

BioNumerics Tutorial:

Setting up a database for multi-user access

1 Aims

There are several situations in which multiple users in the same local area network (LAN) may wish to work with a shared BioNumerics database. Examples are projects on which a number of researchers are collaborating, a centralized culture collection of a laboratory, etc..

The configuration of databases in BioNumerics is very flexible and allows the setup of a shared database. However, a database created in BioNumerics with all settings at their default values is basically configured for single-user database access. When BioNumerics is installed on more than one workstation and a database needs to be accessed *simultaneously* by more than one user, a specific setup is required. The current tutorial describes this database setup step-by-step.

2 Principle

As an overall principle for enabling multi-user database access, it is important to share only the relational database and not the BioNumerics file folders. Industry-standard relational databases such as SQL Server, Oracle and MySQL are designed to deal with multiple concurrent users, while access to a file is always single-user.

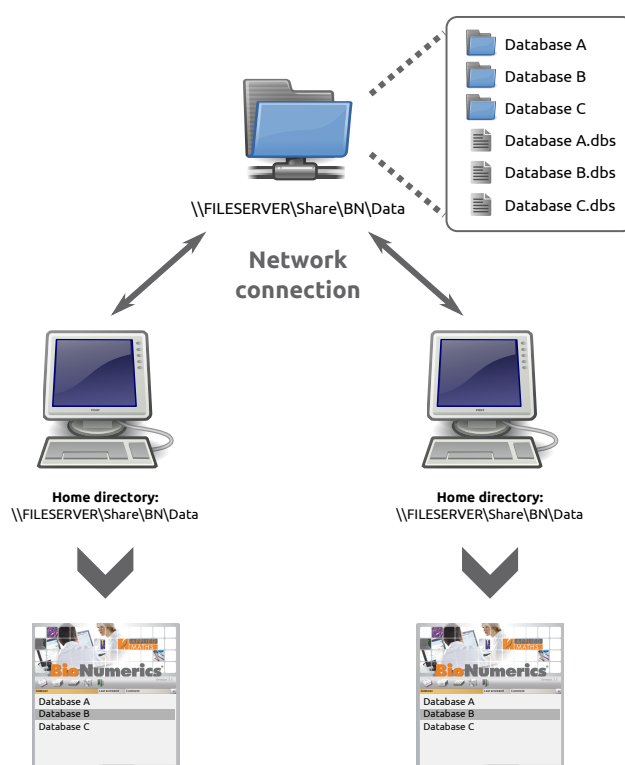


Figure 1: Sharing BioNumerics databases by specifying a common home directory on a network share: the same list of databases appears for both users.

Figure 1 illustrates an **inadvisable** scheme in which a directory on a network share is specified as BioNumerics home directory for two Windows users on different computers.

Although the databases can indeed be accessed by more than one user in this setup, the approach actually has several drawbacks:

- There is no *simultaneous* multi-user access: the first user that opens the database will lock the database descriptor file (*.dbs) and a subsequent user will not be able to open the database (a Windows error code 5 is generated) as long as it is opened by the first user.
- All databases appear in the startup screen of all users that have specified the same home directory.
- Preferences (such as window sizes and positions, last-used settings in dialog boxes, etc.) cannot be stored for individual users, but are instead the same for all users of a database.

In the setup illustrated in Figure 2, each user has his/her own home directory. This directory can reside on the user's own machine or on a file server (the latter being easier to include in backup routines, see 6). Selected databases are shared via an ODBC connection.

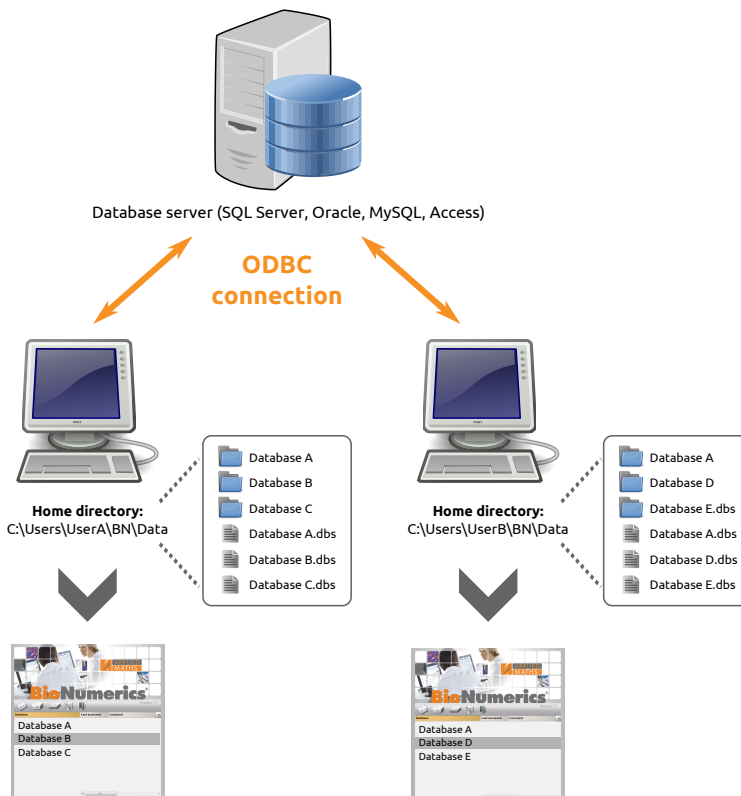


Figure 2: Sharing a BioNumerics database via its ODBC connection: Database A is shared by both users, while Database B and Database C are only shown in the *BioNumerics Startup* window of the user shown on the left. Similarly, Database D and Database E are personal databases of the other user.

As a result, this setup offers:

- Simultaneous multi-user access to a shared database.
- Flexibility in sharing databases: specific databases can be shared with other users, while other databases can be kept private.

- Since settings are stored per database in the home directory, they are now remembered for individual (Windows) users.

The steps in which to achieve a setup as shown in Figure 2 will be illustrated in this tutorial.

3 Prerequisites

Before proceeding with this tutorial, please check if following prerequisites are met:

- A **database server** needs to be set up and access rights granted for all intended database users. Recommended database management software (DBMS) to use in conjunction with BioNumerics is Microsoft SQL Server (Express), Oracle Database and MySQL Server. For relatively lightweight applications (i.e. less than 10 Gigabyte per database) and ease of installation, the SQL Server Express instance embedded with the BioNumerics setup can be used for this purpose.
- A **new, empty relational database** or schema (Oracle) should be created on the database server. Although BioNumerics can automatically create a relational database that acts as personal data storage (i.e. for a single user), databases that require simultaneous multi-user access need to be created in the DBMS. SQL Server Management Studio can be used for creating a new database on Microsoft SQL Server. The default schema for all BioNumerics database users must be set to "dbo". This assures that the correct container is used for the BioNumerics SQL objects such as tables, as BioNumerics does not support non-default or custom schemas, and in no way implies that the user has "db_owner" privileges on the database. BioNumerics users that will install updates and plugins will need to be a member of the "db_owner" database role. Regular users must be members of the "db_datareader" and "db_datawriter" database roles, and must have execute permissions on the usp_GETAUTONM SQL stored procedure. The following SQL query illustrates how you can grant the appropriate permissions to regular BioNumerics users:

```
EXEC sp_addrolemember @rolename = 'db_datareader', @membername = N'<User>';
EXEC sp_addrolemember @rolename = 'db_datawriter', @membername = N'<User>';
GRANT EXECUTE ON OBJECT::[dbo].[usp_GETAUTONM] TO [<User>];
```
- All client computers must have an **ODBC driver** version installed which fully supports the target DBMS. The SQL Server 2008 R2 Native Client ODBC driver is automatically installed with the BioNumerics application. This ODBC driver supports the included SQL Server 2008 R2 Express database engine, and other editions of SQL Server 2005 and 2008.
- **Source files directory:** create and share a directory on a file server to contain any data that is not stored in the relational database. Make sure that all intended BioNumerics users have Full Control permissions on the network share, and Modify NTFS permission on the directory.

4 Connecting the first user

First, verify the location of the user's home directory. Preferably, this should be a personal folder on a file server with a backup system in place.

1. Double-click on the BioNumerics icon () on the desktop to start BioNumerics.

2. In the *BioNumerics Startup* window, click on the  button and select **Change home directory** from the menu that pops up.

3. Check the location specified for the home directory and press <**Cancel**> to return to the *BioNumerics Startup* window.



Keep in mind that, if you change the home directory and still want to see the same databases listed, that the content of the current home directory needs to be transferred to the new home directory.

Next, we will create a new BioNumerics database.

4. In the *BioNumerics Startup* window, click on the  button to start the *New database wizard*.

5. Enter the name of the new database as you want to see it listed in the *BioNumerics Startup* window (e.g. “Shared database”) and press <**Next**>.

6. In the second step of the *New database wizard*, leave all settings to their default values and press <**Finish**>.

So far, the database directory has been created with a number of predefined files and sub-folders. While the database directory contains e.g. (user-specific) display settings for dialogs and windows, the actual data will be stored in the relational database (and in some cases also in the source files directory). The remaining part of the *New database wizard* deals with the setup of the relational database.

7. In the *New database* dialog box, check **Use existing** (since the database has already been created in the DBMS; see 3) and press <**Next**>.

8. In the *Locate database* dialog box, press the <**Build**> button to create a connection to the relational database.

The dialog box that pops up now is generated by your Windows operating system and may differ depending on the Windows version installed.

The instructions below (Instruction 9 to Instruction 18) are valid for creating a new file data source in Windows 7, to connect to a SQL Server 2008 R2 database using Integrated Windows Authentication. Note that the procedure will be different when another DBMS and/or another method of authentication is used.

9. In the *File data source tab* of the *Select data source dialog box*, click on <**New**> to create a new DSN file.

10. Select the ODBC driver that corresponds with the DBMS from the list (here: “SQL Server Native Client 10.0”) and press <**Next**>. Please note that the “SQL Server” ODBC driver is outdated, instead a “SQL Server Native Client” version that matches or exceeds the SQL Server version should be selected.

11. Enter a name for the DSN file, e.g. “BioNumerics shared database” and press <**Next**>.

12. Press <**Finish**>. The ODBC driver will prompt you for additional information (see Figure 3).

13. Optionally, provide a **Description** for the connection.

14. Enter the database server name or select it from the drop-down list and press <**Finish**>.

15. Optionally, check the connection by pressing <**Test data source**>.

16. Press <**OK**> twice. The *SQL Server Login dialog box* pops up (see Figure 6).

17. Press <**Options**> to reveal additional options in the *SQL Server Login dialog box*.

18. Select the **Database** from the corresponding drop-down list and press <**OK**>.

The connection string is now filled out in the *Locate database* dialog box (back in BioNumerics).

19. From the **SQL dialect** drop-down list, select the option that corresponds with the DBMS you are using (e.g. **SQL Server®**). Next, press <**Finish**>.

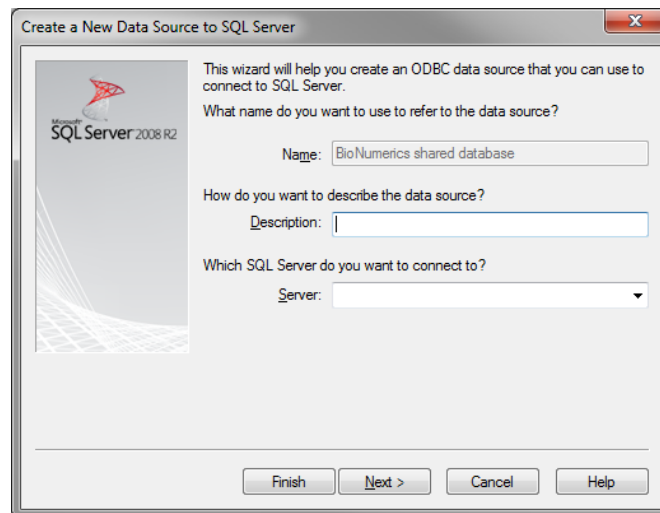


Figure 3: Additional information required by the ODBC driver.

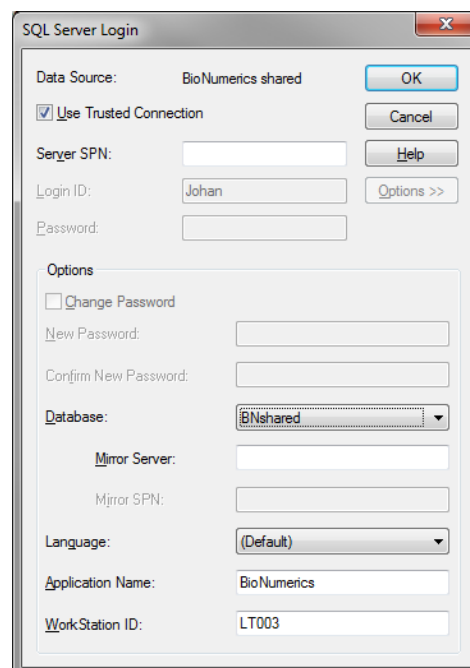


Figure 4: The *SQL Server Login* dialog box.

Since this is a new database, that does not contain the required table structure yet, the software asks to create the necessary tables.

20. Answer **<Yes>** to have the table structure automatically created. This process may take some time.



A log file called BUILDREPORT.TXT is created in the BioNumerics home directory during this process. This log file can be very useful to track any errors that occur.

21. Answer **<Yes>** to the question "Do you want to review database settings now?".

This action pops up the *Database settings* dialog box, in which we will specify the source files directory. The default location is fine for databases for personal use, but if the database has to be shared by several users, it is important to specify a location that is accessible by all users (e.g. a shared folder on a network

drive).

22. Press the **<Browse>** button to specify a *Source files location* on a shared network drive.
23. Press **<OK>** in the *Database settings* dialog box and in the warning message that appears.
24. In the *Plugins* dialog box that pops up, press **<Proceed>** to open the new database.

The user can now start working with the database.

5 Connecting subsequent users

The procedure to connect any subsequent user to the same database shares several steps with the procedure described in 4. However, because there are a few important differences, the complete set of instructions is given here below.

Verify the location of this user's home directory. The home directory should differ from the other users' home directory and should preferably be located in a personal folder on a file server with a backup system in place.

1. Double-click on the BioNumerics icon () on the desktop to start BioNumerics.
2. In the *BioNumerics Startup* window, click on the  button and select **Change home directory** from the menu that pops up.
3. Check the location specified for the home directory and press **<Cancel>** to return to the *BioNumerics Startup* window.



Keep in mind that, if you change the home directory and still want to see the same databases listed, that the content of the current home directory needs to be transferred to the new home directory.

Next, we will create a new BioNumerics database.

4. In the *BioNumerics Startup* window, click on the  button to start the *New database wizard*.
5. Enter the name of the new database as you want to see it listed in the *BioNumerics Startup* window. To avoid confusion, it is recommended to use the same database name as for other users. Press **<Next>** to proceed to the second step of the *New database wizard*.
6. In this step, leave all settings to their default values and press **<Finish>**.

So far, the database directory has been created with a number of predefined files and sub-folders. While the database directory contains e.g. (user-specific) display settings for dialogs and windows, the actual data will be stored in the relational database (and in some cases also in the source files directory). The remaining part of the *New database wizard* deals with the setup of the relational database.

7. In the *New database* dialog box, check **Use existing** and press **<Next>**.
8. In the *Locate database* dialog box, press the **<Build>** button to create a connection to the relational database.

The dialog box that pops up now is generated by your Windows operating system and may differ depending on the Windows version installed.

The instructions below (Instruction 9 to Instruction 18) are valid for creating a new file data source in Windows 7, to connect to a SQL Server 2008 R2 database using Integrated Windows Authentication. Note that the procedure will be different when another DBMS and/or another method of authentication is used.

9. In the *File data source tab* of the *Select data source dialog box*, click on **<New>** to create a new DSN file.
10. Select the ODBC driver that corresponds with the DBMS from the list (here: "SQL Server Native Client 10.0") and press **<Next>**.
11. Enter a name for the DSN file, e.g. "BioNumerics shared database" and press **<Next>**.
12. Press **<Finish>**. The ODBC driver will prompt you for additional information (see Figure 5).

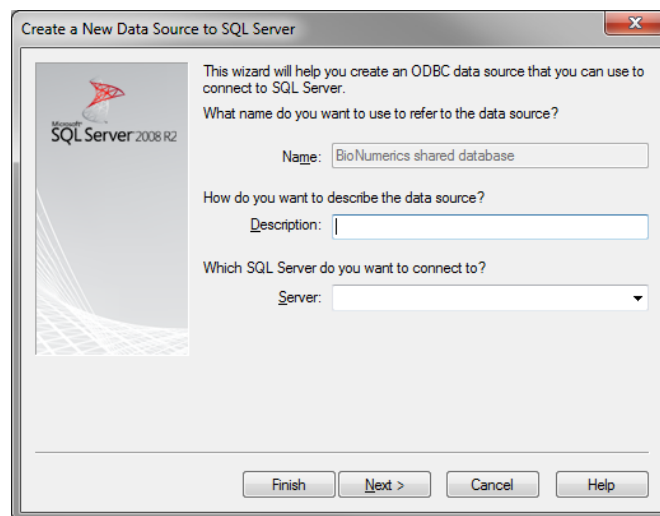


Figure 5: Additional information required by the ODBC driver.

13. Optionally, provide a **Description** for the connection.
14. Enter the database server name or select it from the drop-down list and press **<Finish>**.
15. Optionally, check the connection by pressing **<Test data source>**.
16. Press **<OK>** twice. The *SQL Server Login dialog box* pops up (see Figure 6).
17. Press **<Options>** to reveal additional options in the *SQL Server Login dialog box*.
18. Select the **Database** from the corresponding drop-down list and press **<OK>**.

The connection string is now filled out in the *Locate database dialog box* (back in BioNumerics).

19. From the **SQL dialect** drop-down list, select the option that corresponds with the DBMS you are using (e.g. **SQL Server®**). Next, press **<Finish>**.

BioNumerics automatically detects that the required table structure is present.

20. Answer **<Yes>** to the question "Do you want to review database settings now?".

This action pops up the *Database settings dialog box*, in which we will specify the source files directory. It is important to use the same directory on a shared network drive as was specified for the other users.

21. Press the **<Browse>** button and specify the **Source files location**.

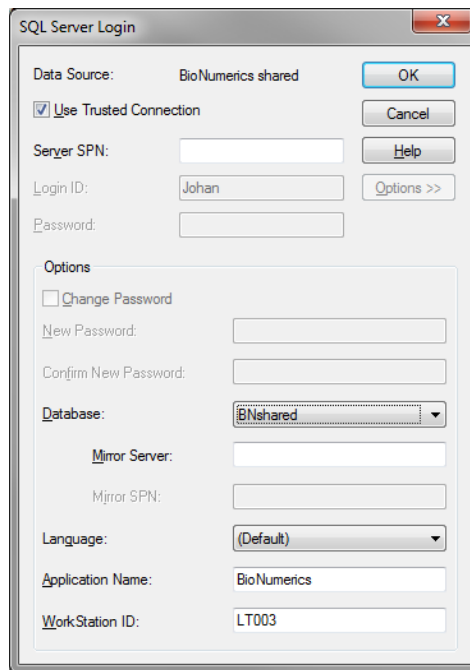


Figure 6: The *SQL Server Login* dialog box.

22. Press **<OK>** in the *Database settings* dialog box and in the warning message that appears.

23. In the *Plugins* dialog box that pops up, press **<Proceed>** to open the new database.

6 Taking backups

To prevent against accidental data loss, it is important to take backups on regular intervals. For a BioNumerics database that is shared according to the setup described here, following data should be included in a backup routine:

- The shared relational database: Each DBMS has its own backup procedures, which can often be scheduled to run at specified intervals (e.g. daily or weekly). We refer to the DBMS documentation for the setup of such automatic backups.
- The source files directory: This should be included in the regular file backup routine.
- The BioNumerics home directory: alongside the user-specific display settings for dialogs and windows, this folder also contains files defining where the BioNumerics data is stored and may contain a number of personal databases. For example, the *.dbs files contain the absolute or relative path to the BioNumerics database folder and the *.xdb files contain the ODBC connection strings. Hence it is recommended to include all personal home directories in the regular backup routine of the file server.