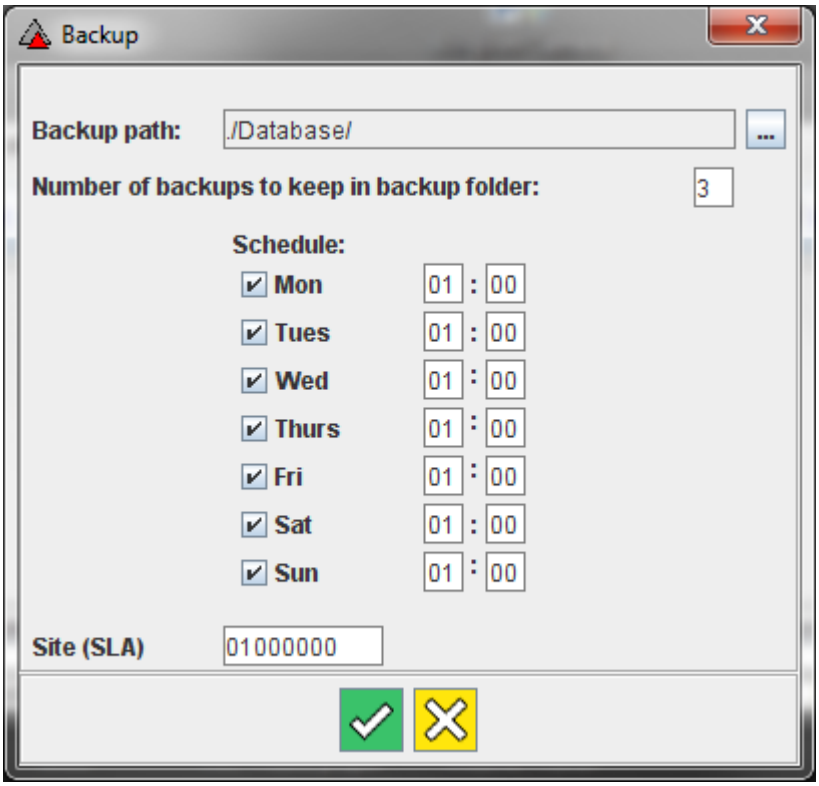
ImproNet database maintenance

The database file is very important for the ImproNet system. Without the database file any changes to the system are impossible. This is why it is very important to make sure that the database file is healthy and that backups are available. This document describes some procedures to create and restore backups, remove old data and repair damaged database files. This applies only to ImproNet systems using Firebird as database backend.

ImproNet utilities

Automatic backups

The *Engine* software has a built in automatic backup option. In *Engine* go to *Configuration* → *Backup* for the settings.



Backup path: the location where the backup files will be stored. Ideally this is a mapped network drive so the backups are not stored on the same machine.

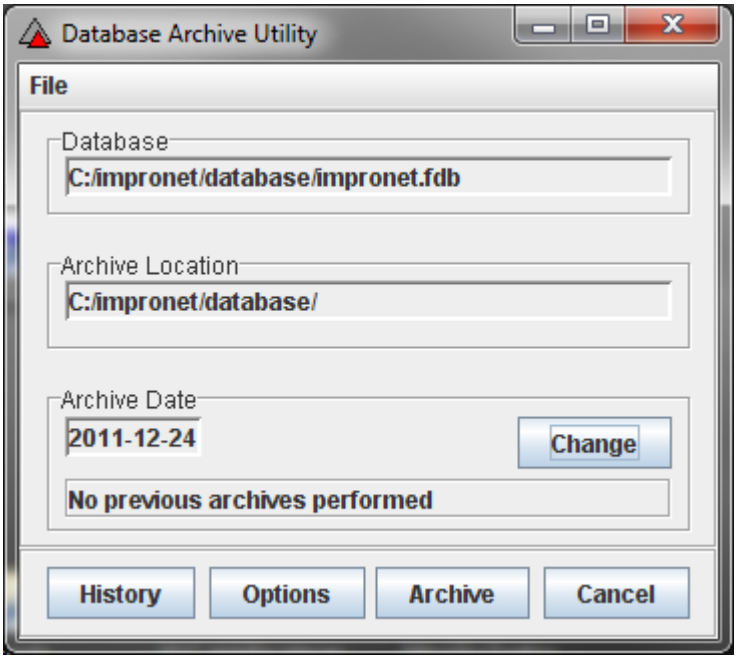
Number of backups to keep in backup folder: the number of backups to keep before older files will be removed.

Schedule: choose on what days of the week and which time the backup must be created.

Backups are created in the Firebird archive format. Please refer to the part about *IBConsole* or *Command line tools* for the procedure to restore those files.

Archive

ImproNet comes with a tool to archive old transactions and remove them from the database. It can be opened from the start menu: *Impronet* → *Utils* → *Archive*.



Archive Date: all transactions earlier than this date will be removed from the database. A copy of the database will be created in the Archive Location that still contains those transactions. Under *Options* you can choose not to archive the transactions but to delete them instead.

After removing the transactions the database file will still have the same size. The size will only be reduced after running the *Pump* utility or after performing a backup and a restore.

Pump

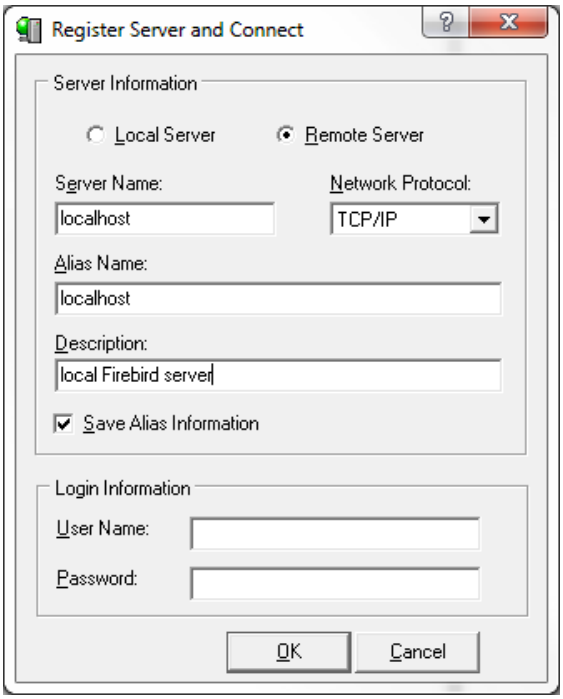
Under *Impronet* → *Utils* you will find the *Pump* utility. The *Pump* utility creates an empty database, copies the data from the existing database to it and replaces the existing database with the new one. This can fix small database problems and should reduce the size of the database file, especially after running the *Archive* utitlity. Only use the pump utility when all Impro software that might use the database is shut down.

IBConsole

IBConsole is a general tool for managing InterBase and Firebird database servers. It's included with ImproNet and you can find it under *Impronet* → *Utils* in the start menu.

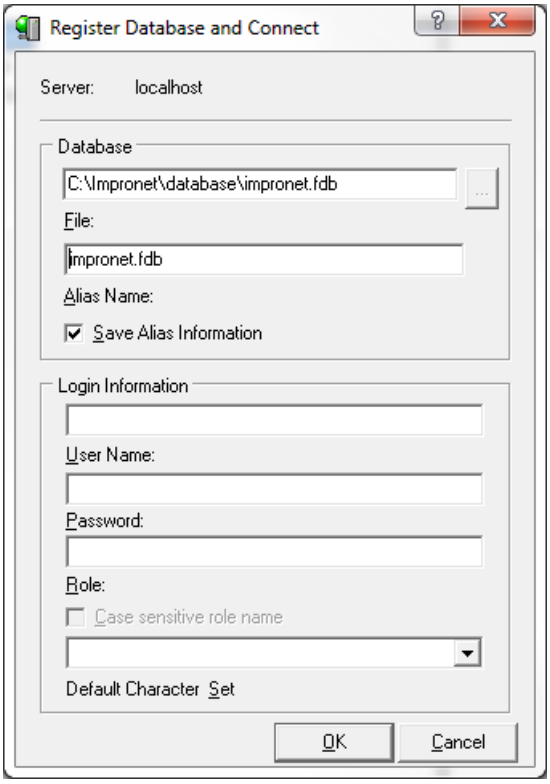
Setup

When starting IBConsole for the first time you will need to register the database server and the database file. Choose *Register..* from the *Server* menu. A Firebird server must always be registered as a remote server. See below for the settings:



You can fill in *User Name* and *Password* as well. IBConsole will then log in immediately after registering the server. Otherwise double click on the newly added server and log in. Click on the plus sign to expand the *Databases*, *Backup*, *Server Log* and *Users* icons.

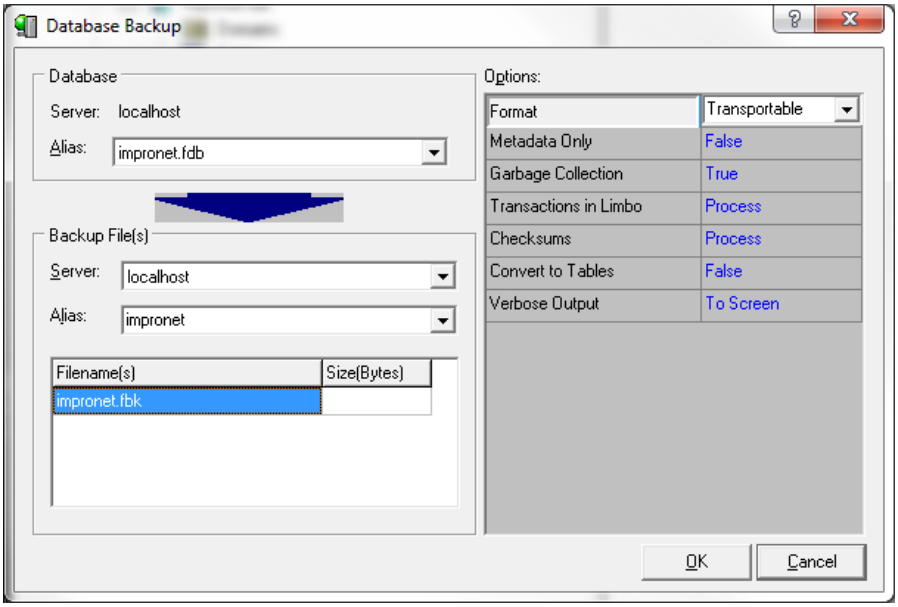
To register the ImproNet database right click on databases and choose *Register..* See below for the settings if impronet is installed in the default location:



When supplieing *User Name* and *Password* IBConsole will immediately log in to the database. Otherwise double click on the newly added database and log in.

Backup

When logged in to the server right click on the *Backup* icon and choose *Backup..* to open the backup window. The following settings will create a backup file called *Impronet.fbk* in the same directory as the Impronet database:



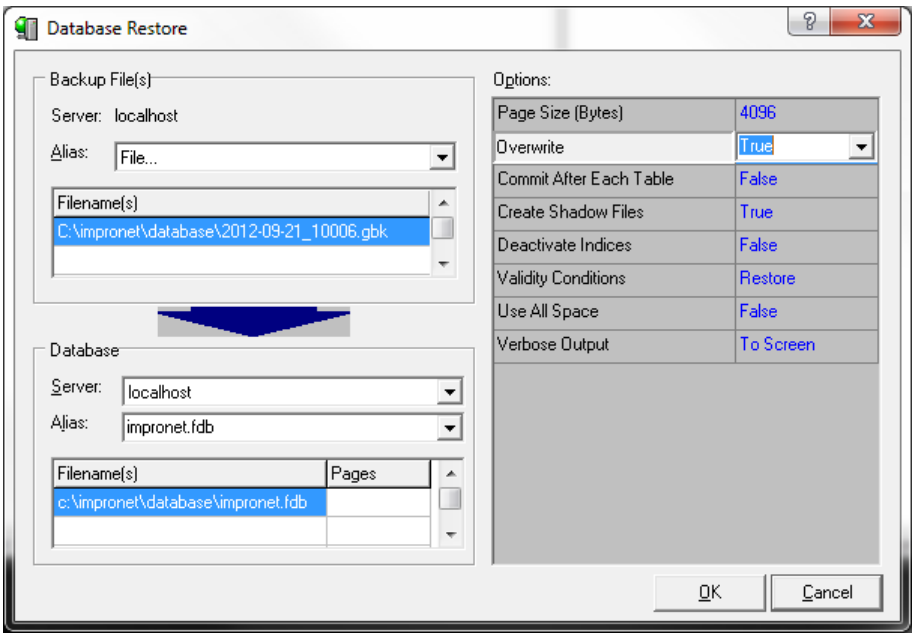
When creating a backup for the first time you'll need to enter a name under alias and fill in the filename under *Filename(s)*. The next time you can choose the alias from the list and the filename will be filled in automatically.

The files that are created this way are in the Firebird archive format. They need to be restored to the database format before they can be used again.

Restore

When logged into the server right click on the *Backup* icon and choose *Restore..* to open the restore window. To restore a backup created earlier with IBConsole you can choose this backup from the *Alias* list. Under *Database* choose the database to restore the backup to and under *Options* set *Overwrite* to *True*. Make sure that the database to restore to is not in use by any program. If you want to restore to a new database file change the filename under *Filename(s)* to something else.

To restore a backup that is not created with IBConsole, for example the backups created by *Engine* choose *File..* as alias under *Backup File(s)*. Under *Filename(s)* enter the full path to the backup file.



In this example the database will be overwritten during restore. As with the restore from a file created by IBConsole you can restore to a new file by changing the filename under *FileName(s)*.

Command-line utilities

ImproNet does include some command line utilities to perform database operations. These are located in the *C:\Impronet\bin* directory. These utilities need to be invoked from the command line or from a batch file. Open the command line by typing *cmd* in the *run* box from the windows start menu. To run the utilities without typing the full path add the *bin* directory to the *PATH* variable with the following command:

```
SET PATH=%PATH%;C:\Impronet\bin
```

Set the variables for user name and password so they don't need to be added to each command:

```
SET ISC_USER=SYSDBA
SET ISC_PASSWORD=masterkey
```

GBAK

GBAK is used to create and restore backups. The following command creates a backup of the ImproNet database called *impronet.fbk*:

```
GBAK -BACKUP_DATABASE localhost:C:\ImproNet\Database\IMPRONET.FDB
C:\ImproNet\Database\IMPRONET.FBK
```

The following command will restore the backup file:

```
GBAK -REPLACE_DATABASE C:\ImproNet\Database\IMPRONET.FBK
localhost:C:\ImproNet\Database\IMPRONET.FDB
```

Use `-CREATE_DATABASE` and a different file name to restore the backup to a new database file.

All options can be abbreviated. The following list shows a few options with the short name and between brackets the full name:

`-B(ACKUP_DATABASE)`	backup database to file
`-C(REATE_DATABASE)`	create database from backup file
`-R(EPLACE_DATABASE)`	replace database from backup file
`-V(ERIFY)`	can be used in combination with backup or restore to show output of what GBAK is doing
-P(AGE_SIZE)	Override default page size when restoring a backup. It's recommended to set this to 8192 instead of the default 1024
-G(ARBAGE_COLLECT)	Does not perform garbage collection during backup, so the backup will be faster. When you plan to do a Restore or Sweep anyway after the backup.

GFIX

GFIX can be used to repair a corrupted database and do some other administration tasks. Indications that a database is corrupted are:

- Unable to logon to ImproNet software when the database server is online
- Errors of type *org.firebirdsql.jdbc.FBSQLException* or other firebird exceptions in the ImproNet log files

The following command will mark corrupt records as unavailable so they will be skipped when performing a backup:

```
GFIX -MEND localhost:C:\ImproNet\Database\IMPRONET.FDB
```

GFIX can also be used to put a database in read only mode. When you see 'invalid transaction handle' errors in the ImproNet log files or when trying to create a backup most of the times creating a backup is still possible when the database is in read only mode. This command puts the database in read only mode:

```
GFIX -MODE READ_ONLY localhost:C:\ImproNet\Database\IMPRONET.FDB
```

After the backup and restore the database must be set back to read/write:

```
GFIX -MODE READ_WRITE localhost:C:\ImproNet\Database\IMPRONET.FDB
```

Here are some options that can be used with GFIX:

-M(END)	Marks corrupt records as unavailable so they are skipped on a subsequent backup
-I(GNORE)	Ignores checksum errors during a validate or sweep. Use this when getting error checksum messages.
-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)
-SH(UT)	Shut down database. Must be used together with -ATTACH, -FORCE or -TRAN
-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 and return an error.
-F(ORCE)	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.
-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.

Following are some examples of using GFIX and GBAK together to fix some database problems.

Restore a corrupted database

Mark corrupted records:

```
GFIX -M localhost:C:\ImproNet\Database\IMPRONET.FDB
```

Create backup:

```
GBAK -B -G -V localhost:C:\ImproNet\Database\IMPRONET.FDB
C:\ImproNet\Database\IMPRONET.FBK
```

Restore backup to a new file:

```
GBAK -P 8192 -C -V C:\ImproNet\Database\impronet.fbk
localhost:C:\ImproNet\Database\IMPRONET_RESTORED.FDB
```

Restore a database that gives 'invalid transaction handle' errors

Try to shut down any active connections. Should not be necessary when all ImproNet software is closed but sometimes a connection is not closed properly

```
GFIX -SH -AT 60 localhost:C:\ImproNet\Database\IMPRONET.FDB
```

Force shut. Fallback for when the normal shut fails

```
GFIX -SH -F 0 localhost:C:\ImproNet\Database\IMPRONET.FDB
```

Put database in read only mode

```
GFIX -MO READ_ONLY localhost:C:\ImproNet\Database\IMPRONET.FDB
```

Put database back online

```
GFIX -O localhost:C:\ImproNet\Database\IMPRONET.FDB
```

Create backup:

```
GBAK -B -G -V localhost:C:\ImproNet\Database\IMPRONET.FDB
C:\ImproNet\Database\IMPRONET.FBK
```

Restore backup to a new file:

```
GBAK -P 8192 -C -V C:\ImproNet\Database\impronet.fbk
localhost:C:\ImproNet\Database\IMPRONET_RESTORED.FDB
```

Put new file back to read/write mode

```
GFIX -MO READ_WRITE localhost:C:\ImproNet\Database\IMPRONET_RESTORED.FDB
```