# Bug Fixing Report

### 5.1. If the directory specified for a database backup doesn’t exist the backup process fails.

The bug is fixed: service reports about invalid directory:

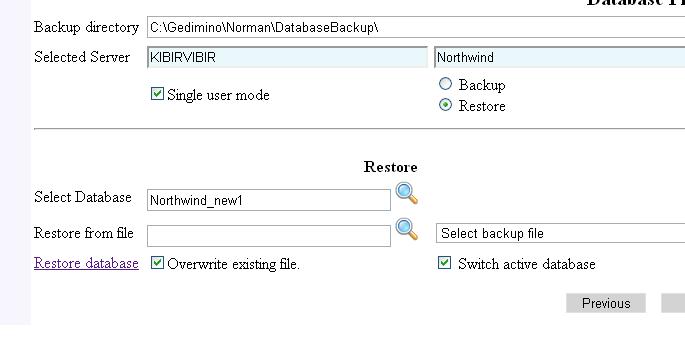


It is normal error message, not crash but a trap is here: IIS and browser are caching pages thus you are fetching the same picture after WEB.CONFIG was corrected. I killed this feature in current version and now it is enough to perform listed below procedures:

1. Correct WEB.CONFIG,
2. Refresh the page,
3. Login into database again.

### 6.1. When restoring a DB from a ‘.bak’ file the wait symbol continues to cycle even after the file has been selected.

Fixed, see attached picture:

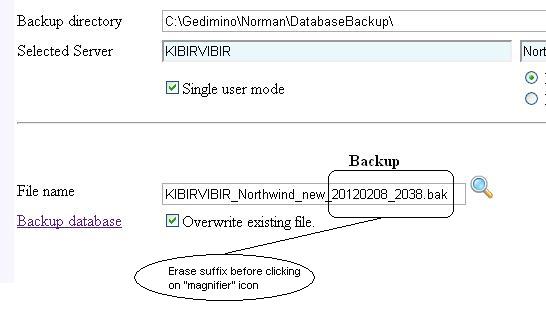


### 6.2. The restore routine doesn’t appear to be looking in the correct sub-directory for the backup file that needs to be restored. Shouldn’t this be the same as the one specified in the web-config file that the backup is written to?

Bug is fixed. The application fetches backup directory name from WEB.CONFIG, key="BackupDirectory". I did the field writable – correct it manually if someone changed WEB.CONFIG while you are working with this page.

### 6.3. If the backup folder contains more than one backup file there is no option to search and select the relevant file. Perhaps this should be the same process as the one used to select the database where there is a choice of more than one in the folder.

The bug is fixed. Take into account that the application appends date and time making name of the backup file unique. Clicking on “magnifier” will result to empty list unless you erase tail of the name. See picture below.



### 6.4. The system doesn’t allow the user to restore a database to a live database and so the option available to us is to leave the application and to create a new empty or blank database via SQL Express. We can then return to the application and select the newly created DB to restore to. Would it be possible to add an option to the end of list of Databases names that says something like ‘Create New’? This would allow all the activity to be completed within the control of the application itself.

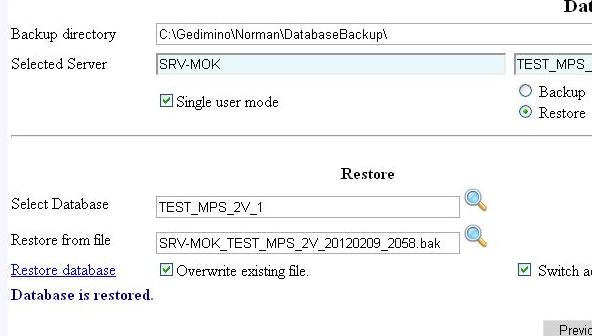
I enhanced the project with procedure for killing processes running on target database (database that you are going to overwrite from backup file). Write it into source database (database that you are using for connecting to server) launching file C:\Repo.git\CFCDatabase\SQLscripts\Create\_usp\_KillUsers.sql. The service will kill all processes connected to target database before switching to single user mode (restore database overwriting existing one is the only place where flag “Select single user mode” is mandatory).

Experimenting with SQL 2008 R2, IIS7.5, Windows WEB Server 2008 R2 I found what goes wrong on your environment and here is instruction how to configure the application:

1. All Microsoft.SqlServer.Management.Smo operation (backup, restore, enumerate tables, …) are running under Web application’s account. Working with IIS7x you need:
   1. Create separate application pool on IIS,
   2. Assign some name and password for the pool (Harry/Potter for example),
   3. Move WEB application into this pool,
2. Create user in your SQL server with the same credentials (Harry/Potter),
3. Login into database using SQL Server Management studio; use SQL authentication mode with Harry/Potter credentials and ensure that this user has sufficient rights:
   1. Try to kill processes launching “EXEC usp\_KillUsers targetDatabase”,
   2. Switch target database to single user’s mode
   3. Backup database,
   4. Restore it from the backup file,
   5. Switch target data back to multiuser mode.

usp\_KillUsers will complain that it is impossible to kill running process. This is normal position and the service ignores this error message.

After configuring account the application will work: I was playing with 3 different computers: service, remote client of the application and remote client of MS SQL server (table’s editor was opened there) and got no errors:



I was surprised: I could to continue data editing after database was restored, but I don’t recommend you “repeating that at home”. I got this lucky case because initial and restored databases were identical.