



Утилиты командной строки для СУБД Firebird

Norman Dunbar

Утилиты командной строки для СУБД Firebird

Norman Dunbar

Перевод на русский язык: Василий Овчинников, Александр Карпейкин, Сергей Ковалёв

Корректор перевода: Павел Меньщиков

Опубликовано 2004

Copyright © 2004 Norman Dunbar

Содержание

1. Введение	1
2. GSEC - Утилита работы с базой данных безопасности	3
Введение	3
Опции командной строки	4
Команды GSEC	6
Интерактивный режим	7
Просмотр информации о пользователях	8
Добавление новых пользователей	8
Удаление существующих пользователей	9
Изменение информации об уже введенных пользователях	9
Вывод подсказки	10
Вывод информации о версии	10
Пакетный режим	10
Просмотр информации о пользователях	11
Добавление новых пользователей	11
Удаление существующих пользователей	11
Изменение информации об уже введенных пользователях	12
Вывод информации о версии	12
Удаленный запуск GSEC	13
Работаем с Firebird 2	13
Особенности работы с GSEC	14
Обычные и привилегированные пользователи	14
Отличия интерактивного и пакетного режимов	14
Коды завершения в пакетном режиме	15
Ошибки в пакетном режиме переводят GSEC в интерактивный режим	15
3. GSPLIT - Фильтр для разбиения на фрагменты	16
Введение	16
Опции командной строки GSPLIT	17
Параметры опций GSPLIT	18
Разбиение резервных копий	18
Объединение фрагментов резервной копии	19
4. Скрипты для Firebird в ОС Linux/Unix	21
Введение	21
Изменение пароля SYSDBA	21
Создание псевдонима для базы данных	22
Параметры скрипта createAliasDB.sh	23
Помощь с файлами makefile	23
Опции fb_config	23
Изменение пользователя для запуска Firebird	24
Параметры скрипта SSchangeRunUser.sh	25
Восстановление прав root как владельца Firebird	25
Запуск приложений Embedded или ClientServer	26
5. Планы на будущее	27
Приложение А: Лицензионное соглашение	28

Введение

Перевод главы на русский язык: Сергей Ковалёв

Дистрибутивный комплект СУБД Firebird 1.5 устанавливает несколько полезных утилит, которые помогают в работе вашего сервера Firebird и баз данных. В этой книге содержится обзор различных утилит, некоторые из них более полезны, другие - менее.

Утилиты можно найти в подкаталоге `/bin` (или в подкаталоге `\bin` для ОС Windows) каталога, в котором была установлена СУБД Firebird. В ОС Linux утилиты можно найти в подкаталоге `/opt/firebird/bin`, в то время, как в ОС Windows расположение по умолчанию `c:\program files\firebird\firebird_1_5\bin`.

- **FB_LOCK_PRINT** - утилита, которая выводит детализацию внутренней страницы блокировок базы данных.
- **GBAK** - утилита резервного копирования и восстановления. Она также позволяет изменять некоторые внутренние параметры базы данных.
- **GDEF** - утилита для метаданных, которая удалена из СУБД Interbase 4.0, и была возвращена в версии 6 (с открытым Кодом). GDEF, скорее всего, лишняя утилита.
- **GFIX** позволяет предпринять попытку починить поврежденную базу данных, открыть (on-line) и остановить (shutdown) базу данных для соединений, решить проблемы "зависших" между несколькими базами данных транзакций (in-limbo), изменить количество страниц буфера, и так далее.
- **GPRE** - это препроцессор, который конвертирует исходный код, написанный на некоторых языках и содержащий встроенный псевдокод SQL, в корректный отформатированный вызов функций Firebird API.
- **GSEC** - это утилита управления базой данных безопасности. Она дает администратору баз данных (или любому привилегированному пользователю) возможность управлять учетными записями пользователей сервера Firebird. Используя разные ключи, вы сможете добавлять, изменять или удалять информацию о пользователях в базе данных безопасности.
- **GSPLIT** - это фильтр, который позволяет преодолевать ограничения на максимальный размер файла, существующие в некоторых операционных системах, при создании резервной копии очень большой базы данных. Эта утилита поставляется только для ОС Windows и, к сожалению, кажется она не работает. К счастью, GBAK позволяет разбивать файлы резервной копии на несколько частей, так что GSPLIT не требуется. В системах ОС Unix существует подходящие утилиты из операционной системы, которые могут быть использованы вместо GSPLIT, если это необходимо.
- **GSTAT** дает возможность администратору СУБД Firebird собирать статистику об общем состоянии и использовании разных частей базы данных.
- **ISQL** - это интерактивный инструмент, который позволяет выполнять запросы к базу данных Firebird. Это консольное приложение (как и многие другие утилиты), и оно есть во всех дистрибутивах СУБД Firebird. ISQL - это лучший способ опробовать ваши скрипты и команды в первый раз.

- QLI - это оригинальный Query Language Interpreter, который был удален из СУБД Interbase 4.0, но был возвращен в версии 6.0, поскольку было принято решение о распространении СУБД Interbase в открытых кодах.
- Есть также несколько скриптов для оболочки, устанавливаемых в ОС Linux и других системах Unix.

Замечание

Работа над этой книгой продолжается. Каждая глава описывает отдельную утилиту, и как только работа над главой завершается (к моему удовольствию), я помещаю ее в репозиторий CVS, где она становится доступной для скачивания. Таким образом, будет медленное и постепенное строительство полезнейшего руководства пользователя.

GSEC - Утилита работы с базой данных безопасности

Перевод главы на русский язык: Василий Овчинников

Введение

GSEC - это утилита для работы с базой данных безопасности (содержащей информацию о пользователях СУБД). Она позволяет системному администратору (или другому привилегированному пользователю) управлять учетными записями пользователей для различных баз данных СУБД Firebird. Используя различные опции, можно добавлять, изменять или удалять учетные записи пользователей из базы данных безопасности.

Замечание

Для некоторых операционных систем существует возможность запретить пользователям запуск GSEC, даже если им известен пароль пользователя SYSDBA. Системные администраторы таких систем могут с помощью установки прав файловой системы запретить выполнение некоторых утилит и программ для соблюдения требований безопасности.

Информация обо всех пользователях баз данных СУБД Firebird хранится в общей базе данных безопасности. Это обычная база данных СУБД Firebird, которая располагается на сервере, и называется `security.fdb`. Начиная с СУБД Firebird 2 она называется `security2.fdb`. По умолчанию этот файл располагается в директории:

- `C:\Program Files\Firebird\Firebird_x_x` для ОС Windows (где `x_x` - версия Firebird, например `1_5`).
- `/opt/firebird` для ОС Linux и других Unix-систем.

В базе данных безопасности присутствуют две таблицы - `users` и `host_info`. Таблица `host_info` пустая, а таблица `users` хранит сведения о каждом пользователе, имеющем право доступа к любой базе данных СУБД Firebird. В СУБД Firebird 2 структура базы данных безопасности изменена, о чем будет сказано ниже. Необходимо также сказать, что для разграничения полномочий пользователей при работе с базами данных можно использовать роли и привилегии.

Утилита GSEC позволяет менять данные в таблице `users` базы данных безопасности. С ее помощью возможно добавление, изменение и удаление пользователей системы. Не все поля таблицы `users` могут быть

показаны, хотя изменить можно любые из них. Например, столбец password (пароль) никогда не показывается утилитой GSEC, но вы его можете изменить.

Как и большинство утилит командной строки из поставки СУБД Firebird, GSEC может быть запущена как в интерактивном, так и в пакетном режиме, и может отображать подсказки с перечислением всех опций, о которых будет сказано ниже.

В этой главе описаны:

- Опции командной строки GSEC.
- Команды и параметры опций GSEC.
- Выполнение GSEC в интерактивном и пакетном режиме, в любом из которых вы можете:
 - просматривать информацию о пользователях;
 - изменять информацию о пользователях;
 - добавлять новых пользователей;
 - удалять существующих пользователей.
- Использование GSEC для управления удаленной базой данных безопасности.
- Особенности работы с GSEC.

Опции командной строки

Вне зависимости от режима выполнения GSEC, некоторые опции могут быть заданы в командной строке. Это:

- **-user <имя>**

Позволяет указать пользователя SYSDBA, если база данных безопасности будет подвергаться изменениям, или обычного пользователя, если предстоит только просмотр информации. Нет необходимости использовать эту опцию, если корректно определены и содержат правильные значения переменные окружения операционной системы ISC_USER и ISC_PASSWORD.

- **-password <пароль>**

Позволяет указать пароль пользователя, указанного выше. Нет необходимости использовать эту опцию, если корректно определены и содержат правильные значения переменные окружения операционной системы ISC_USER и ISC_PASSWORD.

- **-role <имя_SQL-роли>**

Позволяет указать SQL-роль, чьи права будет использовать указанный пользователь.

- **-database <имя_БД_безопасности>**

Вы можете указать полное имя файла базы данных безопасности, что делает возможным удаленное администрирование пользователей СУБД. Если полное имя файла содержит пробелы, заключите все имя файла в двойные кавычки.

- **-z**

Показывает номер версии утилиты GSEC .

- **-help**

Отображает следующую подсказку на экране:

```
gsec utility - maintains user password database

command line usage:
  gsec [ <options> ... ] <command> [ <parameter> ... ]

interactive usage:
  gsec [ <options> ... ]
  GSEC>
  <command> [ <parameter> ... ]

available options:
  -user <database administrator name>
  -password <database administrator password>
  -role <database administrator SQL role name>
  -database <security database>
  -z

available commands:
adding a new user:
  add <name> [ <parameter> ... ]
deleting a current user:
  delete <name>
displaying all users:
  display
displaying one user:
  display <name>
modifying a user's parameters:
  modify <name> <parameter> [ <parameter> ... ]
help:
  ? (interactive only)
  help
displaying version number:
  z (interactive only)
quit interactive session:
  quit (interactive only)

available parameters:
  -pw <password>
  -uid <uid>
  -gid <uid>
  -fname <firstname>
  -mname <middlename>
  -lname <lastname>
```

Перевод:

```
gsec - утилита управление базой данных паролей пользователей

использование в режиме командной строки:
  gsec [ <опция> ... ] <команда> [ <параметр> ... ]

интерактивный режим:
  gsec [ <опции> ... ]
  GSEC>
  <команда> [ <переаметр> ... ]
```

возможные опции:

```
-user <имя администратора базы данных>  
-password <пароль администратора базы данных>  
-role <имя SQL-роли администратора базы данных>  
-database <база данных паролей>  
-z
```

возможные команды:

добавление нового пользователя:

```
add <имя> [ <параметр> ... ]
```

удаление пользователя:

```
delete <имя>
```

вывод информации обо всех зарегистрированных пользователях:

```
display
```

вывод информации только об одном из пользователей:

```
display <имя>
```

изменение информации о пользователе:

```
modify <имя> <параметр> [ <параметр> ... ]
```

помощь:

```
? (только в интерактивном режиме)
```

```
help
```

вывести информацию о версии:

```
z (только в интерактивном режиме)
```

выход из интерактивного режима:

```
quit (только в интерактивном режиме)
```

возможные параметры:

```
-pw <пароль>
```

```
-uid <uid>
```

```
-gid <uid>
```

```
-fname <имя>
```

```
-mname <отчество>
```

```
-lname <фамилия>
```

Команды GSEC

После общих опций указываются команды, которые вам необходимо выполнить. Нижеследующие команды применимы и к интерактивному режиму, и к режиму командной строки. Но в первом случае начальный символ минуса не требуется.

- **-add <имя> [<параметр> ...]**

Эта команда добавляет в базу данных безопасности нового пользователя. В качестве необязательных параметров можно указать имя, отчество, фамилию и пароль нового пользователя. Однако, эти дополнительные данные можно изменить позже командой **modify**.

- **-delete <имя>**

Эта команда удаляет указанного пользователя из базы данных безопасности. Вся информация о пользователе также удаляется и не может быть восстановлена, только если вы сами позже не добавите этого пользователя снова.

- **-display** [<имя>]

Эта команда отображает данные указанного пользователя или всех пользователей, если имя не указано. Пароль никогда не отображается.

- **-modify** <имя> <параметр> [<параметр> ...]

Эта команда позволяет изменить данные пользователя.

Параметр <имя> - это имя, которое пользователь будет указывать при подключении к базе данных Firebird. Некоторые из вышеописанных команд позволяют дополнительно указать один или более следующих параметров:

- **-pw** <пароль>

Этот параметр позволяет вам указать новый пароль для пользователя. Максимальная длина пароля равна 8 символам. Если при работе с GSEC или при подключении к базе данных будет указан пароль большей длины, то символы далее восьмого будут игнорироваться.

- **-uid** <uid>

- **-gid** <gid>

-uid и **-gid** используются на некоторых POSIX-системах для указания идентификаторов `userid` (идентификатор пользователя) и `groupid` (идентификатор группы) Unix, как они указаны в конфигурационных файлах `/etc/passwd` и `/etc/group`. Если эти параметры не указаны, то они считаются равными нулю.

- **-fname** [<имя>]

Этот параметр позволяет задать настоящее имя пользователя. Его можно применять для идентификации пользователя в дополнение к имени для подключения, так как имя для подключения к базе может быть сокращенным или являться аббревиатурой. Вы можете удалить информацию о настоящем имени пользователя, указав этот параметр, но не определяя имя.

- **-mname** [<отчество>]

Этот параметр позволяет задать отчество пользователя. Его можно применять для идентификации пользователя в дополнение к имени для подключения, так как имя для подключения к базе может быть сокращенным или являться аббревиатурой. Вы можете удалить информацию об отчестве пользователя, указав этот параметр, но не определяя отчество.

- **-lname** [<фамилия>]

Этот параметр позволяет задать фамилию пользователя. Его можно применять для идентификации пользователя в дополнение к имени для подключения, так как имя для подключения к базе может быть сокращенным или являться аббревиатурой. Вы можете удалить информацию о фамилии пользователя, указав этот параметр, но не определяя фамилию.

Интерактивный режим

Для запуска утилиты GSEC в интерактивном режиме используйте следующую командную строку:

```
C:\>gsec -user sysdba -password masterkey
GSEC>
```

Приглашение GSEC> указывает на то, что ожидается ввод команды. Опции -user и -password определяют пользователя, от имени которого будут вноситься изменения в базу данных безопасности. Не стоит объяснять, что для внесения изменений пользователем должен быть только sysdba. Остальные пользователи могут иметь к базе данных безопасности только доступ для чтения.

Для выхода из интерактивного режима утилиты GSEC используйте команду quit:

```
GSEC> quit
C:\>
```

Далее рассмотрим, как выполняются различные команды в интерактивном режиме. Подразумевается, что вы запустили утилиту как пользователь sysdba.

Просмотр информации о пользователях

Ниже представлена команда вывода информации о пользователях и результат ее выполнения :

```
GSEC> display
      user name                uid  gid  full name
-----
SYSDBA                        0    0
NORMAN                        0    0    Norman Dunbar
EPOCMAN                       0    0    Benoit Gilles Mascia
GSEC>
```

Для вывода информации о конкретном пользователе используйте команду **display**, указав имя пользователя в качестве параметра.

```
GSEC> display epcman
      user name                uid  gid  full name
-----
EPOCMAN                       0    0    Benoit Gilles Mascia
GSEC>
```

Если в качестве параметра вы укажете имя несуществующего пользователя, то в результате ничего не отобразится, и утилита продолжит работу в интерактивном режиме.

```
GSEC> display alison
GSEC>
```

Добавление новых пользователей

При добавлении пользователя в интерактивном режиме в подтверждение успешности операции ничего не сообщается. После этого нужно использовать команду **display** или **display <имя>** для того, чтобы убедиться, что пользователь добавлен успешно.

```
GSEC> add newuser -pw newuser -fname New -lname User
```

```
GSEC>
```

```
GSEC> display newuser
      user name                uid  gid  full name
-----
NEWUSER                0    0    New User
GSEC>
```

Удаление существующих пользователей

При удалении пользователя в интерактивном режиме в подтверждение успешности операции ничего не сообщается. После этого нужно использовать команду **display** или **display <имя>** для того, чтобы убедиться, что пользователь удален успешно.

```
GSEC> delete newuser
GSEC>
```

```
GSEC> display
      user name                uid  gid  full name
-----
SYSDBA                0    0
NORMAN                0    0    Norman Dunbar
EPOCMAN              0    0    Benoit Gilles Mascia
GSEC>
```

При попытке удаления несуществующего пользователя будет сообщено об ошибке.

```
GSEC> delete newuser
record not found for user: NEWUSER

C:\>
```

Изменение информации об уже введенных пользователях

Для уже существующего пользователя в базе данных безопасности можно изменить пароль и его настоящее имя, отчество и фамилию. Никакого подтверждения об успешности выполнения операции не выводится. После этого нужно использовать команду **display** или **display <имя>** для того, чтобы убедиться, что все выполнено успешно.

```
GSEC> modify norman -pw newpassword
GSEC>
```

```
GSEC> modify norman -mname MiddleName -fname Fred
GSEC>
```

```
GSEC> display norman
      user name                uid  gid  full name
-----
NORMAN                0    0    Fred MiddleName Dunbar
GSEC>
```

Если вы хотите удалить один или несколько атрибутов из записи о пользователе, то просто не определяйте их новых значений после соответствующего параметра.

```
GSEC> modify norman -mname -fname -lname
```

```
GSEC> display norman
      user name                uid   gid   full name
-----
NORMAN                        0     0
```

Теперь пользователь NORMAN известен как "человек без имени", почти как Клинт Иствуд!

Вывод подсказки

Команда **help** в интерактивном режиме отображает такую же подсказку на экране, о которой говорилось при обсуждении опций командной строки.

Вывод информации о версии

Версию утилиты GSEC можно узнать, используя команду **z**.

```
GSEC> z
gsec version WI-V1.5.0.4306 Firebird 1.5
GSEC>
```

Пакетный режим

Замечание

В нижеследующем описании команд пакетного режима подразумевается, что переменные окружения `ISC_USER` и `ISC_PASSWORD` определены и имеют правильные значения. Это позволяет запускать GSEC без необходимости всегда указывать в командной строке опции **-user** и **-password**. В свою очередь это уменьшает длину командной строки, позволяя после преобразования данного XML-файла в pdf уместить их на ширине страницы формата A4.

Не храните правильные значения в этих переменных постоянно, поскольку это небезопасно!

Внимание

Вам может показаться, что вы можете оценить результат выполнения утилиты по значению переменной окружения `%ERRORLEVEL%` в ОС Windows или `$?` в различных вариантах Unix. Это не так. Возвращаемый результат выполнения всегда равен нулю до СУБД Firebird 2.0. Начиная с СУБД Firebird 2.0 GSEC может возвращать ненулевой код выполнения, сигнализирующий о неудачном выполнении.

Для запуска GSEC в пакетном режиме используйте следующую командную строку:

```
gsec [ <опции> ... ] <команда> [ <параметр> ... ]
```

Просмотр информации о пользователях

Ниже представлена команда вывода информации о пользователях и результат ее выполнения:

```
C:\>gsec -display
user name                uid  gid  full name
-----
SYSDBA                   0    0
NORMAN                   0    0    Norman Dunbar
EPOCMAN                  0    0    Benoit Gilles Mascia
```

Для вывода информации о конкретном пользователе используйте команду **display**, указав имя пользователя в качестве параметра.

```
C:\>gsec -display epocman
user name                uid  gid  full name
-----
EPOCMAN                  0    0    Benoit Gilles Mascia
```

Добавление новых пользователей

При добавлении пользователя в пакетном режиме в подтверждение успешности операции ничего не сообщается. После этого нужно использовать команду **display** или **display <имя>** для того, чтобы убедиться, что пользователь добавлен успешно.

```
C:\>gsec -add newuser -pw newuser -fname New -lname User
```

```
C:\>gsec -display
user name                uid  gid  full name
-----
SYSDBA                   0    0
NORMAN                   0    0    Norman Dunbar
NEWUSER                  0    0    New User
EPOCMAN                  0    0    Benoit Gilles Mascia
```

Удаление существующих пользователей

При удалении пользователя в пакетном режиме в подтверждение успешности операции ничего не сообщается. После этого нужно использовать команду **display** или **display <имя>** для того, чтобы убедиться, что пользователь удален успешно.

```
C:\>gsec -delete newuser
```

```
C:\>gsec -display
user name                uid  gid  full name
-----
SYSDBA                   0    0
NORMAN                   0    0    Norman Dunbar
```

```
EPOCMAN          0      0      Benoit Gilles Mascia
```

Изменение информации об уже введенных пользователях

Для уже существующего пользователя в базе данных безопасности можно изменить пароль и его настоящие имя, отчество и фамилию.

```
C:\>gsec -modify norman -pw newpassword
```

```
C:\>gsec -modify norman -mname MiddleName -fname Fred
```

```
C:\>gsec -display
  user name                uid  gid  full name
-----
SYSDBA                    0    0
NORMAN                    0    0    Fred MiddleName Dunbar
EPOCMAN                   0    0    Benoit Gilles Mascia
```

Если вы хотите удалить один или несколько атрибутов из записи о пользователе, то просто не определяйте их новых значений после соответствующего параметра.

```
C:\>gsec -modify norman -mname -fname -lname
```

```
C:\>gsec -display
  user name                uid  gid  full name
-----
SYSDBA                    0    0
NORMAN                    0    0
EPOCMAN                   0    0    Benoit Gilles Mascia
```

Теперь пользователя NORMAN никто не узнает :o)

Вывод информации о версии

Версию утилиты GSEC можно узнать используя команду **-z**. Учтите, что эта команда после выполнения переводит вас в интерактивный режим и не возвращает из него. Она не завершается так, как это делают остальные команды пакетного режима. Для выхода из интерактивного режима используйте команду **quit**. Но обходной путь существует, и он показан ниже. Сначала посмотрим на проблему - вот она:

```
C:\>gsec -z
gsec version WI-V1.5.0.4306 Firebird 1.5
GSEC>
```

Решение проблемы в создании маленького файла, содержащего команду **quit** и указании GSEC читать этот файл, когда ему потребуется ввод пользователя:

```
C:\>copy con fred
quit
^Z
```

```
1 file(s) copied.
```

```
C:\>gsec -z < fred
gsec version WI-V1.5.0.4306 Firebird 1.5
GSEC>
C:\>
```

Такое решение можно использовать для любых команд, которые переходят в интерактивный режим в то время, когда вы думаете, что они выполняются в пакетном. По перенаправлению ввода на командный файл, GSEC считает строку текста из него в тот момент, когда ему понадобится ввод пользователя с клавиатуры. А заставляя его читать команду **quit**, вы завершите работу в интерактивном режиме.

Замечание

Команда `-z` не нуждается в опциях `-user` и `-password`. Если вы их не укажете, то в этом случае она отображает информацию о версии, а потом известит вас, что не указан пользователь и пароль. Это сообщение вы можете не принимать во внимание.

Удаленный запуск GSEC

GSEC можно использовать для управления базой данных безопасности на удаленном сервере. Для этого вы должны указать в командной строке имя удаленной базы данных, как показано в примере, где утилита GSEC для ОС Windows XP устанавливает соединение с Linux-сервером Ganymede и позволяет управлять информацией о пользователях на сервере Linux.

```
C:\>gsec -database ganymede:/opt/firebird/security.fdb
      -user sysdba -password masterkey
GSEC>
```

Замечание

В приведенном примере команда напечатана в двух строках. Это сделано для исключения проблем при переводе документа в формат PDF. На самом деле команда должна вводиться одной строкой.

Подключившись к удаленной базе данных безопасности, вы можете вносить изменения в интерактивном или в пакетном режиме, как было описано выше.

Работаем с Firebird 2

В СУБД Firebird 2 изменен формат базы данных безопасности. Ей дано новое имя `security2.fdb`. Для идентификации пользователей она содержит новую таблицу `RDB$USERS`, которая хранит пароли, хэшированные по новым правилам. Для замещения таблицы `USERS` в новой базе есть одноименное представление (`view`), позволяющее пользователям менять их собственные пароли. Версия утилиты GSEC, поставляемая с СУБД Firebird 2, может быть использована для работы с базой данных безопасности предыдущих версий СУБД Firebird. Однако, по причине смены формата базы данных безопасности предыдущие версии GSEC не смогут работать с базой данных безопасности СУБД Firebird 2. Рекомендуется также ознакомиться с Release Notes для СУБД Firebird 2 на предмет изменений, коснувшихся системы безопасности и утилиты GSEC.

Особенности работы с GSEC

Ниже представлены некоторые особенности, выявленные при практическом использовании утилиты GSEC. Некоторые из них упомянуты выше, а некоторые, быть может, нет. Просматривая все это в одном месте, вы, скорее всего, сможете определить причину возникновения вашей конкретной проблемы при работе с GSEC.

Обычные и привилегированные пользователи

Вносить изменения в базу данных безопасности может только пользователь sysdba. Обычный пользователь может запустить GSEC только для просмотра информации. Начиная с СУБД Firebird 2 пользователь может изменить любую информацию о себе. Вот пример того, что происходит, когда обычный пользователь пытается добавить нового пользователя в базу данных безопасности:

```
C:\>gsec -user norman -password norman
GSEC> add myuser -pw mypassword
add record error
no permission for insert/write access to TABLE USERS
```

Обычный пользователь может только просматривать информацию в базе данных. Причем до версии 2 СУБД Firebird он имеет возможность просмотреть список всех пользователей, зарегистрированных в базе данных безопасности (см. ниже), а в СУБД Firebird 2 он увидит только самого себя.

```
C:\>gsec -user norman -password norman -display
      user name                uid   gid   full name
-----
SYSDBA                        0     0
NORMAN                        0     0     Norman Dunbar
ЕРОСМАН                       0     0     Benoit Gilles Mascia
```

Отличия интерактивного и пакетного режимов

Команды GSEC применимы к обоим режимам, но при работе в пакетном режиме вы должны использовать для них префикс "знак минус" (-), иначе получите нижеследующее сообщение об ошибке:

```
C:\>gsec -user sysdba -password masterkey display
invalid parameter, no switch defined
error in switch specifications
GSEC>
```

Заметьте, что GSEC оставляет вас в интерактивном режиме в случае возникновения ошибки. Чтобы этого избежать, достаточно поставить минус перед командой **display**:

```
C:\>gsec -user sysdba -password masterkey -display
      user name                uid   gid   full name
-----
SYSDBA                        0     0
```

NORMAN	0	0	Norman Dunbar
ЕРОСМАН	0	0	Benoit Gilles Mascia

Теперь GSEC выполняет то, что от него требуют - отображает список пользователей, и завершает работу.

Внимание

Переменные окружения операционной системы `ISC_USER` и `ISC_PASSWORD` позволят запустить GSEC без использования опций командной строки `-user` и `-password`. Однако помните, что оставлять правильные значения в этих переменных на длительное время небезопасно.

Внимание

Как и в случае с другими утилитами командной строки помните, что лучше всего в работе использовать утилиту GSEC, поставляемую с версией вашего сервера баз данных.

Коды завершения в пакетном режиме

Запуская утилиту GSEC в ОС windows, вы можете проанализировать код завершения через значение переменной `%ERRORLEVEL%` и определить, успешно завершилась GSEC или возникла ошибка при выполнении требуемой задачи.

В случае с ОС Unix любой разновидности код завершения можно оценить по значению переменной `$?`.

К сожалению, до СУБД Firebird 2 утилита GSEC всегда завершается с нулевым кодом, что делает невозможным перехватить ошибку ее выполнения в пакетном режиме.

Ошибки в пакетном режиме переводят GSEC в интерактивный режим

Как уже говорилось ранее, в некоторых ситуациях ошибка пакетного режима может привести к тому, что утилита GSEC перейдет в интерактивный режим, и скрипт прервет выполнение, ожидая клавиатурного ввода пользователя. Путь решения данной проблемы был описан выше.

GSPLIT - Фильтр для разбиения на фрагменты

Перевод главы на русский язык: Василий Овчинников

Введение

Еще не так давно многие операционные системы имели ограничение на размер создаваемого файла. Предельный размер составлял 2 Гб для одних и 4 Гб для других систем. Например, для HP-UX 10.20 или 11.00 Unix максимальный размер создаваемого файла ограничен 2 Гб, даже если в настройках файловой системы включена опция *largefiles*. Такое ограничение существует и для некоторых систем, используемых в настоящее время.

GSPLIT - это фильтр, который появился в СУБД Interbase 5.0, позволяющий разделить на фрагменты получаемый в результате работы утилиты GBAK файл резервной копии базы данных, или объединить ранее полученные с его помощью фрагменты резервной копии для восстановления базы данных. До выхода СУБД Interbase 5.0 максимальный размер файла резервной копии составлял 2 Гб из-за ограничений, накладываемых самой утилитой GBAK, причем не зависимо от используемой файловой системы.

В этой главе рассматриваются:

- Опции командной строки GSPLIT.
- Параметры опций GSPLIT.
- Разделение резервной копии на фрагменты с помощью GSPLIT.
- Объединение фрагментов резервной копии с помощью GSPLIT.

Замечание

Начиная с СУБД Interbase 6.0, необходимость использовать GSPLIT пропала, так как GBAK начал поддерживать разбиение больших файлов самостоятельно. Описание GSPLIT, приведенное в этом документе, рекомендуется для прочтения только в справочных целях. А для разбиения больших резервных копий вам необходимо использовать GBAK, даже если GSPLIT входит в поставку СУБД Firebird, используемую вами.

GSPLIT поставляется только с Windows-версией СУБД Firebird 1.5 и не поставляется с версией для ОС Linux. В ОС Linux нет необходимости в отдельной утилите, поскольку там имеется возможность разбить большой файл командой **split**, впрочем, можно также использовать и GBAK.

Внимание

В тестах СУБД Firebird 1.5 на ОС Windows XP Home утилита GSPLIT не работает, и всегда завершает работу с кодом ошибки 9.

Именно из-за проблем с корректной работой GSPLIT вам настоятельно рекомендуется использовать утилиту GBAK и ее возможность разбиения больших файлов, и не пытаться использовать GSPLIT.

Опции командной строки GSPLIT

Для GSPLIT существует три опции командной строки и, по правде говоря, опция **-help** отображает не совсем корректную информацию. Вот эти опции :

- **-split_bk_file <параметры>**

Указывает на то, что GSPLIT используется для разбиения выходного потока GBAK при создании резервной копии базы данных на несколько фрагментов, сохраняемых в виде отдельных файлов. При указании этой опции допускается сократить ее до вида **-s**.

- **-join_bk_file <параметры>**

Указывает на то, что GSPLIT используется для объединения ранее полученных фрагментов резервной копии и отправки потока на вход утилиты GBAK при восстановлении базы данных из резервной копии. При указании этой опции допускается сократить ее до вида **-j**.

- **-help**

При задании ключа **-help**, обнаружении неверного ключа или если не указано ни одного ключа, отображается следующая информация :

```
gsplit: invalid option '-help'
gsplit: Command Line Options Are:
gsplit -S[PLIT_BK_FILE] <file> <size>{k|m|g}
        [... <file> [<size>{k|m|g}]] or
gsplit -J[OINT_BK_FILE] <file> [... <file>]
gsplit: option can be abbreviated to the unparenthesized characters
gsplit: Exiting before completion due to errors
```

Перевод:

```
gsplit: неверная опция '-help'
gsplit: Опции командной строки:
gsplit -S[PLIT_BK_FILE] <файл> <размер>{k|m|g}
        [... <файл> [<размер>{k|m|g}]] или
gsplit -J[OINT_BK_FILE] <файл> [... <файл>]
gsplit: опции могут быть сокращены до символов вне скобок
gsplit: Выполнение прервано из-за ошибок
```

Замечание

В приведенном тексте есть ошибка. Правильное написание опции объединения фрагментов большой резервной копии на самом деле не **-JOINT_BK_FILE**, как написано, а **-JOIN_BK_FILE**.

Параметры опций GSPLIT

Обе основные опции командной строки утилиты GSPLIT требуют указания параметров. При разбиении файла резервной копии это :

- **-S [PLIT_BK_FILE] <файл> <размер>{к|m|г} [... <файл> [размер>{к|m|г}]]**

Первый параметр - имя файла и его максимально допустимый размер. Размер можно указать в килобайтах, мегабайтах или гигабайтах. Пробелы между цифрами и буквой-идентификатором не допускаются. Однако между именем файла и размером пробел необходим.

Остальные параметры нужны для указания второго и последующих файлов и их размеров. Последний файл не нуждается в указании размера, так как он будет заполнен остатком данных, которые превышают размер ранее указанных файлов. Если размер последнего файла все же указать, то он будет просто проигнорирован, но без вывода сообщения о какой-либо ошибке или предупреждения.

Если вы планируете создать резервную копию, которая будет занимать 4 Гб и укажете параметры для создания всего двух файлов по 1 Гб, то GSPLIT проигнорирует размер второго файла, и этот файл будет заполняться до окончания работы GBAK.

Размер файла менее 1 Мб указывать нельзя. В этом случае выводится сообщение об ошибке.

Замечание

GSPLIT правильно считает килобайт как 1024 байт, мегабайт как 1024 килобайт, и гигабайт как 1024 мегабайт.

- **-J [OIN_BK_FILE] <файл> [... <файл>]**

Для объединения файлов и последующего восстановления базы данных просто укажите файлы в правильной последовательности. Если порядок указан неверно, то GSPLIT сообщит об этом и прервет работу.

Разбиение резервных копий

Для использования GSPLIT вы должны указать его как фильтр в командной строке GBAK, например:

```
C:\>gbak -b norman.fdb stdout |
    gsplit -split norman_1.fbk 1m norman_2.fbk 1m norman_3.fbk
```

Внимание

Подразумевается, что переменные окружения ISC_USER и ISC_PASSWORD установлены и содержат правильные значения. Прежде чем определять значения этих переменных в реальной системе, помните о том, что это небезопасно, поскольку позволяет злоумышленникам получить их значения.

В приведенном примере команда напечатана в двух строках. Это сделано для исключения проблем при переводе документа в формат PDF. На самом деле команда должна вводиться одной строкой.

К сожалению, как показано ниже, утилита завершает работу с ошибкой выполнения 9:

```
C:\>gbak -b norman.fdb stdout |
    gsplit -split norman_1.fbk 1m norman_2.fbk 1m norman_3.fbk
fail to read input from ib_stdin, errno = 9
gsplit: program fails to generate multi-volumn back-up files
Done with volume #0, "stdout"
    Press return to reopen that file, or type a new
    name followed by return to open a different file.
Name: ^C
```

Если после приглашения Name: вы укажете имя файла, то произойдет формирование единого файла безо всяких разбиений. Лучше просто нажмите **CTRL-C**, чтобы ненароком не переписать данные какого-нибудь важного файла и избежать связанных с этим проблем.

Однако, первый файл из указанного списка (norman_1.fbk), все же создается, и туда записывается 100 байт - заголовок файла, по которому определяется, что это файл, созданный GSPLIT.

Внимание

Подразумевается, что переменные окружения ISC_USER и ISC_PASSWORD установлены и содержат правильные значения. Прежде, чем определять значения этих переменных в реальной системе, помните о том, что это небезопасно, т.к. позволяет злоумышленникам получить их значения.

В приведенном примере команда напечатана в двух строках. Это сделано для исключения проблем при переводе документа в формат PDF. На самом деле команда должна вводиться одной строкой.

Замечание

Ошибки в написании слов 'program' и 'volumn' в примере воспроизводятся самим GSPLIT.

Объединение фрагментов резервной копии

Если бы предыдущий пример был работоспособен, то команда объединения созданных фрагментов резервной копии и восстановление базы данных выглядела бы так:

```
C:\>gsplit -join norman_1.fbk norman_2.fbk norman_3.fbk |
    gbak -c stdin create_norman.fdb
```

Внимание

В приведенном примере команда напечатана в двух строках. Это сделано для исключения проблем при переводе документа в формат PDF. На самом деле команда должна вводиться одной строкой.

Фрагменты резервной копии, полученные с помощью утилиты GBAK и не обработанные GSPLIT нельзя объединять с помощью GSPLIT. Вот так в этом случае выглядит ошибка выполнения:

```
C:\>gsplit -join norman_1.fbk norman_2.fbk norman_3.fbk |
    gbak -c stdin create_norman.fdb
```

```
gsplit: expected GSPLIT description record
gsplit: Exiting before completion due to errors
gsplit: program fails to join multi-volume back-up files
gbak: ERROR: expected backup description record
gbak: Exiting before completion due to errors
```

Внимание

В приведенном примере команда напечатана в двух строках. Это сделано для исключения проблем при переводе документа в формат PDF. На самом деле команда должна вводиться одной строкой.

Получается, что утилиты GSPLIT и GBAK формируют различные заголовки в файлах резервной копии, и несовместимы между собой.

Замечание

Ошибки в написании слов 'program' и 'volume' в примере воспроизводятся самим GSPLIT.

Скрипты для Firebird в ОС Linux/Unix

Перевод главы на русский язык: Александр Карпейкин

Введение

После завершения установки СУБД Firebird 1.5 в директории `/opt/firebird/bin` вы можете увидеть несколько полезных скриптов. В данной главе рассказывается для чего предназначены эти скрипты и как ими пользоваться.

Здесь мы расскажем о следующих скриптах:

- `changeDBAPassword.sh`
- `createAliasDB.sh`
- `fb_config`
- `changeRunUser.sh`
- `restoreRootRunUser.sh`
- `changeGdsLibraryCompatibleLink.sh`

Замечание

Приведённый список корректен на время написания статьи и проверялся на СУБД Firebird 1.5 SuperServer, установленной под ОС Linux. Другие клоны Unix-систем могут иметь другие или изменённые скрипты. В настоящее время я не готов обсуждать эти возможные изменения из-за отсутствия доступа к другим Unix-системам.

Изменение пароля SYSDBA

Скрипт `changeDBAPassword.sh` позволяет изменить пароль для пользователя SYSDBA (администратора баз данных) и изменить владельца других, стартующих автоматически (`startup`), скриптов. Скрипт выполняется в неинтерактивном режиме, как часть процесса установки (при установке создаётся случайно сгенерированный пароль, который сохраняется в файле `/opt/firebird/SYSDBA.password`). Пароль в последствии используется при запуске скрипта `/etc/rc.d/init.d/firebird`, который также является символической ссылкой на `/etc/init.d/firebird`. Новый пароль также изменяется в базе данных безопасности (с логинами и паролями пользователей) `/opt/firebird/security.fdb`.

Предостережение

Скрипт должен выполняться пользователем root, и в процессе выполнения сначала будет запрошен ввод текущего пароля SYSDBA, а затем нового пароля. Оба пароля будут видны на экране, поэтому для повышения безопасности не позволяйте никому смотреть на экран при выполнении скрипта.

После выполнения скрипта файл /opt/firebird/SYSDBA.password будет содержать новый пароль в текстовом виде, поэтому убедитесь, что этот файл не доступен для чтения никому, кроме пользователя root.

Следующий пример показывает изменение пароля SYSDBA с 'masterkey' на 'biroguin', который не является словом и будет более устойчив к взлому и угадыванию.

```
# cd /opt/firebird/bin
# ./changeDBAPassword.sh
Please enter current password for SYSDBA user : masterkey
Please enter new password for SYSDBA user : biroguin
GSEC> GSEC>

Running ed to modify /etc/init.d/firebird
#
```

Предостережение

Если вы меняете пароль SYSDBA с помощью утилиты GSEC, поменяйте его также в автозапускаемых скриптах. Всегда используйте скрипт изменения пароля SYSDBA, чтобы быть уверенным, что изменения применены везде.

Создание псевдонима для базы данных

Скрипт createAliasDB.sh позволяет создать новую базу данных и записать её псевдоним (сопоставленный с полным именем файла базы данных; он же "алиас") в файл /opt/firebird/aliases.conf.

Замечание

Скрипт createAliasDB.sh должен выполняться пользователем root.

Если ваша система неверно сконфигурирована, то новая база данных может не быть создана, но её псевдоним будет добавлен в файл псевдонимов. В дальнейшем это может не позволить добавить новый псевдоним при помощи скрипта, так как скрипт не заменяет существующий псевдоним при совпадении их имен. Вы должны сами удалить ошибочный псевдоним из конфигурационного файла.

Для создания новой базы данных владельцем той директории, где база данных создается, должен быть пользователь firebird из группы firebird. Ниже приведён пример создания директории пользователем root для баз данных СУБД Firebird.

```
# cd /u01
# mkdir databases
# chown firebird:firebird databases
#
```

Теперь директория /u01/databases доступна для хранения одной или нескольких баз данных СУБД Firebird. Очевидно, что директория /u01 в данном примере уже существовала.

Парметры скрипта *createAliasDB.sh*

Для выполнения скрипта `createAliasDB.sh` используйте командную строку следующего вида:

```
# createAliasDB.sh <новый_псевдоним> <имя_файла_БД>
```

Скрипт имеет два обязательных параметра:

- **новый_псевдоним**

Имя нового псевдонима. Имя псевдонима не должно повторять уже существующие имена в конфигурационном файле, в противном случае выведется сообщение об ошибке и выполнение скрипта прервётся.

- **имя_файла_БД**

Полный путь к файлу базы данных. Вы не должны указывать здесь относительный путь, поскольку он может быть некорректным при соединении с базой данных. Скрипт отклоняет любые попытки ввода относительного пути вместо полного.

Файл с новой пустой базой данных будет создан только в случае, если база данных с таким именем файла не существует. Если же такая база данных уже есть, то будет создан только её псевдоним в конфигурационном файле.

Помощь с файлами *makefile*

Скрипт `fb_config` предназначен для задания различных установок и опций компилятора в файлах `makefile`, которые будут затем использоваться при инсталляции пакетов.

Замечание

Этот скрипт может выполняться любым пользователем, имеющим право его выполнять. Вам не обязательно быть пользователем `root` для его выполнения.

Опции *fb_config*

Для выполнения скрипта используется командная строка следующего вида:

```
fb_config <опция> [ <опция> [...]]
```

Скрипт может иметь один или несколько параметров в командной строке:

- **--help**

Выводит список доступных параметров. Он не поддерживается совместно с другими параметрами.

- **--cflags**

Выводит список директорий, где расположены файлы СУБД Firebird. Это требуется только для компиляторов С и С++ для корректного заполнения директивы `#include`. На моей системе скрипт с этим параметром вывел следующую строчку: `'-I/opt/firebird/include'`.

- **--libs**

Выводит список директорий, в которых находятся библиотеки СУБД Firebird, и список библиотек, которые требуются при линковке для создания клиент-серверных приложений для СУБД Firebird. На моей системе скрипт вывел `'-L/opt/firebird/lib -lfbclient'`.

- **--embedlibs**

Выводит список директорий, в которых находятся библиотеки СУБД Firebird, и список библиотек, которые требуются при линковке для создания встроенного в приложение сервера Firebird. На моей системе скрипт вывел `'-L/opt/firebird/lib -lfbembed'`.

- **--bindir**

На моей системе скрипт вывел `'/opt/firebird/bin'` как полный путь к директории `/bin` СУБД Firebird.

- **--version**

Выводит строку с версией, составленную из версий СУБД Firebird, пакета (после дефиса) и архитектуры системы (после точки). Мой лэптоп с операционной системой Linux вывел такую строку - `'1.5.0.4290-0.i686'`.

Ниже приведён отрывок из файла `makefile`, который показывает, как с помощью `fb_config` корректно задать значения для двух макросов - `FBFLAGS` и `FBLIBS`. Отметим, что вместо символа прямой одинарной кавычки (`'`) используется обратная одинарная кавычка (```).

```
...
FBFLAGS = `fb_config --cflags`
FBLIBS = `fb_config --libs`
...
```

Изменение пользователя для запуска Firebird

Существует два варианта скрипта `changeRunUser.sh`. Для варианта `SuperServer` название скрипта начинается с `'SS'`, а для варианта `ClassicServer` - с `'CS'`. Далее рассматривается только скрипт для версии `SuperServer`.

Замечание

Этот скрипт должен выполняться пользователем `root`.

Скрипт `SSchangeRunUser.sh` позволяет изменить группу и пользователя, от имени которых запускается и работает СУБД Firebird. По умолчанию это группа и пользователь `firebird`. Отметим, что в более

ранних версиях сервер Firebird работал от имени пользователя root, что нежелательно с точки зрения безопасности и секретности системы, и позволяет создавать базы данных в любом месте файловой системы. Новый пользователь firebird имеет ограничения в плане размещения создаваемых баз данных.

Скрипт изменяет владельца (пользователя и группу) для директории установки СУБД Firebird и для директорий и файлов внутри неё и автозапускаемого (при старте операционной системы) скрипта `/etc/rc.d/init.d/firebird`, который предназначен для запуска и остановки сервера Firebird.

Параметры скрипта `SSchangeRunUser.sh`

Для выполнения скрипта используется командная строка следующего вида:

```
SSchangeRunUser.sh <имя_пользователя> <имя_группы>
```

Скрипт использует два параметра командной строки, оба из которых являются необязательными - при их отсутствии оба параметра будут запрошены. Если задан только один параметр, то он ассоциируется с параметром `<имя_пользователя>`, а параметр `<имя_группы>` будет запрошен.

- **имя_пользователя**

Этот параметр определяет пользователя, от имени которого запускается и работает СУБД. Пользователь должен существовать в файле `/etc/passwd`.

- **имя_группы**

Этот параметр определяет группу, от имени которой запускается и работает СУБД. Группа должна существовать в файле `/etc/group`.

Ниже приведён пример смены владельца (пользователя и группы) на firebird с использованием скрипта `SSchangeRunUser.sh`. После выполнения скрипта пользователь и группа firebird будут теперь по умолчанию управлять работой сервера Firebird, если вы уже не сделали этого.

```
# cd /opt/firebird/bin
# ./SSchangeRunUser.sh firebird firebird
Updating /opt/firebird
Updating startup script
Completed
#
```

Восстановление прав root как владельца Firebird

Существует две варианта скрипта `restoreRootRunUser.sh`. Для варианта SuperServer название скрипта начинается с 'SS', а для ClassicServer - с 'CS'. Далее рассматривается только скрипт для версии SuperServer.

Замечание

Этот скрипт должен выполняться пользователем root.

Скрипт просто восстанавливает `root` как пользователя и группу, от имени которых запускается и работает СУБД Firebird. Данный скрипт является модификацией `SSchangeRunUser.sh` путём замены имени пользователя и группы на `root`.

Запуск приложений Embedded или ClientServer

Скрипт `changeGdsLibraryCompatibleLink.sh` доступен только при установке `ClassicServer`, и используется для замены символической ссылки (`symlink`) `libgds.so` ссылкой на место установки соответствующей библиотеки. Возможны два вида символической ссылки на библиотеку:

- `/opt/firebird/lib/libfbclient.so` для клиент-серверных приложений;
- `/opt/firebird/lib/libfbembed.so` для приложений со встраиваемым сервером (`embedded`).

После установки символическая ссылка `libgds.so` по умолчанию указывает на библиотеку для клиент-серверных приложений. Если вы хотите запустить приложение со встраиваемым сервером, то необходимо выполнить данный скрипт для изменения символической ссылки `libgds.so` на библиотеку для встраиваемого сервера.

Замечание

Этот скрипт должен выполняться пользователем `root`.

Пример показывает использование скрипта для изменения символической ссылки на библиотеку со встраиваемой на клиент-серверную:

```
# cd /opt/firebird/bin
# ./changeGdsCompatibleLibraryLink.sh
For classic server there are two optional backward compatible
client libraries. These are libfbclient.so and libfbembed.so.

libfbclient.so) enables your client to be multithreaded but must
                  connect to a database via a server.
libfbembed.so)  allows the client to directly open the database file,
                  but does not support multithreaded access

Your current setting is:
/usr/lib/libgds.so -> /opt/firebird/lib/libfbembed.so

Which option would you like to choose
(client|embed|remove) [client] client
#
```

Опция по умолчанию - **client** - пересоздаст символическую ссылку с указанием пути на клиент-серверную библиотеку, опция **embed** - на библиотеку для встраиваемого сервера, а **remove** - вообще удалит ссылку.

В случае успешного выполнения скрипта не выводится никаких сообщений, но при повторном его запуске вы увидите изменённый путь для символической ссылки на библиотеку - в отличие от предыдущего запуска.

Планы на будущее ...

Так как работа над книгой продолжается, прошу прощения за "неожиданное" окончание. По мере изучения и документирования утилит командной строки я добавлю в книгу новые главы. А до того времени данный раздел будет содержать краткое описание моих планов по завершению этой книги.

- **FB_LOCK_PRINT** - утилита, которая выводит детализацию внутренней страницы блокировок базы данных.
- **GBAK** - утилита резервного копирования и восстановления. Она также позволяет изменять некоторые внутренние параметры базы данных.
- **GDEF** - утилита для метаданных, которая удалена из СУБД Interbase 4.0, и была возвращена в версии 6 (с открытым Кодом). GDEF, скорее всего, лишняя утилита.
- **GFIX** позволяет предпринять попытку починить поврежденную базу данных, открыть (on-line) и остановить (shutdown) базу данных для соединений, решить проблемы "зависших" между несколькими базами данных транзакций (in-limbo), изменить количество страниц буфера, и так далее.
- **GPRE** - это препроцессор, который конвертирует исходный код, написанный на некоторых языках и содержащий встроенный псевдокод SQL, в корректный отформатированный вызов функций Firebird API.
- **GSTAT** дает возможность администратору СУБД Firebird собирать статистику об общем состоянии и использовании разных частей базы данных.
- **ISQL** - это интерактивный инструмент, который позволяет выполнять запросы к базе данных Firebird. Это консольное приложение (как и многие другие утилиты), и оно есть во всех дистрибутивах СУБД Firebird. ISQL - это лучший способ опробовать ваши скрипты и команды в первый раз.
- **QLI** - это оригинальный Query Language Interpreter, который был удален из СУБД Interbase 4.0, но был возвращен в версии 6.0, поскольку было принято решение о распространении СУБД Interbase в открытых кодах.

Norman Dunbar.

Приложение А: Лицензионное соглашение

Содержание данного набора документации является предметом Public Documentation License версии 1.0 (далее «Лицензия»); вы можете пользоваться данным набором документации только в случае согласия с условиями Лицензии. Копии Лицензии доступны по ссылкам <http://www.firebirdsql.org/pdfmanual/pdl.pdf> (PDF) и <http://www.firebirdsql.org/manual/pdl.html> (HTML).

Оригинальное название документа: *Firebird Commandline Utilities*.

Автор исходного документа: Norman Dunbar.

Copyright (C) 2004–2006. Все права защищены. Адрес электронной почты для контакта с автором: NormanDunbar at users dot sourceforge dot net.

Перевод на русский язык: Василий Овчинников.

Разделы, переведенные Василием Овчинниковым: Copyright (C) 2006. Все права защищены. Контактный адрес электронной почты: ova at tula dot net.

Перевод на русский язык: Александр Карпейкин.

Разделы, переведенные Александром Карпейкиным: Copyright (C) 2006. Все права защищены. Контактный адрес электронной почты: karp.fb at gmail dot com.

Перевод на русский язык: Сергей Ковалёв.

Разделы, переведенные Сергеем Ковалёвым: Copyright (C) 2006. Все права защищены. Контактный адрес электронной почты: mrKovalev at yandex dot ru.

Корректор перевода: Павел Меньщиков.

Copyright (C) 2006. Все права защищены. Контактный адрес электронной почты: developer at ls-software dot ru.