Firebird Superserver Manager

Norman Dunbar

26 October 2009 – Document version 1.4

Table of Contents

Introduction.

Fbmgr is used to start and stop your Firebird Superserver. This is different from starting and stopping a database - as explained in the gfix manual - as the running databases rely on the Firebird service/daemon. This daemon is the Firebird *engine* - without it, nothing will run under Superserver.

> **Note**
>
> If you are using Firebird Classic, this chapter does not apply and it is entirely possible that the executables mentioned in this chapter may not even exist in your installation directory.

On Windows systems, the engine runs as a service and is slightly different. Windows will be discussed separately in this manual for that reason.

In the remainder of this manual, we will discuss the following:

- Differences between Windows and Linux systems.
- Command-line options for fbmgr.
- Fbmgr commands and their parameters.
- Running fbmgr in interactive or batch modes.
- Some caveats, gotchas and foibles of fbmgr.

Linux Systems.

On Linux, the Firebird engine is started whenever you carry out an install - from an RPM or via a script - and at system boot time. This means that Firebird is available whenever your system is running once you have Firebird installed.

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/rcfire-bird` 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. The engine is started under the watchful eye of the guardian process (fbguard) and is set to automatically restart if the engine fails for any reason. The engine runs under the firebird user even though you have to be root to run the **rcfirebird** script.

- **stop**

Stops the Firebird engine. If already stopped, does nothing. It kills the process for the running engine by reading the pidfile (by default `/var/run/firebird/firebird.pid`). If the process is not running, no error is reported.

- **status**

Checks the current status of the Firebird engine. This command can return a status code (in $?) as follows:

```
tux> rcfirebird status ; echo $?
```

- 0 - the engine is running.
- 1 - the engine was not running but the pidfile was found.
- 2 - the engine was not running but the lock file was found.
- 3 - the engine was not running.

> **Note**
>
> The **status** parameter is not available on every Linux distribution. You can check if it is available on your installation by running the **rcfirebird** command with no parameters:
>
> ```
> tux> rcfirebird
> Usage: /usr/sbin/rcfirebird {start|stop|status|try-restart|restart|
> force-reload|reload}
> ```
>
> The command will list all the options available for your particular installation.

- **try-restart**

Restarts the Firebird engine but only 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.

It might be necessary for you to manually shut down the engine from time to time, and to restart it again afterwards. To do this, and assuming that either none of the options to **rcfirebird** are suitable or you do not have root privileges, fbmgr is your tool. Fbmgr is, by default, available to all users of the system to execute - however, in order to start and stop the Firebird engine, root or SYSDBA rights are required.

> **Caution**
>
> It is noted that the default shell for the firebird user (at least of Suse systems) defaults to /bin/false and so, you cannot log in as the firebird user. This means that you must use **rcfirebird** to start and stop the engine because if you use fbmgr as root, then all databases created by the engine will be owned by root.
>
> In order to allow the firebird user the ability to start and stop the database engine, you need to alter the default shell from /bin/false to something like /bin/bash *and* set a password, while logged in as root, as follows:
>
> ```
> tux> usermod --shell /bin/bash firebird
> tux> passwd firebird
> ```

Fbmgr is a very short shell script which currently - as of version 2.1.3 - exports the FIREBIRD environment variable and calls fmbgr.bin passing all supplied parameters over, so fbmgr.bin does all the hard work.

As the two are interchangeable, I shall use the former (and shorter!) fbmgr in the remainder of this chapter.

Fbmgr Commands.

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.

> **Note**
>
> When you log in as the firebird user, the home directory ($HOME) is set to be /opt/firebird. The bin subdirectory, where all the firebird binaries live, is immediately beneath $HOME.

Getting Help.

To see a list of all the commands - except, for some reason, the **start** command - run fbmgr and pass the **-help** command (in batch mode) or type **help** at the prompt in interactive mode. The resulting output is the same regardless, including the grammatical error "*... also be used as an option switches ...*":

```
tux> fbmgr -help

Usage:          fbmgr -command [-option [parameter]]

or              fbmgr<RETURN>
                FBMGR> command [-option [parameter]]

                shut  [-now]           shutdown server
                show                   show host and user
                user <user_name>       set user name
                password <password>    set DBA password
                pidfile <filename>     file to save fbserver's PID
                help                   prints help text
                quit                   quit prompt mode

Command switches 'user' and 'password' can also be used
```

```
as an option switches for commands like start or shut.
For example, to shutdown server you can:

fbmgr -shut -password <password>

or

fbmgr<RETURN>
FBMGR> shut -password <password>

or

fbmgr<RETURN>
FBMGR> password <password>
FBMGR> shut
```

Starting Up.

When the Firebird engine is started, it normally runs under the watchful eyes of the guardian process. The guardian will restart the engine any time it determines that the engine has crashed and in doing so, will hopefully reduce the downtime that the users may suffer as a result of a crashed engine.

It is possible, however, to force the engine to be started up and the guardian will allow it to stay down if it detects a crash. Fbmgr allows the DBA or System Administrator to decide which of the two startup methods will be used.

Start And Stay Running.

At system boot time, the engine is started in the mode that allows the guardian to restart it in the event of a crash. To perform this task manually using fbmgr you would carry out one of the following:

```
tux> fbmgr -start -forever
server has been successfully started
```

Or, in interactive mode:

```
tux> fbmgr
FBMGR> start -forever
server has been successfully started
```

If you are logged in as a *privileged user* then you don't need to specify a **-user** as these login accounts default to SYSDBA when fbmgr is run. If you log in as any other user, even supplying a user name will not be enough to allow you to restart a closed database engine.

> **Note**
>
> A privileged user is one that the Firebird engine considers to be privileged enough to automatically be given SYSDBA rights. This means that it can start and stop the engine without being required to authenticate as SYSDBA. At present there are four login names that are assumed to be privileged, these are:
>
> - root
> - firebird
> - interbase
> - interbas (without the 'e')

The **start** command defaults to **-forever** if nothing is specified.

```
tux> fbmgr -start
server has been successfully started
```

Start And Stop Running.

Under normal circumstances you would wish for the database engine to remain running as long as possible. At other times, however, you may wish for any crashes to be investigated prior to restarting the engine. This is possible using fbmgr as the following shows:

```
tux> fbmgr -start -once
```

Or, running interactively:

```
tux> fbmgr
FBMGR> start -once
```

Shutting Down.

Shutting down the engine stops the guardian process from restarting it. If this was not the case, it would be very difficult to actually stop the Firebird engine!

> **Warning**
>
> Any user who has logged on to the database server and who is armed with the SYSDBA password, can close the engine down. This is a slight inconsistency as only the privileged users can start the engine.

If you are logged in as a privileged user, you do not need to supply a user name to shut down the engine, you are only required to supply the SYSDBA password.

```
tux> fbmgr -shut -password secret
server shutdown completed
```

You *are* required to supply a user name if you log in as your own account.

```
tux> fbmgr -shut -password secret
no permissions to perform operation

tux> fbmgr -shut -user sysdba -password secret
server shutdown completed
```

Showing Details.

The show command displays the name of the server that you are running on and details of the user you are currently using. If you are running fbmgr as a privileged user, then the user will be SYSDBA unless you specified a different one with the **-user** parameter.

```
tux> fbmgr -show
```

```
Host:    localhost
User:    SYSDBA
```

You can supply a different user name on the command line if desired:

```
tux> fbmgr -show -username  norman

Host:    localhost
User:    NORMAN
```

In interactive mode, the process is almost identical:

```
tux> fbmgr
FBMGR> show

Host:    localhost
User:    SYSDBA
```

Or, using a different user name:

```
tux> fbmgr -user norman
FBMGR> show

Host:    localhost
User:    NORMAN
```

It appears that the **show** command always displays the hostname as localhost while you are logged onto that server. It is unfortunate that the **show** command doesn't show any details about the running (or otherwise) Firebird engine. You can find this out as follows:

```
tux> ps -ef|grep -i fire[b]ird
firebird  3752     1  0 14:13 ?        00:00:00 /opt/firebird/bin/fbguard -o
firebird  3753  3752  0 14:13 ?        00:00:00 /opt/firebird/bin/fbserver
```

Look closely at the process details for the guardian, you can see a **-o** parameter. This indicates that the engine is running and is in **-once** mode. If it crashes at any time, it will not be restarted by the guardian. In **-forever** mode, the parameter is **-f**.

If you don't see either the guardian or the server processes, then you can assume that the (Superserver) engine is not running.

> **Tip**
>
> When using **grep** to filter out the interesting processes from a call to **ps -ef**, for example, using a character class for one single character prevents the **grep** process itself from being listed. The following two grep commands produce identical results but using the character class in the first saves having to call **grep -v grep** to filter out the unwanted **grep** process.
>
> ```
> ps -ef | grep -i fire[b]ird
> ps -ef | grep -i firebird | grep -v grep
> ```

Exiting.

The **quit** command gets you out of an interactive session. There is no quit command for batch mode.

Windows Systems.

On Windows 2000/2003, Vista and NT systems as well as the non-home versions of XP, the Firebird engine runs as a service as does the Firebird Guardian. Both of these run under the local system user account. For best results and stability, you are advised to keep Firebird running as a service rather than as an application.

After installation, these two services are defined to be started and stopped automatically on server boot and shutdown. If you wish to control when the services are started and stopped, use Control Panel to change their properties to manual.

On Windows ME, 95, 98 and XP Home, the engine runs as an application as does the guardian. In this case, an icon will be seen in the system tray and you can carry out manual maintenance by right clicking on the icon.

> **Note**
>
> The remainder of this section assumes that you have the Firebird engine and the guardian running as services. All the engine commands are accessed via a right-click on the guardian icon in the system tray when running as an application.

On Windows, the guardian process is the file `fbguard.exe` and the engine itself is the file `fbserver.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. When discussing the **instsvc** command below, it will be used only to start and stop services rather than installing and removing them.

Starting The Engine.

When you start the guardian service on Windows, it will automatically bring up the database engine without you needing to physically start it too.

Using Net Start.

Provided you know the exact service name, you can use the **net** commands to start (and stop) the guardian and database engine. Up until Firebird 2.0, the service name was simply "FirebirdServer". From Version 2.1 onwards, there can be many different services running. In this section, only the default services for Firebird 2.x will be discussed and these are "Firebird Guardian - DefaultInstance" and "Firebird Server - DefaultInstance". As there are spaces in the names, double quotes are required to prevent errors.

```
C:\>net start "firebird guardian - defaultinstance"
The Firebird Guardian - DefaultInstance service is starting.
The Firebird Guardian - DefaultInstance service was started successfully.
```

Using Instsvc.

As you can see from the above, under Firebird 2.x, the service names are a bit unwieldy. It is easier to use **instsvc** to start the services as you do not need to know the default service names.

```
C:\>instsvc start
Service "Firebird Guardian - DefaultInstance" successfully started.
```

Stopping The Engine.

When you stop the guardian service on Windows, it will automatically bring down the database engine without you being required to stop it too.

Using Net Stop.

As with starting the services, in order to use **net stop** to bring the guardian and engine down, you need to know the exact service name (for the guardian) and enclose it in quotes if there are spaces in the name.

```
C:\>net stop "firebird guardian - defaultinstance"
The Firebird Guardian - DefaultInstance service is stopping.
The Firebird Guardian - DefaultInstance service was stopped successfully.
```

Using Instsvc.

As with starting the default services, stopping them is far easier using **instsvc**.

```
C:\>instsvc stop
Service "Firebird Guardian - DefaultInstance" successfully stopped.
```

Querying The Engine.

It is possible to query the services running on the server to check whether the Firebird services are running or not. Once again, the **instsvc** command is used:

```
C:\>instsvc q

Firebird Guardian - DefaultInstance IS installed.
  Status  : stopped
  Path    : C:\Program Files\Firebird\Firebird_2_0\bin\fbguard.exe -s
  Startup : manual
  Run as  : LocalSystem

Firebird Server - DefaultInstance IS installed.
  Status  : stopped
  Path    : C:\Program Files\Firebird\Firebird_2_0\bin\fbserver.exe -s
  Startup : manual
  Run as  : LocalSystem
```

It can be seen from the above, that there are two services installed on this server, but none of them are currently running. Other details are displayed such as whether or not these services are started automatically on reboot - in the example above, this is not the case - and the user account (LocalSystem) under which the services will be run, when they are started.

This command is very useful as it displays the two service names - once you have that information, you may use the **net stop** or **net start** commands to bring the database down and to restart it.

Fbmgr Caveats

The following caveats are all specific to the Linux (and by assumption, other Unix systems) and do not apply to Windows versions of Firebird. The reason is simple, Windows doesn't have an fbmgr application as it uses the **instsvc** command instead.

The Help Command.

The `help` command, as described above, doesn't mention that fact that there is a `start` command. There is a very brief mention of the `start` command, but no text explains its use, unlike the other commands which are described.

Who Owns The Databases?

While not an error as such, this is something that you should be aware of. When the server boots, it starts the Firebird engine and runs it as the user firebird. By default, the firebird has no interactive shell - it defaults to `/bin/false` - and so you cannot log in to, or **su** to, the firebird user to manually run a command.

This means that, unless you change the firebird user's default shell as described above, you will have to carry out any manual restarts etc of the engine as the root user - although you can shut it down as any user provided you have the SYSDBA password.

If you do restart the engine as root, then it will now be running as root. While existing databases - owned by the firebird user - will happily be read from and written to by root, any new databases created will be owned by root. Everything will work fine for a while but after the server is next rebooted, these new databases will then fail to be accessible as the engine is now running as the firebird user again and that user has no permissions to access the databases owned by root.

You are therefore advised to always start and stop the server either:

- as root, using the **rcfirebird** command; or
- as firebird, using fbmgr.

but never, ever, as root using fbmgr. If you follow the above instructions, all your databases will be owned by the firebird user.

What's That Defunct Guardian Process Doing?

If you run fbmgr and shut down the database engine, the `fbserver` process vanishes from the list of processes running under the firebird user. However, a **ps -ef|grep -i fir[e]bird** command will show the following:

```
tux> ps -ef|grep -i fir[e]bird
```

```
firebird 29978 29844  0 15:57 pts/0     00:00:00 /opt/firebird/bin/fbmgr.bin
firebird 29979 29978  0 15:57 ?         00:00:00 [fbguard] <defunct>
root      29992 29955  0 15:57 pts/1     00:00:00 grep -i fire
```

The guardian process, fbguard, doesn't vanish until you exit from fbmgr. This again is not a major problem, but knowing that all you have to do is exit from fbmgr to make it vanish is helpful.

I Can Shut Down But Not Start Up The Engine.

If you log in to the server as a privileged user and have the SYSDBA's password, you can shut down and start up the engine with no problems. If, on the other hand, you log in as any other user and try to use fbmgr to stop and start the Firebird engine, you will find that you can stop it with no problems but you will not be allowed to restart it. Only privileged users can start the engine. The following shows the problem.

```
tux> # Shutdown Firebird engine as user 'norman'.
tux> fbmgr -shut -password secret
no permissions to perform operation

tux> fbmgr -shut -user sysdba -password secret
server shutdown completed


tux> # Try to restart the Firebird engine as user 'norman'.
tux> fbmgr -start
no permissions to perform operation

tux> fbmgr -start -password secret
no permissions to perform operation

tux> fbmgr -start -user sysdba -password secret
no permissions to perform operation


tux> # Give up and restart the engine as 'firebird'.
tux> su - firebird
tux> bin/fbmgr -start
server has been successfully started
```
It can be seen from the above that a normal user armed with the SYSDBA user name and password can stop the Firebird engine but is completely unable to restart it afterwards.

The reason why this is the case is quite simple. Until the firebird engine is started, there is no way to check that the password supplied for the SYSDBA user is actually correct. Because of this, logging in as a privileged user assumes SYSDBA privileges and allows you to start the engine.

Shut with no parameters appears to be a bug.

Running fbmgr's -shut command with no user or password gives the following cryptic output:

```
tux> fbmgr -shut
Invalid clumplet buffer structure: buffer end before end of clumplet -
no length component
can not attach to server
```

The same result is obtained even with a user name supplied:

```
tux> fbmgr -shut -user sysdba
Invalid clumplet buffer structure: buffer end before end of clumplet -
no length component
can not attach to server
```

Appendix A:
Document history

The exact file history is recorded in the manual module in our CVS tree; see http://sourceforge.net/cvs/?group_id=9028. The full URL of the CVS log for this file can be found at http://firebird.cvs.sourceforge.net/viewvc/firebird/manual/src/docs/firebirddocs/fbutil_fbmgr.xml?view=log

Revision History

1.0	29 September 2009	ND	Created as a chapter in the Command Line Utilities manual.
1.1	11 October 2009	ND	Some screen output wrapped as it extended out of the page in the generated pdf.
1.2	20 October 2009	ND	Replaced all references to "root or firebird" account names with "privileged users" as there are more than just these two accounts. Converted to a stand alone manual.
1.3	21 October 2009	ND	A few spelling mistakes corrected.
1.4	26 October 2009	ND	The **status** option to the **rcfirebird** command is not available on all Linux distros. Text amended to warn the reader of this discrepency.

Appendix B:
License notice

The contents of this Documentation are subject to the Public Documentation License Version 1.0 (the "License"); you may only use this Documentation if you comply with the terms of this License. Copies of the License are available at http://www.firebirdsql.org/pdfmanual/pdl.pdf (PDF) and http://www.firebirdsql.org/manual/pdl.html (HTML).

The Original Documentation is titled *Firebird Superserver Manager*.

The Initial Writer of the Original Documentation is: Norman Dunbar.

Copyright (C) 2004–2009. All Rights Reserved. Initial Writer contact: NormanDunbar at users dot sourceforge dot net.