hyperMILL®

© The helmet was programmed and produced by BAUSHIN

OPEN MIND tool database in an SQL server environment

OPEN MIND
THE CAM FORCE

Last updated: 21. April 2022

OPEN MIND Technologies AG

Argelsrieder Feld 5

82234 Wessling

Germany

Tel.: (+49-8153) 933-500

Fax: (+49-8153) 933-501

E-mail: <sales.europe@openmind-tech.com>

Web: www.openmind-tech.com

Table of Contents

Tool database in multi-user mode

NOTE

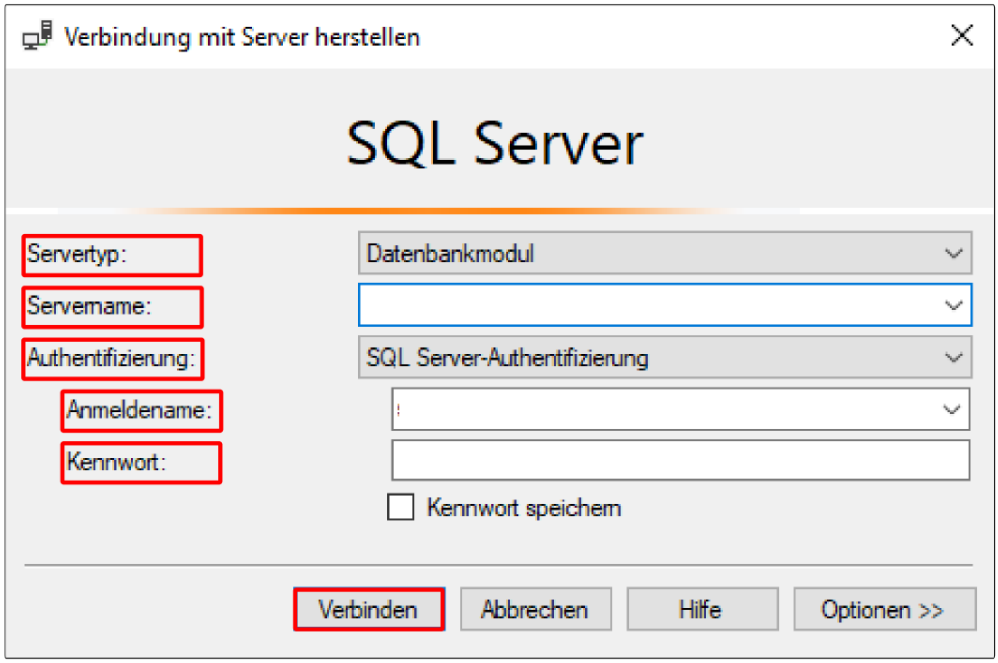
SQLite (`*.db`) format databases **cannot** be used used in multi-user mode. The multi-user mode is only supported by databases opened via `*.dsn` files.

This database format and therefore also the multiple user mode are supported since version 2014.1.

Creating an OPEN MIND tool database on a server

1. Install the **Microsoft SQL Server Management Studio** software on the computer that you use to manage your tool databases. Start the software as **Administrator**. Select **Connect Object Explorer** under **File**.

2. In the **Connect to Server** dialogue, select **Database Engine** under **Server type**, the name of your SQL Server under **Server name** and **SQL Server Authentication** under Authentication. Then enter your login details (**Login** and **Password**) and click **Connect**.



3. Create a new database. To do this, click the **Databases** entry in the file browser and choose **New Database** on the shortcut menu. Enter the required database name under **Database name**. Click **OK** to close the dialogue. The database appears as an entry in the file browser.

4. Open the `sqlserver.sql` script file for your new database's data definition by double-clicking (the script is automatically loaded into Microsoft SQL Server Management Studio) or via the menu item File → Open → File.
   By default, the script file is located in the following directory:

   ```
   C:\Program Files\OPEN MIND\Tool Database\[version]\template database.
   ```

   > **NOTE**
   > Alternatively, you can also open and copy the content of the script file in a text editor. Select Script for database as → Create → New query editor window from the database's shortcut menu. Delete the content in the editor window and paste the content of the database script from the clipboard.

5. Enter the name of the previously created database in the **USE** instruction at the beginning of the script (example: `tool_db_2018.2.db`).

```
USE [tool_db_2018.db]
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
```

6. Execute the script. To do this, click **Execute** on the menu bar.
You have now completed the administrator tasks.

NOTE
Please note that it may be necessary to adapt the permissions for the new database (login details) to the database administrator.

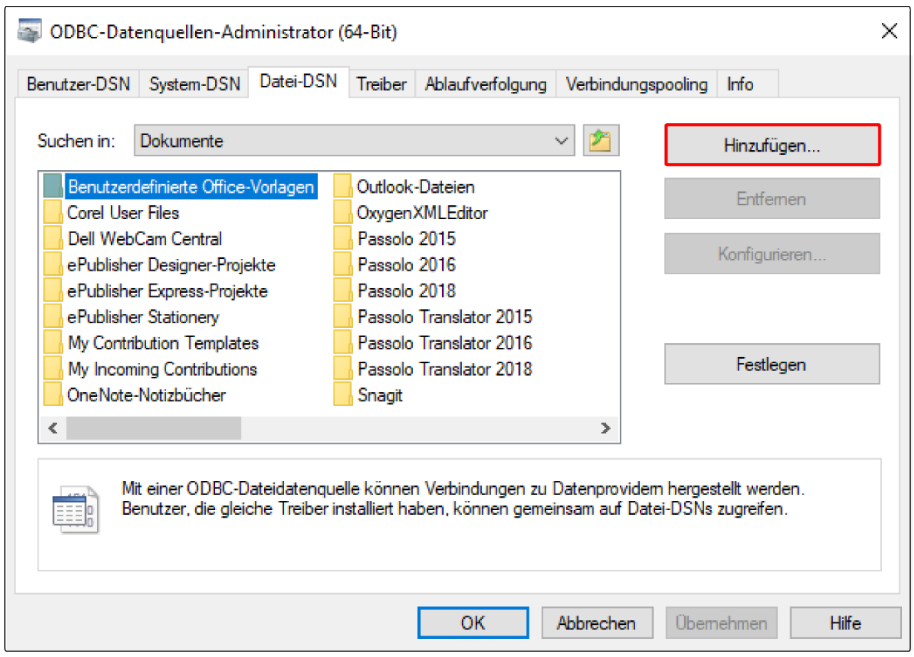
Make new database available on client computers

To ensure that the OPEN MIND tool database can establish a connection to the SQL Server, you must create a DSN file with the connection information for the driver that the SQL Server client needs.

> **NOTE**
> If the permissions were adapted for the database administrator when you created the tool database on the server, the following tasks must be performed on the database administrator's system.

1. Click the magnifying glass icon on the Windows taskbar, enter ODBC and select **Set up ODBC data sources (64-Bit)**.

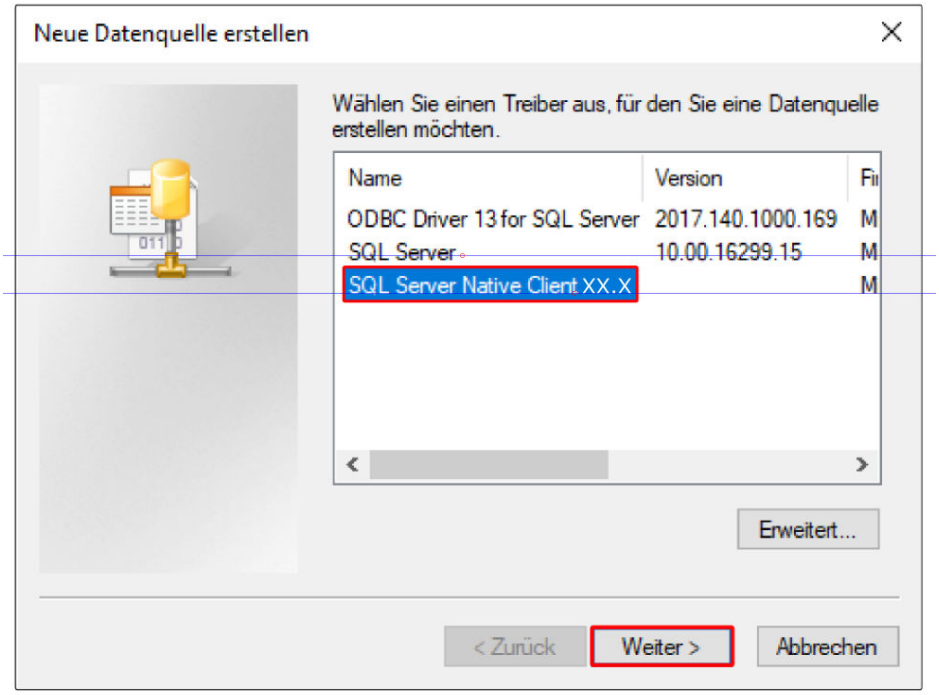
2. Start the software, switch to the **File DSN** tab and click **Add**.



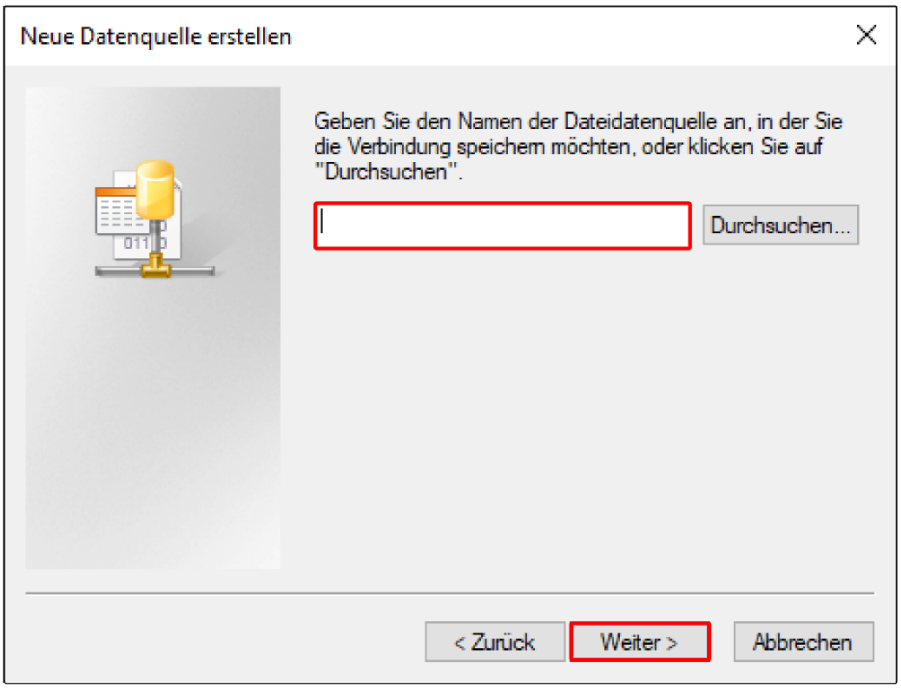
3. In the next dialogue, always select the highest available version of the SQL Server Native Client from the list of available drivers and click **Next**.

> **WARNING**
> **Microsoft ODBC Driver for SQL**
>
> Ensure that you are also using an up-to-date driver. The name of the driver should read: `SQL Server Native Client XX.X`.
>
> Do not use drivers with the name **SQL Server**. This driver cannot transfer more than 400 kB of data to the server. As a result, you cannot use it to store 3D tools in your database.

Further information on the current driver is available under the following link:
ODBC Driver for SQL Server

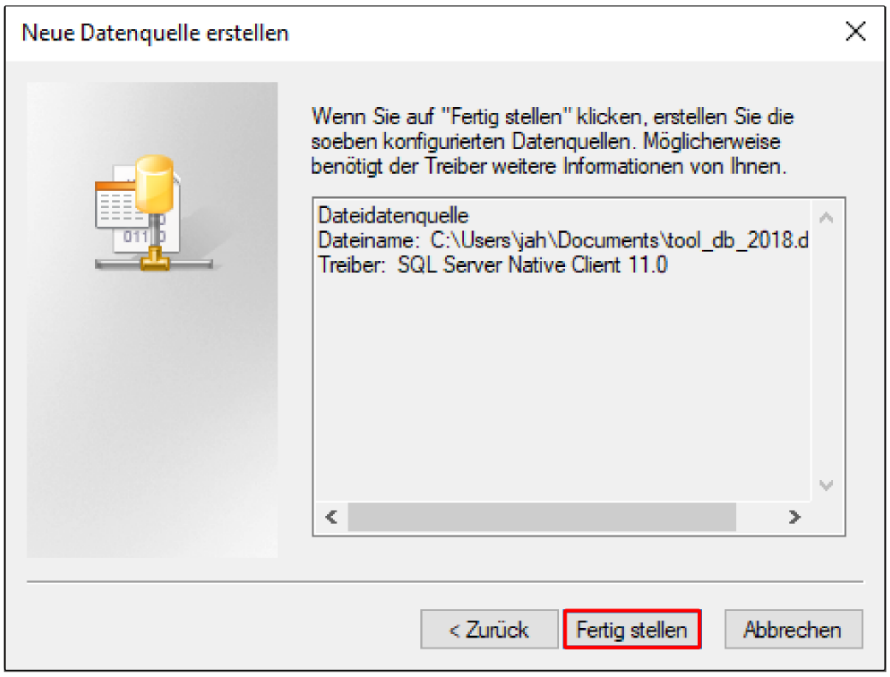


4. Specify a name and the storage location for the `*.dsn` file and click **Next**.

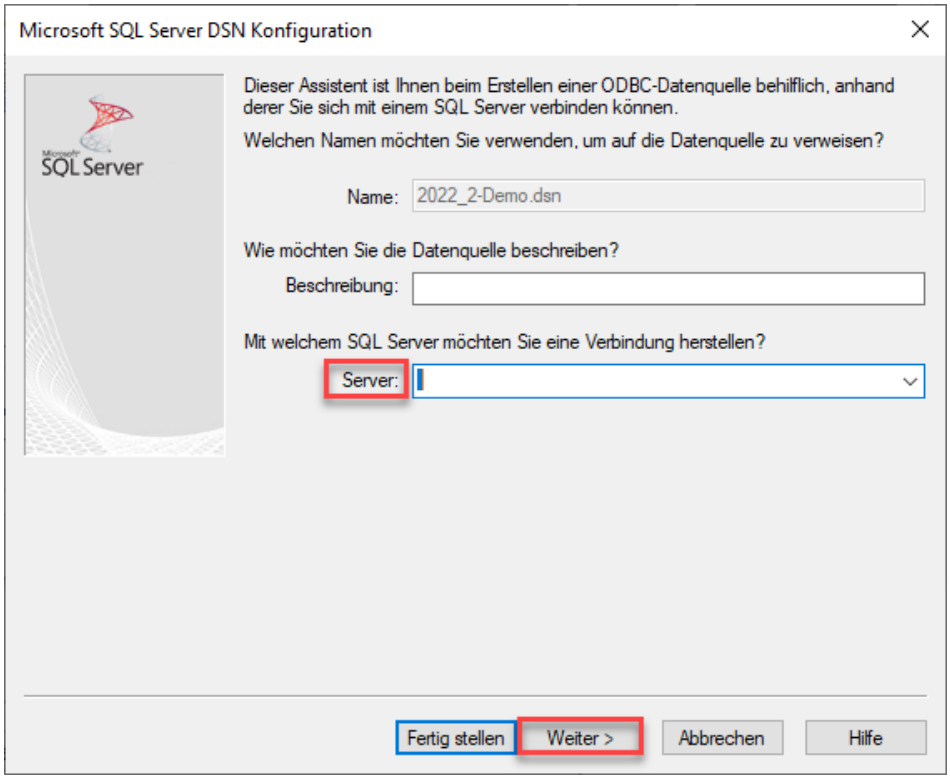


5. Then click **Finish**.
   By default, the `*.dsn` file is saved in the following directory:

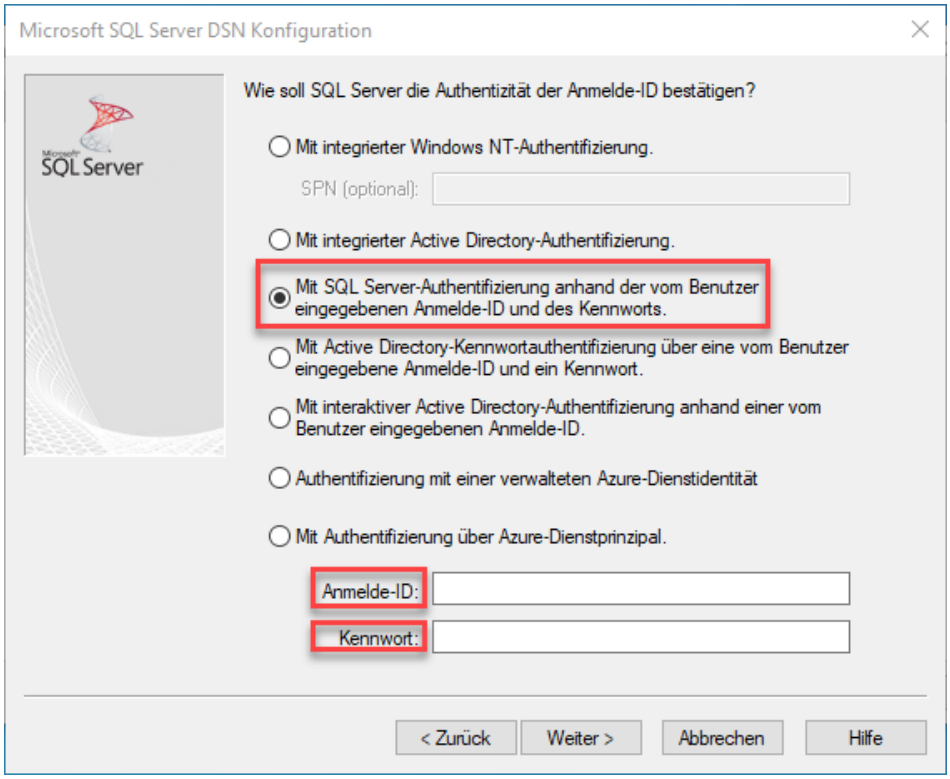
```
C:\Users\[Benutzername]\Documents\
```



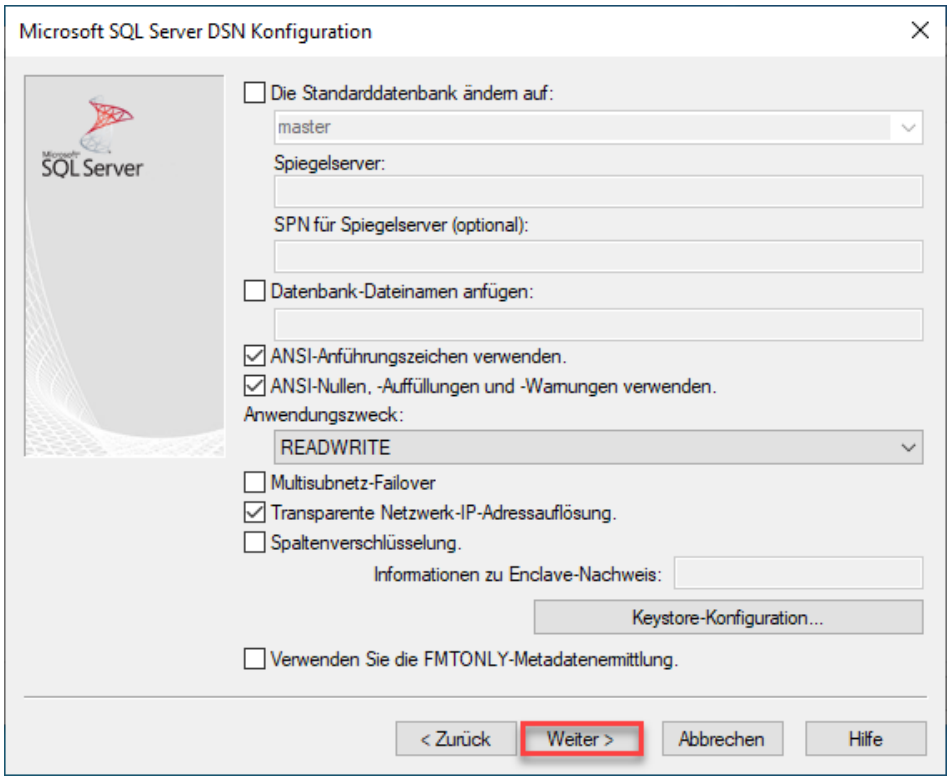
6. The **Microsoft SQL Server Configuration** dialogue opens and the name of the previously created `*.dsn` file is displayed. If you choose, you can provide a description of your data source under **Description**. Select the name of the SQL Server under **Server** and click **Next**.



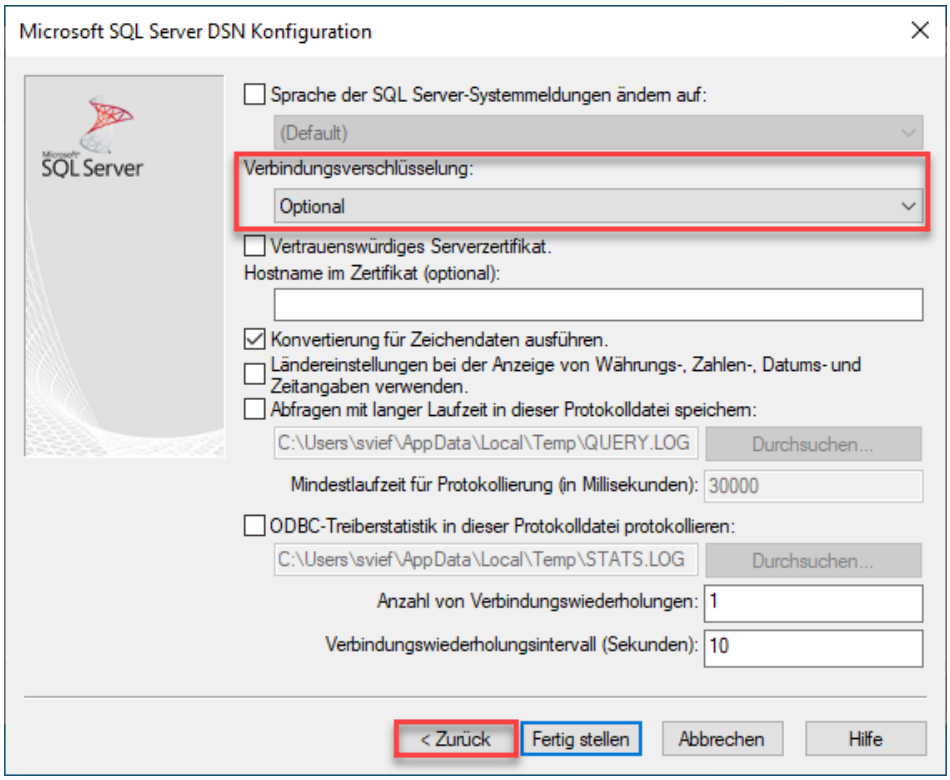
7. Then select **With SQL Server authentication using a login ID and password entered by the user**, enter the login details and click **Next.**



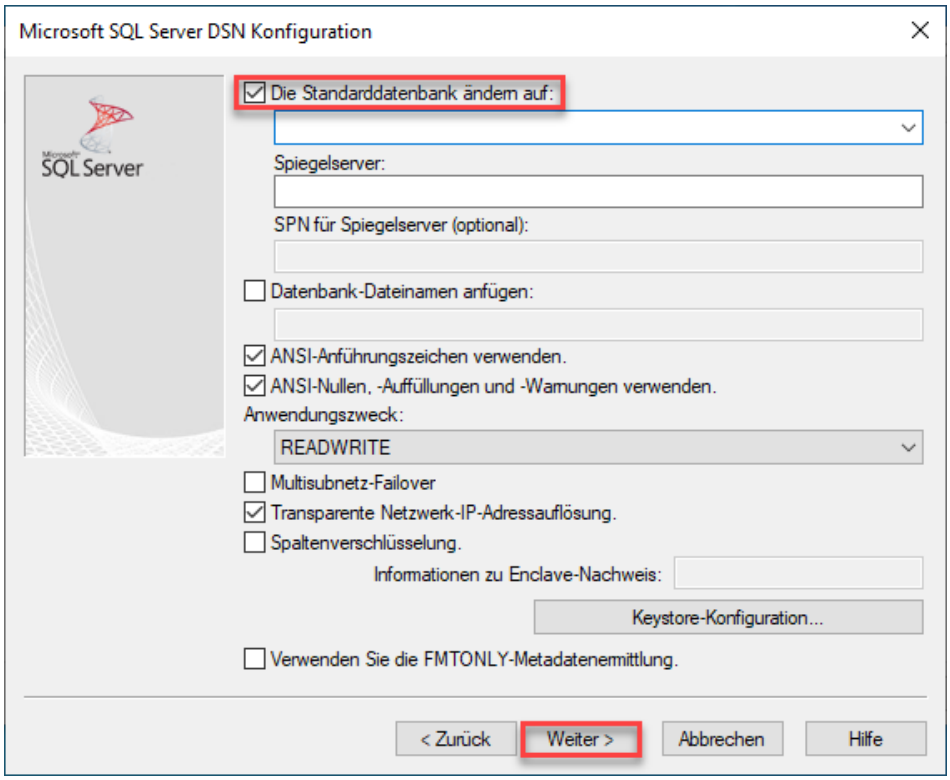
8. Click **Next** in the following dialog.



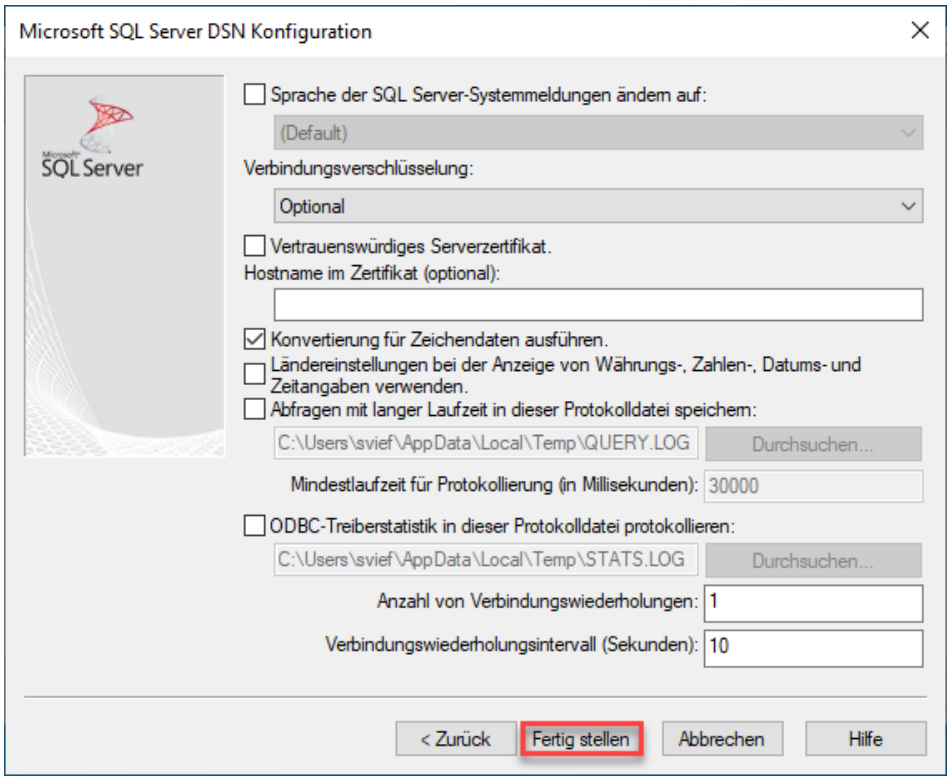
9. Then, set **Optional** for **Connection Encryption** and click **Back**.



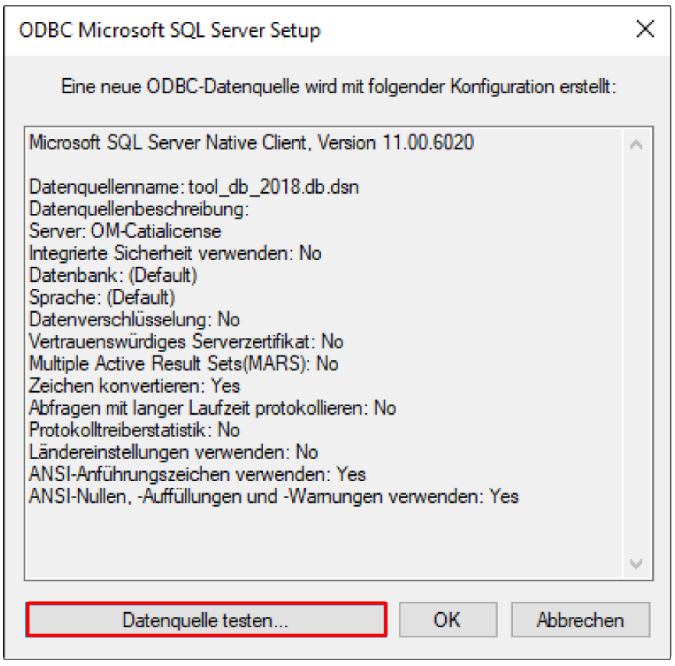
10. Activate the **Change the default database to**: option and enter the name of the database previously created in step A. Click **Next**.



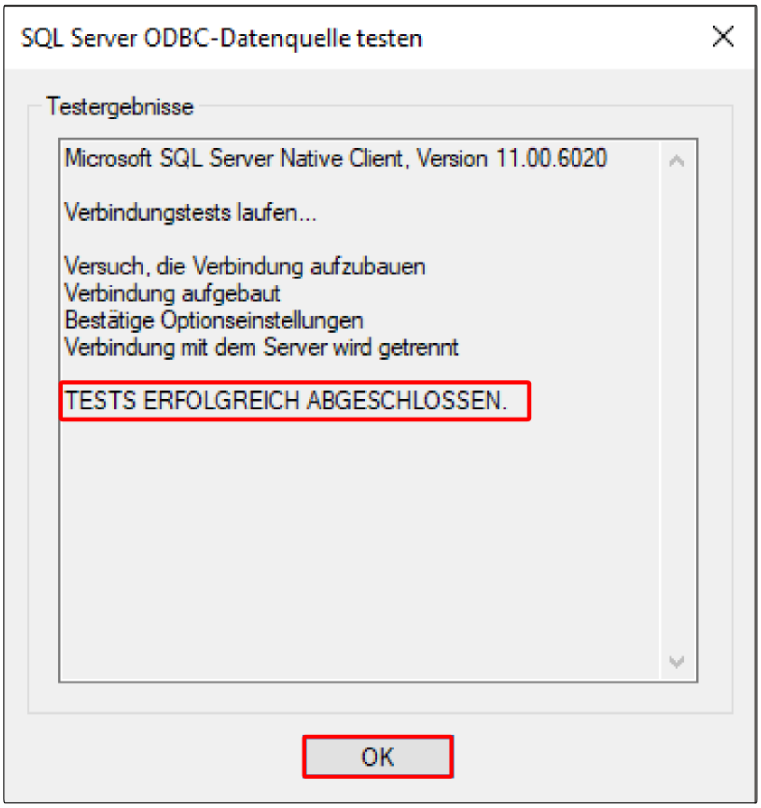
11. Click **Finish**.



12. A summary of all the information is displayed on the last dialogue page. Click **Test Data Source** to test the connection between the tool database and SQL Server.



13. If the **TESTS COMPLETED SUCCESSFULLY!** message appears, the connection is functioning correctly. Click **OK** to close the dialogue.



The `*.dsn` file should now contain the following information:

```
DRIVER=SQL Server Native Client 11.0
UID=[your login name]
PWD=[your password]
DATABASE=[your database name]
WSID=[your client computer name]
APP=Microsoft® Windows® Operating System
SERVER=[your server computer name]
Description=[optional – your database description]
```

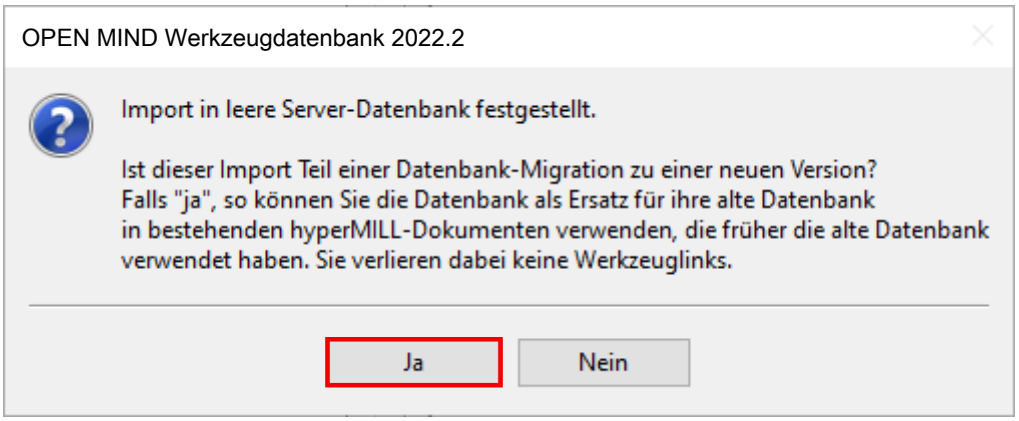
NOTE

If the password entry is not present, add the password (PWD) manually to the file. To do this, select the file and then select **Open with**. Choose a standard text editor for editing and save the `*.dsn` file.

Importing tools into the newly created SQL database

1. Start your tool database on the client computer, select File → Open and select `ODBC File Data Source (*.dsn)` as the format in the **Open Tool Database** dialogue. Under **File name**, select the name of the previously created data source file (`*.dsn`).
   You have now opened the new database and can import your tools from an older version of the database into it.

2. To import your tools, click the name of the newly created database in the tool database browser and select **Import...** from the shortcut menu. Set `ODBC File Data Source` as the file type and select the `*.dsn` file of your previous database version. Answer the question **Is this import part of a database migration to a new version?** in the following dialogue with **Yes**.



3. The progress of the tool import is shown in the **Import State** dialogue. The **Messages** section indicates how many tools were imported and whether the import was completed successfully.

> **NOTE**
> If your tool data is available in SQLite format (`*.db`) or as a neutral exchange format (`*.xml`), you can also import it into a newly created SQL Server database.

4. Click **Close** to close the dialogue.
   You have now successfully imported your tools from your old database into the new version of the database.

Update an existing SQL database

1. Start your tool database on the client computer, select File → Open and select `ODBC File Data Source (*.dsn)` as the format in the **Open Tool Database** dialogue. Under **File name**, select the name of the database to updated (for example: `Update database.dsn`).

2. To update your tools in the file browser, click the name of the database you want to update (for example: **Update Database.dsn**) and select **Import...** in the context menu. Set `ODBC File Data Source` as the file type and select the `*.dsn` file of your previous database version (for example: `database.dsn`).

3. The following message appears: Detected import into empty server database.Is this import part of a database migration to a newer version? Answering "yes" allows you to use the database as replacement for the old database in existing hyperMILL documents that formerly used the old database without losing tool links.
   Confirm the message with **Yes**,

4. The progress of the tool import is shown in the **Import State** dialogue. The **Messages** section indicates how many tools were imported and whether the import was completed successfully.

5. Click **Close** to close the dialogue.
   You have successfully updated your existing database.