Firebird and InterBase command-line utilities

Several services and command-line tools are provided with Firebird/InterBase®. They perform the same range of functions as the Server Manager and run on both UNIX and Windows platforms. Like the Server Manager, they can access servers on any platform that Firebird /InterBase® supports. The command-line tools include the following:

s Firebird and InterBase® services

- ibmgr - windows-based server management program.
- fbguard - the Firebird Guardian service.
- fbserver - the actual Firebird Superserver database service.
- fb_inet_server.exe - the Firebird Classic database service.
- fbmgr - Linux only: starts and shuts down the Firebird demon.
- fbsvcmgr - introduced in Firebird 2.1; provides a command-line interface to the Services API.
- instsvc - service setup.
- instreg - registry parameters setup.

Firebird and InterBase® administration tools

- isql - Interactive SQL.
- gbak and gsplit
- gbak - Firebird backup and restore.
- nbak - engine support module.
- nbackup - incremental backups.
- gsec - user administration.
- gfix - various parameters, repair.
- gstat - statistics.
- fbtracemgr - command-line interactive tool for the execution of DDL and DML commands and scripts.

Refer also to the documentation relevant for your Firebird/InterBase® version:

- InterBase 6.0 Operations Guide (this is also the basis for Firebird 1)
- Firebird 1.5.6 Release Notes
- Firebird 2.0.4 Release Notes
- Firebird 2.1 Release Notes
- Firebird 2.5 Release Notes

The majority of the options provided by these command-line tools are also offered by IBExpert. Please refer to IBEScript.exe and IBEBLOCK for further information.

From 2009 to 2010 the Firebird Foundation updated their own command-line tool documentation, which can also be found on the IBExpert documentation site. Documentation for the individual tools is mentioned under the respective subjects below.

Firebird and InterBase® services

ibmgr

(ibmgr.exe)

ibmgr is a windows-based server management program, and includes the functionalities found in gsec, gbak and gfix.

back to top of page

fbguard

(fbguard.exe)

The Firebird Guardian monitors the server process. Should the server go down for whatever reason the Guardian automatically restarts it. Please refer to Firebird Guardian in the Download and Install Firebird chapter for further information.

Firebird users: please also refer to the Firebird Superserver Manager.

FBGuard: command-line switches

Basic fbguard.exe parameters

-a	Run both Server Guardian as application	Not needed on Windows 95/98.

Source: https://www.volny.cz/iprenosil/interbase/fbserver_switches.html

back to top of page

fbserver

(fbserver.exe)

This is the Firebird/InterBase® Superserver binary.

Firebird users: please also refer to the Firebird Superserver Manager.

FBServer: command-line switches

Basic fbserver.exe parameters

- -a Run as application. **This is default on Windows 95/98.**
- -n No icon in tray. **Applicable only if Firebird runs as application.**
- -s Run as service **Used when installing service; obsolete.**

Priority of server process

- -b Run with high (boosted) priority.
- -r Run with normal priority. **Default.**

TCP/IP parameters

-p Specify port number/name or named pipe. **It overrides setting in firebird.conf.**

Connection protocols

- -i Accept tcp/ip connections. **If neither of -i, -l, -w is specified, all protocols will be enabled.**
- -l Accept local connections. **Superserver only.**
- -w Accept NetBEUI connections. **Windows NT/2000/XP only.**
- -x Accept XNET connections. **New version of local protocol (under development).**

Other

-z Print server version and exit. **(Does not work…)**

Source: https://www.volny.cz/iprenosil/interbase/fbserver_switches.html

Note: Firebird 1 allows only these switches: -a, -n, -s, -p, -z.

fb_inet_server

(fb_inet_server.exe)

This is the Firebird Classic binary.

back to top of page

fbmgr

(fbmgr.exe)

fbmgr is the Services Manager for Linux systems. It can be used if it becomes necessary for you to manually shut down the engine and to restart it again afterwards. fbmgr can be found in the bin subdirectory. It can only be started by an administrator.

fbmgr can be run interactively or in batch mode. The commands are identical whichever mode you use and the following section describes the commands and shows examples of each, running in both modes. To enter interactive mode, simply log in as the firebird user and type the command fbmgr. If /opt/firebird/bin is not on your path, type the command bin/fbmgr instead.

At boot time, the script /etc/init.d/firebird is executed to start up the Firebird engine on Suse servers, other Linux servers use the script /etc/rc.d/init.d/firebird instead. The same script is executed just before the server shuts down to stop the Firebird engine. The engine runs as the firebird user and not as root.

The script is installed to the location described above, and a symbolic link is created to /usr/sbin/rcfirebird and the script can therefore be called in both ways, whichever you find easiest, but you must be the root user:

```
tux> /etc/init.d/firebird <parameter>
```

or

```
tux> rcfirebird <parameter>
```

This script can take one of the following parameters:

start Starts the Firebird engine. If already running, does nothing. **stop** Stops the Firebird engine. If already stopped, does nothing. **status** Checks the current status of the Firebird engine. **try-restart** Restarts the Firebird engine if it is currently running, otherwise, does nothing. **restart** Stops and restarts the Firebird engine, whether it was running or not. **force-reload** This is exactly the same as the restart command. **reload** Reload is not implemented.

Starting up

The service process can be started using the –start option:

```
./fbmgr -start
```

A further option may be specified to determine whether the Guardian should also run or not:

–once Starts Firebird without the Guardian. –forever Starts Firebird with the Guardian. This is the default parameter.

```
./fbmgr -start -forever
```

Shutting down

Using the option –shut all transactions are rolled back (ROLLBACK), all client connections disconnected and the service process shut down. The SYSDBA password has to be specified:

```
./fbmgr -shut -password masterkey
```

Please refer to the Firebird Superserver Manager for the full documentation.

Source: Firebird SQL Server 2.x Administration Handbook by Stefan Heymann, and the Firebird Superserver Manager by Norman Dunbar.

See also:

- Using IBExpert and Delphi applications in a Linux environment: accessing Firebird
- Firebird Superserver Manager

back to top of page

fbsvcmgr

(fbsvcmgr.exe)

Introduced in Firebird 2.1, the fbsvcmgr utility provides a command-line interface to the Services API, enabling access to any service that is implemented in Firebird.

fbsvcmgr does not emulate the switches implemented in the traditional “g*” utilities. Rather, it is just a frontend through which the Services API functions and parameters can pass. Users therefore need to be familiar with the Services API as it stands currently. The API header file - ibase.h, in the ../include directory of your Firebird installation - should be regarded as the primary source of information about what is available, backed up by the InterBase® 6.0 beta API Guide.

For further information please refer to the New command-line utility fbsvcmgr chapter in the Firebird 2.1 Release Notes.

back to top of page

instsvc

(instsvc.exe)

Normally the instsvc command is used to install or remove services. It can also be used to start and stop services that already exist.

Windows doesn't have an fbmgr application as it uses the instsvc command instead.

See also:

- Firebird 2.1.3 Release Notes: Parameter for instance name added to instsvc.exe
- Firebird 2.0.4 Release Notes: Interactive option added to instsvc.exe
- Installing Firebird from a zip kit

back to top of page

instreg

(instreg.exe)

Registry parameters setup. instreg is used for the registration of an installation in the registry (only necessary when installing manually). The instreg utility does all the work, making the necessary entries in the right places, and installs everything required in the registration. It usually installs the Firebird Guardian too, and concludes by starting the service.

See also:

Installing Firebird from a zip kit

back to top of page

Firebird and InterBase® administration tools

isql

isql is a command-line utility program that can run SQL queries on the database. isql supports data definitions and data manipulation commands as well as SQL scripts with multiple SQL commands within one script. It can be used to create and modify the database's metadata, insertion, alteration and deletion of data, data queries and the display of results (all this can be done in the IBExpert SQL Editor), adding and removal of user database rights (see the IBExpert User Manager and Grant Manager) and execution of other database administrative functions. It is very similar to DSQL, with some omissions, such as cursors, and a few additions, for example, SET and SHOW.

isql commands end with ;. Each command must be explicitly committed using the commit statement.

When unknown parameters are used, isql shows all of the command-line parameters and their explanations instead of just a simple list of allowed switches:

```
opt/firebird/bin] isql -?

Unknown switch: ?
usage: isql [options] [<database>]
```

- -a(all) Extract metadata incl. legacy non-SQL tables.
- -b(ail) Bail on errors (set bail on).
- -c(ache) <num> Number of cache buffers.
- -ch(arset) <charset> Connection charset (set names).
- -d(atabase) <database> Database name to put in script creation.
- -e(cho) Echo commands (set echo on).
- -ex(tract) Extract metadata.

- -i(nput) <file> Input file (set input).
- -m(erge) Merge standard error.
- -m2 Merge diagnostic.
- -n(oautocommit) No autocommit DDL (set autoddl off).
- -now(arnings) Do not show warnings.
- -o(utput) <file> Output file (set output).
- -pag(elength) <size> Page length.
- -p(assword) <password> Connection password.
- -q(uiet) Do not show the message Use CONNECT….
- -r(ole) <role> Role name.
- -r2 <role> Role (uses quoted identifier).
- -sqldialect <dialect> SQL dialect (set sql dialect).
- -t(erminator) <term> Command terminator (set term).
- -tr(usted) Use Windows trusted authentication.
- -u(ser) <user> User name.
- -x Extract metadata.
- -z Show program and server version.

For new features and switches introduced in Firebird 2 please refer to Firebird 2.0.4. Release Notes: isql query utility and Firebird 2.1 Release Notes: Utilities support for database triggers. The -nodbtriggers parameter was introduced in Firebird 2.1 to suppress database triggers from running. They are available only to the database owner and SYSDBA.

New to Firebird 2.5: A new client-side API function, fb_sqlstate() is available to convert the status vector item for an error into the corresponding SQL-2003 standard 5-alphanumeric sqlstate.

- The sqlstate code represents the concatenation of a 2-character SQL class and a 3-character SQL subclass.
- Statements now return an sqlstate completion code.
- The isql utility now prints the sqlstate diagnostic for errors instead of the sqlcode one
- The sqlcode diagnostic is deprecated—meaning it will disappear in a future release.

Deprecated sqlcode: Although the sqlcode is deprecated and use of the sqlstate is preferred, it remains in Firebird for the time being. The isc_sqlcode() API function is still supported, as is the when sqlcode exception handling. Appendix A: sqlstate provides a list of all sqlstate codes in use in this release, along with the corresponding message texts.

See also:

Firebird Interactive SQL Utility

back to top of page

gbak and gsplit

gbak is an Firebird/InterBase® command-line utility, which can be used to back up and restore databases. gsplit backs up and restores multiple file databases.

gsplit is documented fully in the Firebird Backup File Splitting Filter. gbak is fully documented in the Firebird Backup & Restore Utility.

Below you will find a synopsis of the options available in gbak along with a range of examples.

gbak

The parameters and options offered by gbak can be found in the IBExpert Backup Database and Restore Database menus.

gbak is fully documented in the Firebird Backup & Restore Utility. For new and altered Firebird 2 parameters, please refer to: Firebird 2.0.4. Release Notes: gbak backup/porting/restore utility, Firebird 2.1 Release Notes: Utilities support for database triggers and Firebird 2.5 Release Notes: gbak.

Many thanks to Stefan Heymann (https://www.destructor.de) for the following overview of options and examples.

gbak is Firebird's/InterBase's command-line tool for online backup and restore of a complete database.

General Syntax

gbak <options> -user <username> -password <password> <source> <destination>

Backup

For backups, <source> is the database you want to back up, <destination> is the file name of the backup file. The usual extension is .fbk for Firebird and .gbk for InterBase®.

Only the SYSDBA or the database owner can perform a backup. For multi-file databases, specify only the name of the first file as the database name.

Restore

For restores, <source> is the backup file and <destination> is the name of the database that is to be built up from the backup file. You will have to specify the -C option for restore. Please note that if you run the gbak restore in verbose mode, it can take an awful long time.

Options

(Parts in square brackets are optional)

- -b[ackup_database] Back up. This switch is optional. **Backup only**
- -bu[ffers] Set cache size for restored database. **Restore only**
- -c[reate_database Restore (mandatory). **Restore only**
- -co[nvert] Converts external tables to internal tables. **Backup only**
- -e[xpand] Creates an uncompressed backup. **Backup only**
- -fa[ctor] n Blocking factor for tape device. **Backup only**
- -g[arbage collect] Does not perform garbage collection (sweeping) during backup. **Backup only**
- -i[nactive] All indices will be restored as inactive. **Restore only**
- -ig[nore] Ignores checksum errors while backing up. **Backup only**
- -k[ill] Does not create shadows that are defined in the backup. **Restore only**

- -l[imbo] Ignores Limbo transactions while backing up. **Backup only**
- -m[etadata] Only backs up metadata (schema). No table data will be stored. **Backup only**
- -mo[de] read_write Restores to a read/write database (This is the default). **Restore only**
- -mo[de] read_only Restores to a read-only database. **Restore only**
- -n[o_validity] Does not restore validity constraints. So you can restore data that does not meet these constraints and could not be restored otherwise. **Restore only**
- -nod[btriggers] Do not run database triggers (Firebird 2.1).
- -nt Non-transportable format (use only when you know you will restore on same platform and database version). **Backup only**
- -o[ne_at_a_time] Restores one table at a time. You can use this to partially restore databases with corrupt table data. **Restore only**
- -ol[d_descriptions] Old-style format. **Backup only**
- -p[age_size] <size> Sets page size of new database. <size> can be one of 1024, 2048, 4096, 8192. Default is 1024. **Restore only**
- -pa[ssword] <password> Database password.
- -r[ecreate database] <o[verwrite]>* Create (or replace if overwrite is used). **Restore only**
- r[eplace_database] * Restores over an existing database. This can only be performed by the SYSDBA or the owner of the database hat is overwritten. Do NOT restore over a database that is in use! **Restore only**
- -role <role> Connect as role.
- -se[rvice] <hostname>:service_mgr *Backup*: creates the backup file on the database server, using the Service Manager. Restore: creates the database from a backup file on the server, using the Service Manager.
- -t[ransportable] Creates a transportable backup (transportable between platforms and server versions). **Backup only**
- -tru[sted] Use trusted authentication. **Restore only**
- -u[ser] <username> Database user name.
- -use_[all_space] Normally, on restore, database pages will be filled to about 80 %. With the use_all_space option, database pages will be filled to 100 %. (Useful for read-only databases which will see no more modifications). **Restore only**
- -v[erbose] Verbose output of what gbak is doing.
- -v[erify] Report each action taken.
- -y <filename> Redirect all output messages to <filename>. Note: the file must not exist before running gbak!
- -y suppress_output Quiet mode.
- -z Show gbak version and server version number.

New to Firebird 2.0: Change to gbak -r semantics

An important change has been done to prevent accidental database overwrites as the result of users mistakenly treating -r as an abbreviation for restore. gbak -r was formerly a shortcut for -replace_database. Now the -r switch no longer restores a database by overwriting an existing one, but instead reports an error. If you actually want the former behaviour, you have two alternatives:

- Specify the full syntax gbak -replace_database. There is a new shortcut for the -replace_database switch: gbak -rep

or

- Use the new command -r[ecreate_database] overwrite. The -r shortcut now represents the -r[ecreate_database] switch and the overwrite keyword must be present in either the full or the abbreviated form.

Warning: If you use the full syntax, you are expected to know what this restore mode actually means and have some recovery strategy available if the backup subsequently turns out to be unrestorable.

New to Firebird 2.0: gbak -v and the counter parameter

During Firebird 1 development, an optional numeric <counter> argument was added to the -v[erbose] switch of gbak for both backup and restore. It was intended to allow you to specify a number and get a running count of rows processed as the row counter passed each interval of that number of rows. It caused undesirable side-effects and was removed before Firebird 1.0 was ever released. So, although it never happened, it was documented as “implemented” in the release notes and other places.

gbak examples

A “normal” backup:

```
gbak -v -t -user SYSDBA -password "masterkey" dbserver:/db/warehouse.fdb
c:\backups\warehouse.fbk
```

Backup with output to a logfile:

```
gbak -v -t -user SYSDBA -password masterkey -y c:\backups\warehouse.log
dbserver:/db/warehouse.fdb  c:\backups\warehouse.fbk
```

A “normal” restore:

```
gbak -c -v -user SYSDBA -password masterkey c:\backups\warehouse.fbk
dbserver:/db/warehouse2.fdb
```

Restore to an already existing database:

```
gbak -c -r -v -user SYSDBA -password masterkey c:\backups\warehouse.fbk
dbserver:/db/warehouse.fdb
```

Create a read-only database:

```
gbak -c -v -mode read_only -use_all_space -user SYSDBA -password masterkey
c:\backups\warehouse.fbk c:\files\warehousedb.fdb
```

Multi-file backups

Syntax for backup:

```
gbak [options] <database> <target file 1> <size 1> <target file 2> <size 2>
... <target file n>
```

Note: Do not specify a size for the last file. It will always be filled to take up what is left over, no matter how large. Size can be given in bytes (8192), kilobytes (1024k), megabytes (5m), or gigabytes (2g)

Syntax for restore:

```
gbak -c [options] <source file 1> <source file 2> ... <source file n>
<database>
```

Restoring to a multi-file database

```
gbak -c [options] <source file> <db file 1> <size 1> <db file 2> <size 2>
... <db file n>
```

Note: do not specify a size for the last database file. It can always grow unlimited to take up the rest. Size can be given in bytes (8192), kilobytes (1024k), megabytes (5m), or gigabytes (2g) Restoring from a multi-file backup to a multi-file database:

```
gbak -c [options] <source file 1> <source file 2> ... <source file n> <db
file 1> <size 1> <db file 2>  <size 2> ... <db file n>
```

See also:

- Why is a database backup and restore important?
- Firebird 2.0.4. Release Notes: gbak backup/porting/restore utility
- Backup database
- Firebird Backup & Restore Utility
- Server and client backup strategies

back to top of page

On-line incremental backup

New to Firebird 2.0: the implementation of new, fast, on-line, page-level incremental backup facilities. The backup engine comprises two parts:

- nbak, the engine support module and
- nbackup, the tool that does the actual backups.

nbak

nbak was introduced in Firebird 2.0. Together with nbackup it offers possibilities not present in gbak - Firebird's pre-existing backup tool - but doesn't replace the latter. Refer to Firebird 2.1.3 Release Notes: gbak backup/porting/restore utility for details.

See also:

- Firebird's nbackup tool
- Firebird 2.1 Release Notes: Command-line utilities

nbackup

nbak is fully documented in Firebird's nbackup tool.

nbackup was introduced in Firebird 2.0. It offers possibilities not present in gbak - Firebird's pre-existing backup tool - but doesn't replace the latter. Refer to Firebird 2.1.3 Release Notes: gbak backup/porting/restore utility for details.

Valid Options

- -L <database>: Lock database for filesystem copy.
- -N <database>: Unlock previously locked database.
- -F <database>: Fixup database after filesystem copy.
- -B <level> <database> [<filename>]: Create incremental backup.
- -R <database> [<file0> [<file1>…]]: Restore incremental backup.
- -U <user>: User name.
- -P <password>: Password.
- -T: Added in Firebird 2.1 to suppress database triggers from running. They are available only to the database owner and SYSDBA.

Note:

1. <database> may specify a database alias.
2. incremental backups of multi-file databases are not supported yet.
3. “stdout” may be used as a value of <filename> for the -B option.

Source: Firebird 2.0.4 Release Notes: Command-line utilities

Firebird 2.5 introduced a Services API interface to the backup/restore operations. Backup and restore must be run on the host station and the only way to access them previously was by running nbackup. The two new service actions enabling nbackup backup and restore to be requested through the Services API are:

- isc_action_svc_nbak - incremental nbackup.
- isc_action_svc_nrest - incremental database restore.

The parameter items are:

- isc_spb_nbk_level - backup level (integer).
- isc_spb_nbk_file - backup file name (string).
- isc_spb_nbk_no_triggers - option to suppress database triggers.

Please refer to the Firebird 2.5 Release Notes chapter, nbackup support, for usage examples.

See also:

- Firebird's nbackup tool
- Firebird 2.1.3 Release Notes: gbak backup/porting/restore utility
- Firebird 2.5 Release Notes: nbackup support

back to top of page

gsec

(gsec.exe)

gsec is a Firebird/InterBase® command-line utility, which manages server security. It can be used to add, modify, and delete authorized users on the server. gsec commands apply to the database server and not to individual databases, as with the majority of other command-line utilities. gsec is fully documented in Firebird's Password File Utility.

All options offered by gsec can be found in the IBExpert User Manager and Grant Manager.

Many thanks to Stefan Heymann (https://www.destructor.de) for the following overview of commands, options and examples.

All database users are stored in the security database named security.fdb (since Firebird 2 this file is now called security2.fdb) in the Firebird directory. There is at least one user, the system database administrator, SYSDBA. gsec can be operated in interactive mode or batch mode.

After installation, the SYSDBA password is masterkey. (Exception: Firebird 1.5 for Linux). Only the first 8 characters of a password are significant. The password should be altered as soon as possible and should not contain space characters.

Invoking gsec

To use gsec for the local machine, use:

```
gsec -user sysdba -password <password> [options]
```

To use gsec for a remote machine, use:

```
gsec -user sysdba -password <password> -database <databasename>
```

where <databasename> is the database name of the remote security.fdb/security2.fdb database.

You can use gsec as an interactive command-line tool or give all commands on one command line.

Commands

- di[splay] Displays all users.
- di[splay] <username> Displays all information for the given user.
- a[dd] <username> -pw <password> [options] Add a new user.
- mo[dify] <username> [options] Modify user.
- de[lete] <username> Delete user.
- h[elp] Display help.
- ? Display help.
- q[uit] Quit interactive mode.

- z Display gsec version number.

If you don't want to invoke the interactive mode, you can enter all commands directly in the command line. To do that, precede the commands with a dash.

Options

- -pa[ssword] <password> Password of the user who is performing the change.
- -user <username> User name of the user who is performing the change.
- -pw <password> Password of target user (or new password).
- -fname <first name> Target user's first name.
- -mname <middle name> Target user's middle name.
- -lname <last name> Target user's last name.

Examples

Add user Elvis Presley as user ELVIS, password is Aaron:

```
gsec -user SYSDBA -password masterkey
GSEC> add elvis -pw Aaron -fname Elvis -lname Presley
GSEC> quit
```

Change password of user ELVIS to chuck:

```
gsec -user SYSDBA -password masterkey
GSEC> modify elvis -pw chuck
GSEC> quit
```

Change password of SYSDBA on remote Linux server harry to hamburg:

```
gsec -user SYSDBA -password masterkey -database
harry:/opt/firebird/security.fdb -modify sysdba -pw hamburg
```

Change password of SYSDBA on remote Windows server sally to hannover:

```
gsec -user SYSDBA -password masterkey -database sally:"C:\Program
Files\Firebird\security.fdb" -modify sysdba -pw hannover
```

Change password of SYSDBA on remote server jake on TCP port 3051 to london:

```
gsec -user SYSDBA -password masterkey -database
"jake/3051:/opt/firebird/security.fdb" -modify sysdba -pw london
```

Delete user Joe on local server:

```
gsec -user SYSDBA -password masterkey -delete joe
```

Notes: On InterBase® systems, the security database is named isc4.gdb. There will be a warning when a new password is longer than 8 characters.

See also:

- Firebird's password file utility
- Firebird 2.5 Release Notes: gsec improvements
- Firebird 2.5 Quick Start Guide: Server configuration and management
- Security in Firebird 2
- User Manager
- Grant Manager
- Change user password per batch

back to top of page

gfix

(gfix.exe)

gfix is an Firebird/InterBase® command-line utility, offering a number of options to validate and repair databases. gfix is fully documented in Firebird Database Housekeeping Utility.

All these options are included in the IBExpert menu items Services / Database Validation and Database Properties.

New to Firebird 2.0: New shut-down states (modes) were introduced in Firebird 2.0. Please refer to Firebird 2.0.4 Release Notes: gfix server utility below for the full syntax and examples.

The following articles are published here with the kind permission of Stefan Heymann (https://www.destructor.de/).

General Syntax

```
gfix [options] -user <username> -password <password> <database> [options]
```

Should your database ever suffer from corruption, we recommend taking the following procedure:

- Copy your database file somewhere safe: employee.gdb database.gdb
- Validate database: gfix -v -full database.gdb
- On error try to mend: gfix -mend -full -ignore database.gdb
- Check again: gfix -v -full database.gdb
- On error try backup without garbage collection: gbak -backup -v -ignore -garbage database.gdb database.gbk
- Finally try a restore: gbak -create -v database.gbk database.gdb

Further information and examples can be found under the following subjects:

- Database shutdown using gfix
- Database repair and sweeping using gfix
- gfix - miscellaneous parameters
- Using gfix
- Firebird database housekeeping utility
- New to Firebird 2

Database shutdown using gfix

by Stefan Heymann

Database Shutdown

When a database has been shut down, only SYSDBA and the database owner are able to connect to the database in order to perform administrative tasks.

Options

- -at[tach] <seconds> Used with the -shut option. Waits <seconds> seconds for all current connections to end. If after <seconds> seconds there are still connections open, the shutdown will be cancelled.
- -f[orce] <seconds> Used with the -shut option. Waits <seconds> seconds for all connections and transactions to end. After this time, all connections and transactions are cancelled and the database is shut down. Use with caution.
- -o[nline] If a -shut operation is pending, it is cancelled. Otherwise, takes a database back online.
- -sh[ut] Shut down database. Must be used together with -attach, -force or -tran.*
- -tr[an] <seconds> Used with the -shut option. Waits <seconds> seconds for all running transactions to end. If after <seconds> seconds there are still running transactions, the shutdown will be cancelled.

* Firebird 2.0 introduced new -shutdown parameters to be used with gfix. Please refer to the Firebird 2.0.4 Release Notes chapter, Command-line utilities or below for further information.

Examples

Shut down database, wait 60 seconds until all connections are closed:

```
gfix -user SYSDBA -password "masterkey" dbserver:/db/mydb.fdb -shut -attach
60
```

Note that gfix will terminate with an error if there are still connections open after 60 seconds.

Shut down database, force shutdown after 60 seconds:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -shut -force 60
```

Shut down database, force shutdown NOW:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -shut -force 0
```

Put database online again:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -online
```

See also:

- Database shutdown

- Active users
- Firebird database housekeeping utility

back to top of page

Database repair and sweeping using gfix

by Stefan Heymann

Options

- -f[ull] Use with the -v option. Examines all records and pages and releases unassigned record fragments.
- -h[ousekeeping] 0 Switch off automatic sweeping.
- -h[ousekeeping] <n> Set sweep interval to <n> transactions (default is 20000).
- -i[gnore] Ignores checksum errors during a validate or sweep.
- -m[end] Marks corrupt records as unavailable so they are skipped on a subsequent backup.
- -n[o_update] Use with the -v option. Examines all records and pages and reports errors but does not repair them.
- -s[weep] Forces an immediate sweep.
- -v[alidate] Check database for validity. At the same time, errors are reported and repaired.

Examples

Validate database:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -v -f
```

Sweep database now:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -s
```

Set sweep interval to 50000 transactions:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -h 50000
```

Switch off automatic sweeping:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -h 0
```

See also:

- Repairing a corrupt database
- Database sweep/Sweep interval
- Firebird for the Database Expert: Episode 4 - OAT, OIT and Sweep
- Database validation

back to top of page

gfix - miscellaneous parameters

by Stefan Heymann

Options

- -b[uffers] <pages> Default cache buffers for the database will be set to <pages> pages.
- -c[ommit] <id> Commits limbo transaction specified by the given <id>.
- -c[ommit] all Commits all limbo transactions.
- -k[ill] Drops shadows and unavailable shadows. Syntax is gfix -k (no database name).
- -l[ist] Display IDs of all Limbo transactions and what would happen to each transaction if you would use -t on it.
- -mo[de] read_write Set mode of database to read/write (default). Requires exclusive access to database (shutdown).
- -mo[de] read_only Set mode of database to read-only. Requires exclusive access to database (shutdown).
- -pa[ssword] <password> Database password.
- -p[rompt] Use with -l. Prompts for action.
- -r[ollback] <id> Rolls back limbo transaction specified by the given <id>.
- -r[ollback] all Rolls back all limbo transactions.
- -s[ql_dialect] 1 Sets SQL dialect 1 for the database.
- -s[ql_dialect] 3 Sets SQL dialect 3 for the database.
- -t[wo_phase] <id> Performs automated two-phase recovery for limbo transaction with the given <id>.
- -t[wo_phase] all Performs automated two-phase recovery for all limbo transactions.
- -user <name> Database username.
- -w[rite] sync Enables forced writes.
- -w[rite] async Disables forced writes.
- -z Show gfix and server version.

Examples

Set database to read-only:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -shut -attach 60g

gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -shut -force 0

gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -mode read_only

gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -online
```

Set database to SQL dialect 3:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -sql_dialect 3
```

Enable forced writes:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -write sync
```

Disable forced writes:

```
gfix -user SYSDBA -password masterkey dbserver:/db/mydb.fdb -write async
```

See also:

- Database Corruption
- Database validation

back to top of page

New gfix -shut[down] options in Firebird 2

The options for gfix -shut[down] have been extended to include two extra states or modes to govern the shutdown.

Syntax

```
gfix <command> [<state>] [<options>]
<command> ::= {-shut | -online}
<state> ::= {normal | multi | single | full}
<options> ::= {-force <timeout> | -tran | -attach}
```

- **normal state**: online database.
- **multi state**: multi-user shutdown mode (the legacy one, unlimited attachments of SYSDBA/owner are allowed).
- **single state**: single-user shutdown (only one attachment is allowed, used by the restore process).
- **full state**: full/exclusive shutdown (no attachments are allowed).

Note: Multi is the default state for -shut, normal is the default state for -online.

The modes can be switched sequentially:

```
normal <-> multi <-> single <-> full
```

Examples

```
gfix -shut single -force 0
gfix -shut full -force 0
gfix -online single
gfix -online
```

You cannot use -shut to bring a database one level more "online" and you cannot use -online

to make a database more protected (an error will be thrown).

These are prohibited:

```
gfix -shut single -force 0
gfix -shut multi -force 0
gfix -online
gfix -online full
gfix -shut -force 0
gfix -online single
```

Source: Firebird 2.0.4 Release Notes: gfix server utility

back to top of page

gstat

(gstat.exe)

gstat is an Firebird/InterBase® command-line utility, which can be used to display database statistics related to transaction inventory, data distribution within a database, and index efficiency. The Firebird gstat utility is fully documented in the Firebird Database Statistics Reporting Tool.

All information offered by this tool can be found in the IBExpert Services menu item, Database Statistics.

New to Firebird 2.5: gstat now has -? and -help switches for requesting help about available switches and arguments.

gstat should be called as follows:

```
gstat database_name [switches]
```

It is also possible to call it as follows:

```
gstat [switches] database_name
```

However, although it does work this way, problems arise when the -t[able] switch is used.

```
Available switches:
* <color #c3c3c3>-a</color> analyze data and index pages
* <color #c3c3c3>-d</color> analyze data pages
* <color #c3c3c3>-h</color> analyze header page
* <color #c3c3c3>-i</color> analyze index leaf pages
* <color #c3c3c3>-s</color> analyze system relations
* <color #c3c3c3>-u</color> username
* <color #c3c3c3>-p</color> password
* <color #c3c3c3>-r</color> analyze average record and version length
* <color #c3c3c3>-t</color> tablename
* <color #c3c3c3>-z</color> display version number
```

Note: In Firebird versions prior to 2.0, the -l[og] switch could also be used. This reported on the details of the logging page(s) within the database. The logging pages have not been used for some time and the switch has now been removed from gstat.

fbtracemgr

A new command-line utility, named fbtracemgr, was added in Firebird 2.5 for working interactively with trace services. It has its own syntax of switches and parameters, discussed in detail, with examples, in the Firebird 2.5 Release Notes chapter, Command-line utilities.

All functionalities offered by fbtracemgr can be found in the IBExpert Trace and audit menu.

Usage is as follows:

```
fbtracemgr <action> [<parameters>]
```

Action switches

- -STA[RT] Start trace session
- -STO[P] Stop trace session
- -SU[SPEND] Suspend trace session
- -R[ESUME] Resume trace session
- -L[IST] List existing trace sessions

Parameters

Action parameters

- -N[AME] <string> Session name
- -I[D] <number> Session ID
- -C[ONFIG] <string> Trace configuration file name

Connection parameters

- -SE[RVICE] <string> Service name
- -U[SER] <string> User name
- -P[ASSWORD] <string> Password
- -FE[TCH] <string> Fetch password from file
- -T[RUSTED] <string> Force trusted authentication

Examples using fbtracemgr

```
fbtracemgr -SE remote_host:service_mgr -USER SYSDBA -PASS masterkey -LIST
fbtracemgr -SE service_mgr -START -NAME my_trace -CONFIG my_cfg.txt
fbtracemgr -SE service_mgr -SUSPEND -ID 2
fbtracemgr -SE service_mgr -RESUME -ID 2
fbtracemgr -SE service_mgr -STOP -ID 4
```

Notes:

1. All switches and parameter identifiers are case-insensitive.
2. To stop an interactive trace session on any platform, press [Ctrl + C].

As well, the general *Services* utility, fbsvcmgr, can be used for submitting service requests from the command line, as exemplified in the following examples.

a. Start a user trace named My trace using a configuration file named fbtrace.conf and read its output on the screen:

```
fbsvcmgr service_mgr action_trace_start trc_name "My trace" trc_cfg
fbtrace.conf
```

To stop this trace session, press [Ctrl + C] at the fbsvcmgr console prompt. (See also (e), below).

b. List trace sessions:

```
fbsvcmgr service_mgr action_trace_list
```

c. Suspend trace sesson with ID 1

```
fbsvcmgr service_mgr action_trace_suspend trc_id 1
```

d. Resume trace sesson with ID 1

```
fbsvcmgr service_mgr action_trace_resume trc_id 1
```

e. Stop trace sesson with ID 1

```
fbsvcmgr service_mgr action_trace_stop trc_id 1
```

Tip: List sessions (see b.) in another console, look for the ID of a session of interest and use it in the current console to stop the session.

back to top of page

Further reading:

- Firebird's nbackup tool
- Firebird Password File Utility
- Firebird Backup & Restore Utility
- Firebird Database Housekeeping Utility
- Firebird Backup File Splitting Filter
- Firebird Database Statistics Reporting Tool
- Firebird 2.5 Language Reference Update
- Firebird 2.1 Language Reference Update
- Firebird 2 Language Reference Update

- Firebird administration
- Firebird SQL Server 2.x Administration Handbook: Administration tools
- Firebird 2.0.4 Release Notes
- Firebird 2.1 Release Notes
- Firebird 2.5 Release Notes: Command-line utilities
- Firebird 2.5 Quick Start Guide: Default disk locations
- Transaction options explained
- Firebird SQL Server 2.x Administration Handbook

From:
http://ibexpert.com/docu/ - **IBExpert**

Permanent link:
http://ibexpert.com/docu/doku.php?id=01-documentation:01-10-firebird-command-line-utilities:command-line-utilities

Last update: **2023/08/13 17:59**