

Configuring Firebird

Before we take a look at the two Firebird configuration files, we would like to point out that the most frequently asked question regarding these subjects is, "I've changed the parameter in the firebird.conf/aliases.conf and nothing's happened!" The simple solution is: remove the hash (#)! It's the symbol used for commenting.

aliases.conf

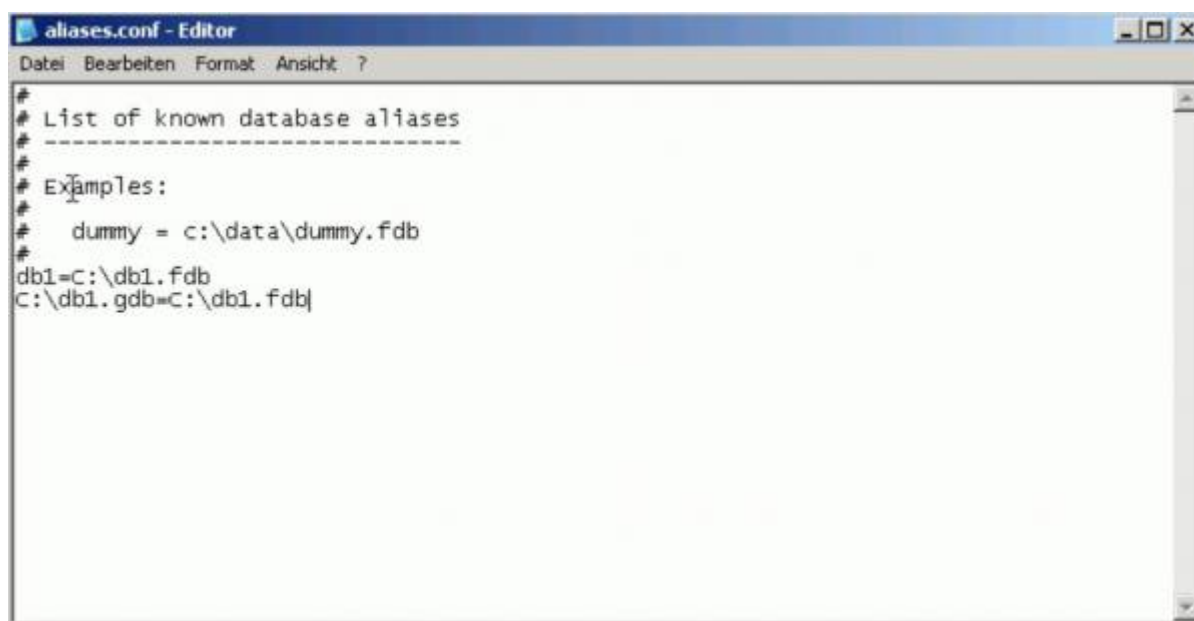
An [alias](#) is a pseudonym for the [database connection string](#) and database file name. The full connection string usually consists of the server name (or localhost) followed by the drive and path to the database file, with the database file name concatenating on the end. This informs the client, where he needs to send his [data](#) packets and access server data.

For security reasons it is not always desirable for each client user to see the full connection string, and there are obvious problems which arise when the database is moved to another drive or machine, as each client has to be informed of the new connection string. For these reasons it is recommended to give databases an alias name. All alias names are set in aliases.conf. There are no syntactical restrictions to the naming of aliases.

Using an alias, users are not able to see where the database really is and, should it be relocated, the new connection string only needs to be altered once in the aliases.conf. Let's look at an example:

The alias db1 should refer to the database name, db1.fdb.

```
db1=c:\path\db1.fdb
```

A screenshot of a text editor window titled "aliases.conf - Editor". The window contains the following text:

```
# List of known database aliases
# -----
# Examples:
# dummy = c:\data\dummy.fdb
#
db1=c:\db1.fdb
C:\db1.gdb=c:\db1.fdb
```

This user alias has been specified for the database server. The client can also define such an alias connection when [registering the database](#) or subsequently in the IBEExpert's [Database Registration Info](#). The connection string is:

```
servername:aliasname
```

If the user wishes to connect to db1, he simply needs to enter

```
localhost:db1
```

in the [Database Alias](#) field. The aliases.conf file shows the server which database the client wishes to connect to.

When working with IBExpert, a database alias can be specified when registering the database. Refer to [Register Database / Alias](#) for further information.

Resolving the XP Windows System Restore problem

Windows XP has the unfortunate tendency to consider all files with the .GDB suffix to be a constituent of the Windows System Restore. This means that when you try to open your DB1.GDB, XP (default setting) first decides to make a copy of the file (just in case you need to restore it at some point), not allowing you access until it's completed. In the case of large database files, you can imagine how long this can take!

If you don't want to rename your database files just to suit Microsoft, then simply create an alias:

```
C:\db1.gdb = C:\db1.fdb
```

[back to top of page](#)

firebird.conf

Possible file locations are set in firebird.conf. The full set of firebird.conf parameters are described in detail in the [firebird.conf](#) file. The server needs to be restarted following any changes made in the firebird.conf for them to become valid. The following describes briefly the most important parameters:

RootDirectory

If you are using several installations of Firebird servers, use the RootDirectory parameter to specify where the active Firebird server can be found.

DatabaseAccess

The DatabaseAccess parameter in firebird.conf can be set to Restrict to limit access to explicitly listed file system trees, or even to None to allow access to aliased databases only. Default is All, i.e. no restrictions. An [alias](#) entry needs to exist. If a path is entered here, [database files](#) may only be stored in this path or its subdirectories.

Note that this is not the same thing as the [file system-level access protection](#): when DatabaseAccess is anything other than All, the server will refuse to open any databases outside the defined scope even if it has sufficient rights on the database files.

```
DatabaseAccess = NONE
```

means that only file locations set in [aliases.conf](#) are available. The server can't access any other entries. This is a great security feature, because even when someone has a user name on the database server, he cannot create a database file, because it is not possible to specify an alias remotely.

ExternalFileAccess

Firebird has a mechanism enabling a [table](#) to be created externally, (i.e. not in the [database](#)), using the command:

```
create table external file
```

In order to allow such external files it is necessary to explicitly activate the ExternalFileAccess parameter. Options include: None, Full or Restrict. If you choose Restrict, provide a ';'-separated trees list, where external files are stored. Default value None disables any use of external files on your site.

UdfAccess

[User-defined functions](#) are used in Firebird to complement and extend the Firebird server's language. This parameter specifies where UDFs can be found. They are usually to be found in the subdirectory /UDF, and should - if possible - remain there. UdfAccess may be None, Full or Restrict. If you choose Restrict, provide a ';'-separated trees list, where UDF libraries are stored.

[back to top of page](#)

TempDirectories

Here you can specify where [temporary files](#) should be created. When the Firebird server receives a [query](#) including ORDER BY or similar, without an index, then Firebird has to sort the [data](#) somewhere. Firebird has a so-called Sort Buffer, which is principally a memory area where such sorting processes can be performed. If however you have a sorting operation that is 10 GB, Firebird needs somewhere to do this. From a certain size, when the Sort Buffer is no longer sufficient, it moves the job out into a temporary file, and you can specify here where these temp files should be.

Because of the intense batting backwards and forwards, you need to know where your temp file is in relation to your database. As soon as you need a temp file, it's because you don't have enough RAM or you've exceeded your internal limits. By its very nature, it's going to be reading things from the database cache and wanting to put things in the temp directory. So keeping those on separate disks will make a big difference. And you want to know where they are, to see how big they're getting.

What do you do if your database crashes mid-[sort file](#)? The temp files just sit there. So if your system

hangs and you need to reboot, you could suddenly have a lot of temp files. While they're being used they have a handle on them, so if you are allowed to delete or rename them, then it's fine because they're orphans.

The default value is determined using `FIREBIRD_TMP`, `TEMP` or `TMP` environment options. Every directory item may have optional size argument to limit its storage, this argument follows the directory name and must be separated by at least one space character. If the size argument is omitted or invalid, then all available space in this directory will be used.

Examples

```
TempDirectories = c:\temp;d:\temp
```

or

```
TempDirectories = c:\temp 100000000;d:\temp 500000000;e:\temp
```

DefaultDbCachePages

This influences the cache by setting the number of pages from any one database that can be held in the cache at once. By default, the Superserver allocates 2048 pages for each database and the Classic allocates 75 pages per client connection per database. Before altering either of these values please refer to [Page size](#) and [Memory configuration](#).

See also:

[Firebird 2.5 Release Notes: FileSystemCacheThreshold](#)

[back to top of page](#)

RemoteServiceName

This is the TCP service name to be used for client database connections. It is only necessary to change either the `RemoteServiceName` or `RemoteServicePort`, not both. The order of precedence is the `RemoteServiceName` (if an entry is found in the `services. file`) and then the `RemoteServicePort`.

You don't need to change this if it's your only install.

E.g. `RemoteServiceName = gds_db`

RemoteServicePort

This is the TCP port number to be used for client database connections. It is only necessary to change either the `RemoteServiceName` or `RemoteServicePort`, not both. The order of precedence is the `RemoteServiceName` (if an entry is found in the `services. file`) then the `RemoteServicePort`.

You don't need to change this if it's your only install.

E.g. RemoteServicePort = 3052

RemoteBindAddress

Allows incoming connections to be bound to the IP address of a specific network card. It enables rejection of incoming connections through any other network interface except this one. By default, connections from any available network interface are allowed.

See also:

[Firebird 2.5 Release Notes: New configuration parameters and changes](#)

CpuAffinityMask

This parameter only applies to Superserver on Windows.

In an SMP (Symmetric Multi-Processing) system, this sets which processors can be used by the server. The value is taken from a bit map in which each bit represents a CPU. Thus, to use only the first processor, the value is 1. To use both CPU 1 and CPU 2, the value is 3. To use CPU 2 and CPU 3, the value is 6. The default value is 1. It does make sense however to allow Firebird to use at least 2 CPUs, so that if the traffic on one of them gets halted due to, for example, a query going wrong, all other traffic can use the second CPU.

```
CpuAffinityMask = 1
```

See also:

[Firebird 2.1 Release Notes: New configuration parameters and changes](#)

[back to top of page](#)

security.fdb / security2.fdb

The Firebird security database optionally specifies one or more user names, passwords, or roles for use when attaching to the database. PC clients must always send a valid user name and password. If a Firebird/InterBase® user has ISC_USER and ISC_PASSWORD environment variables set and the user defined by those variables is not in the isc4.gdb/security.fdb/security2.fdb, the user will receive the following error when attempting to view `isc4.gdb/security.fdbsecurity2.fdb` users from the local server manager connection: undefined user name and password. This applies only to the local connection; the automatic connection made through Server Manager bypasses user security. Only those names may be entered when creating a database, which already exist in the server security database ISC4.GDB, security.fdb or since Firebird 2.0 the new [security2.fdb](#) (which stores server rights; user rights for the database objects are stored in the database itself). The person creating the database becomes the database owner. Only the database owner and the SYSDBA (System Database Administrator) are allowed to perform certain operations upon the database (such as a database shutdown). Therefore if the database owner is defined as the SYSDBA, this is the only person entitled to perform these operations. Note: when a role with the name SYSDBA is created, no other users (not

even the SYSDBA) can access the database. Therefore ensure the database is created by another user already registered in the security database and not the SYSDBA. This way there are at least two users able to perform key administrative tasks. The passwords are encrypted in the ISC4.GDB, security.fdb or security2.fdb. If you insist upon using the SYSDBA name as the database owner, at least change the standard password (masterkey) to ensure at least some degree of security! The masterkey password should be changed as soon as possible after creating the database. Firebird/InterBase® verifies only the first 8 characters of a password, even if a longer word is entered, i.e. in the case of the masterkey password only “masterke” is verified. All characters following the 8th are ignored. [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. All options offered by gsec can be found in the IBExpert [User Manager](#) and [Grant Manager](#). If you need to upgrade your security database from security.fdb to security2.fdb, because Firebird 2.0 only works with security2.fdb, please refer to the IBExpert User Manager chapter, [Dealing with the new security database](#).

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

Permanent link:
<http://ibexpert.com/docu/doku.php?id=02-ibexpert:02-01-getting-started:downloading-install-firebird:configuring-firebird>

Last update: **2023/08/23 02:24**

