продолжение)*
 - преобразование
 - влияние на производительность 124
 - с плавающей точкой 124
 - символьные данные 124
 - упакованный десятичный 124
- типы защиты NONE 49
- типы символьных данных 124
- транзакции
 - DB2 Connect Enterprise Edition 26
 - двухфазное принятие 9
 - единица работы 9
 - многоузловое изменение 9, 55
 - мониторы транзакций 26
 - поддержка 58
 - пропускная способность 95
 - распределенные
 - поддерживаемые серверы 55
 - распределенные прикладные программы XA 58
- трассировки
 - DRDA
 - интерпретация 139
 - выходной файл 139, 140
 - данных, передаваемых между DB2 Connect и сервером 139
 - информация буфера для трассировки DRDA 146
 - примеры файлов вывода 142
- требования баз данных
 - группировка
 - производительность 101

У

- удаленная единица работы
 - обзор 11
 - пример 11
 - характеристики 11
- узкие места
 - производительность 95
 - транзакции 95
- узлы
 - имя 31, 37
 - каталог 31
- упакованный десятичный тип данных 124
- управление соединениями 105
- условие FOR FETCH ONLY
 - оператор SELECT 101
- усовершенствования в данном выпуске 3
- устранение неисправностей
 - DB2 Connect 151
 - возможности трассировки 135
 - DRDA 142, 146
 - обзор 131
 - оперативная информация 164
 - производительность 121
 - сбор информации 131
 - соединение 131, 132
 - учебные материалы 164
- утилита db2drdat
 - выходной файл 139
- утилита ddcstrc
 - выходной файл 140
- утилита ps (состояние процесса) 133, 140
- утилита импорта
 - передача данных между хостом и рабочей станцией 59
- утилита просмотра состояния процесса 133, 140
- утилита трассировки 139

- утилита трассировки DB2 (db2trc)
 - выгрузка вывода трассировки 136
- утилита управления
 - DB2 Connect 7
- утилита управления доступом к ресурсам (resource access control facility, RACF)
 - защита 49
- утилита экспорта
 - передача данных между хостом и рабочей станцией 59
- утилиты
 - db2drdat 139
 - ddcspkgn 51
 - ps (состояние процесса) 133, 140
 - связывание 51
 - системный монитор базы данных 7
 - состояние процесса 140
 - трассировка 139
 - управление, DB2 Connect 7
- учебные материалы
 - диагностика и устранение неисправностей 164
 - Наглядное объяснение 163

Ф

- файл dcs1ari.map 63
- файл dcs1dsn.map 63
- файл dcs1qsq.map 63
- файл ddc400.lst 51
- файл ddcsmvs.lst 51
- файл ddcsvm.lst 51
- файл ddcsvse.lst 51
- файлы дампа
 - диагностика ошибок 133

Х

- хранимые процедуры
 - обзор 22

Ц

- Центр управления
 - многоузловое изменение 56

Ч

- часовые пояса 33

Э

- элемент мониторинга имени приложения 72
- эскейп-символы 38



Напечатано в Дании

SC89-1005-00



Spine information:

DB2 Connect, Версия 9.5 версия 9 выпуск 5

DB2 Connect. Руководство пользователя

