![CUSTOM TOOLS logo]

CustomTools 2015

Automate your daily routines

Designed by Jomet Oy

Changing CustomTools Server

Table of Contents

Getting Started

System Requirements

Supported Server Operating Systems

- Windows Server 2008
- Windows Server 2008 R2
- Windows Server 2012
- Windows Server 2012 R2

Software Requirements

To make the CustomTools server change properly and to avoid un-necessary issues, please ensure that you have the following tools installed.

Current Server:

- **Microsoft SQL Server Management Studio.** A free tool for Microsoft SQL Server database related maintenance tasks.
- **CustomTools Administration Tool.** CustomTools application for basic maintenance of the CustomTools database.

New Server:

- **Microsoft SQL Server.** Microsoft SQL Server is used as a database engine for the CustomTools database.
    - **Microsoft SQL Server 2014 Express** (64-bit) is included in **CustomTools 2015** installation package.
- **Microsoft SQL Server Management Studio.** A free tool for Microsoft SQL Server database related maintenance tasks.
- **CustomTools Administration Tool.** CustomTools application for basic maintenance of the CustomTools database.

Supported SQL Server Versions

- Microsoft SQL Server 2005
- Microsoft SQL Server 2008
- Microsoft SQL Server 2008 R2
- Microsoft SQL Server 2012
- Microsoft SQL Server 2014

Other versions of the **Microsoft SQL Server** may also work with CustomTools but, they may not be tested with CustomTools. For that reason it is recommended to use one of the SQL Server versions listed above.

Server Change Process

Before Taking New Server to Use

Before you begin further installation of your new server computer, it is recommended that you back up at least the following things from the server that is currently being used as a CustomTools server:

- CustomTools Database
- CustomTools Database Profiles

Back up the CustomTools Database

Use the **SQL Server Management Studio** to back up the CustomTools database. To make a backup from the database, you need to have the administrative permissions in the server computer and also in the SQL Server. Use the **System Administrator** *(sa)* account in SQL Server to ensure that you have the necessary permissions. On the Windows side, it is recommended that you use the local **Administrator** account to back up the SQL Server database.

To start backing up the desired database, right-click the database in the SQL Server databases tree view and select **Tasks, Back Up**.

The **Back Up Database** dialog box appear.
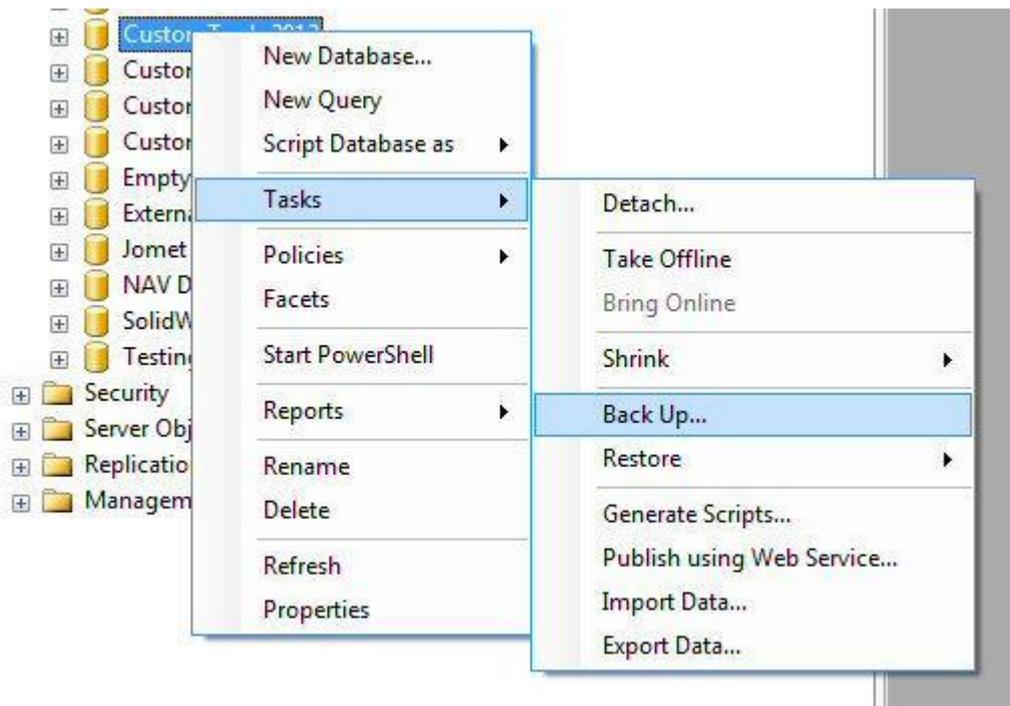
Figure 1. Select Tasks, Back Up to launch the SQL Server database backup wizard.
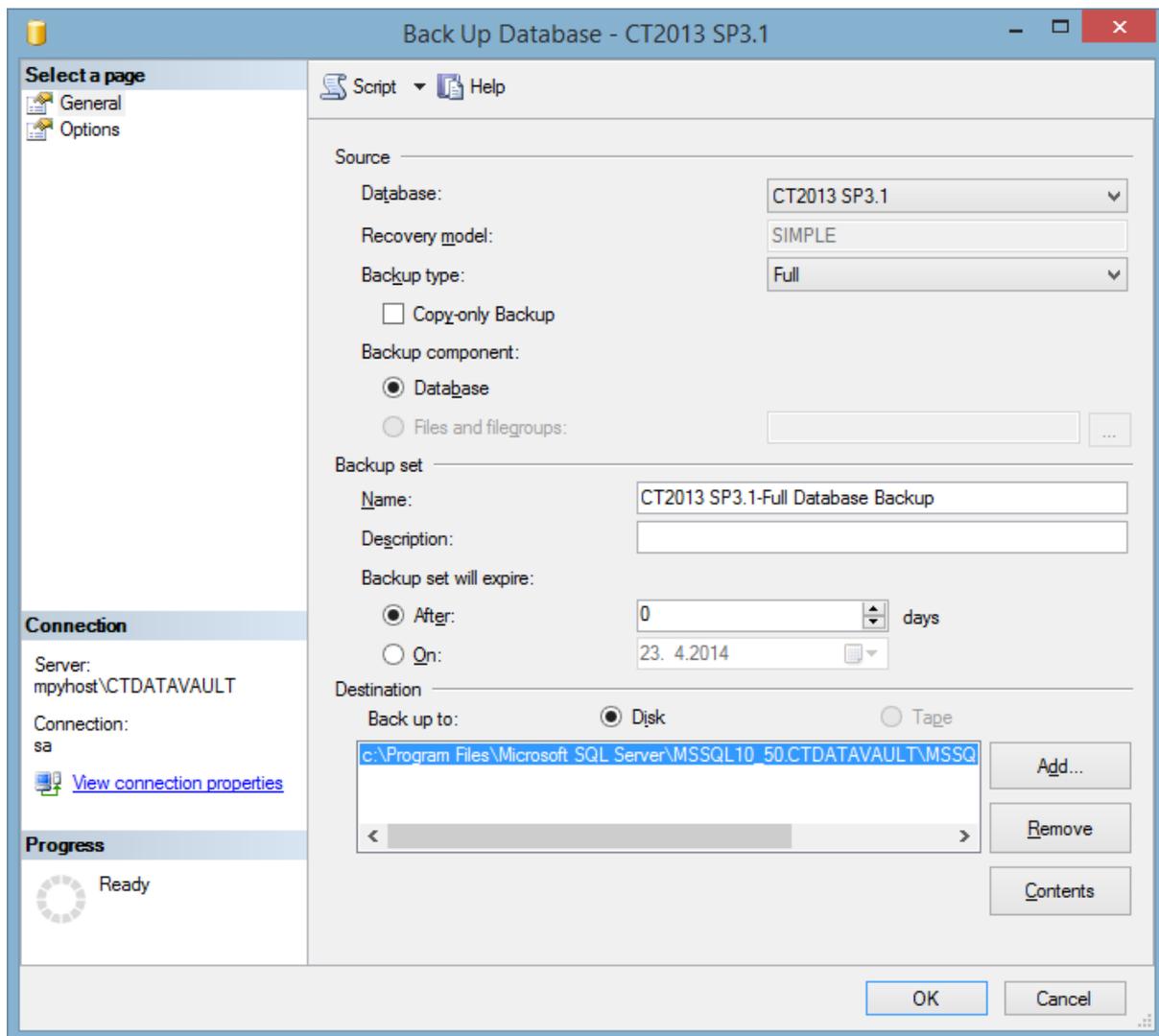
Figure 2. The Back Up Database dialog box is used to back up the selected database.

The **Back Up Database** dialog box shown above is used to setup the necessary settings for the database backup process:

- **Database**. Determines the database to back up.
- **Backup type**. Determines what kind of type backup to make from the selected database. Use the *Full* type of database backup in case you are changing the CustomTools database server.
- **Backup set:**
  - ○ **Name.** Determines the name for the backup set.
  - ○ **Description.** Use the **Description** box to add description for the backup set. (Optional)

- o **Backup set will expire.** Determines the expiration rule for the database backup set. Zero (0) in the **After** box means that the backup set will never expire. Database backup in this case, does not need to expire at all so type *0* into the **After** box.
- **Destination.** Determines the target path where the database backup file (.bak) is saved into. Click **Add** to set the target path.

Once you have set up the backup settings, click **OK** to back up the database.

💡 TIP: You can download the **Microsoft SQL Server 2014 Management Studio Express** for free from the **Microsoft download center** using the following link:

https://www.microsoft.com/en-us/download/details.aspx?id=42299

Back up CustomTools Database Profiles

Backups from the CustomTools database profiles could be taken using the CustomTools Administration Tool. To start CustomTools Administration Tool, click **Start, All Programs, CustomTools 20xx, CustomTools Administration Tool**.

Once you have the Administration Tool running, login to your CustomTools database. Then right-click the desired CustomTools database profile under the **Profiles** node in the **Database View** and select **Export Profile.**

The **Export Profile** dialog box appear.

Installing New Server

The new server should be installed using the necessary server side components available in the CustomTools installation package. Refer to the following examples to determine the server side components needed in your server:

Example 1 - Database Server

- **CustomTools Database**. SQL Server for CustomTools database. A 64-bit SQL Server 2014 Express gets installed by this component.
    - o If you prefer to install the SQL Server separately, you can download the **Microsoft SQL Server 2014 Express** installation package then for free from the **Microsoft Download Center** using the following link:
      https://www.microsoft.com/en-us/download/details.aspx?id=42299

NOTE: If you install 64-bit version of the SQL Server and, the server is used only as a database server in your CustomTools environment, you do not need to install anything from the CustomTools installation package.

Example 2 - Database and Network License Server

- **CustomTools Database**. SQL Server for CustomTools database. A 64-bit SQL Server 2014 Express gets installed by this component.
    - o If you prefer to install the SQL Server separately, you can download the **Microsoft SQL Server 2014 Express** installation package then for free from the **Microsoft Download Center** using the following link:
      https://www.microsoft.com/en-us/download/details.aspx?id=42299
- **CustomTools License Manager**. A Windows service and user interface for CustomTools network license management and monitoring.

NOTE: If you install 64-bit version of the SQL Server and, the server is used as a database server and also as a network license server, you need to install just the **CustomTools License Manager** component from the CustomTools installation package.

NOTE: Since the CustomTools network license have to be activated on the server side, the **Microsoft .NET Framework v4.5.2** is also required in CustomTools network license server. In a computer running the **Windows Server** operating system, you can install the .NET Framework using the Windows Server **Roles and Features Manager** located in the server's management console.

Restore CustomTools Database to a New Server

Once you have the **SQL Server** and the **SQL Server Management Studio** installed in your new server, copy the backup file (**.bak** file) of your CustomTools database, into the new server. Because of the SQL Server user permission policies, it is recommended that you copy the backup file into the default location where the SQL Server looks the backup file from. Typically, the default location is similar like the following one:

C:\Program Files\Microsoft SQL Server\MSSQL10_50.*SQLServerName*\MSSQL\Backup\

Once the database backup file is copied, run the **SQL Server Management Studio** as an Administrator and then login to the SQL Server using the SQL Server's System Administrator (sa) account.

Once logged in, right-click the **Databases** node and select **Restore Database**.
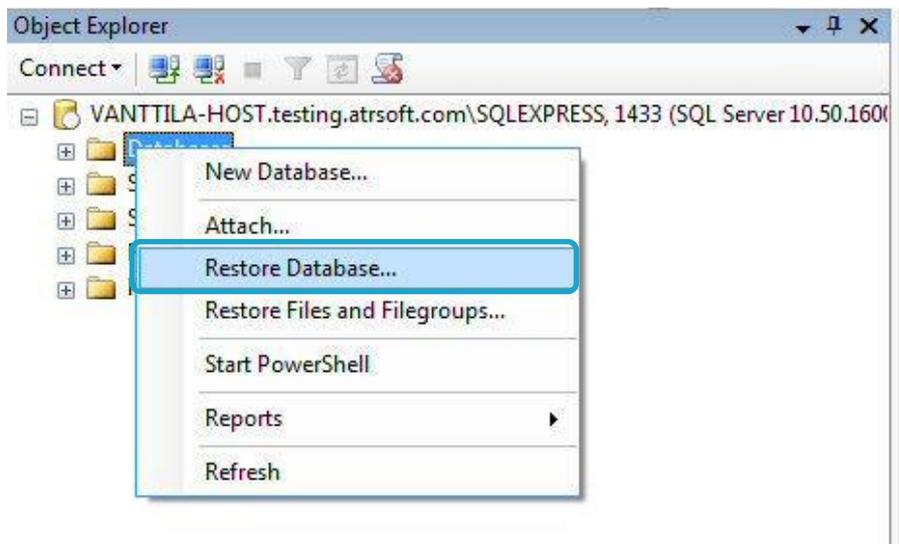

Figure 4. Right-click the Databases node and select Restore Database
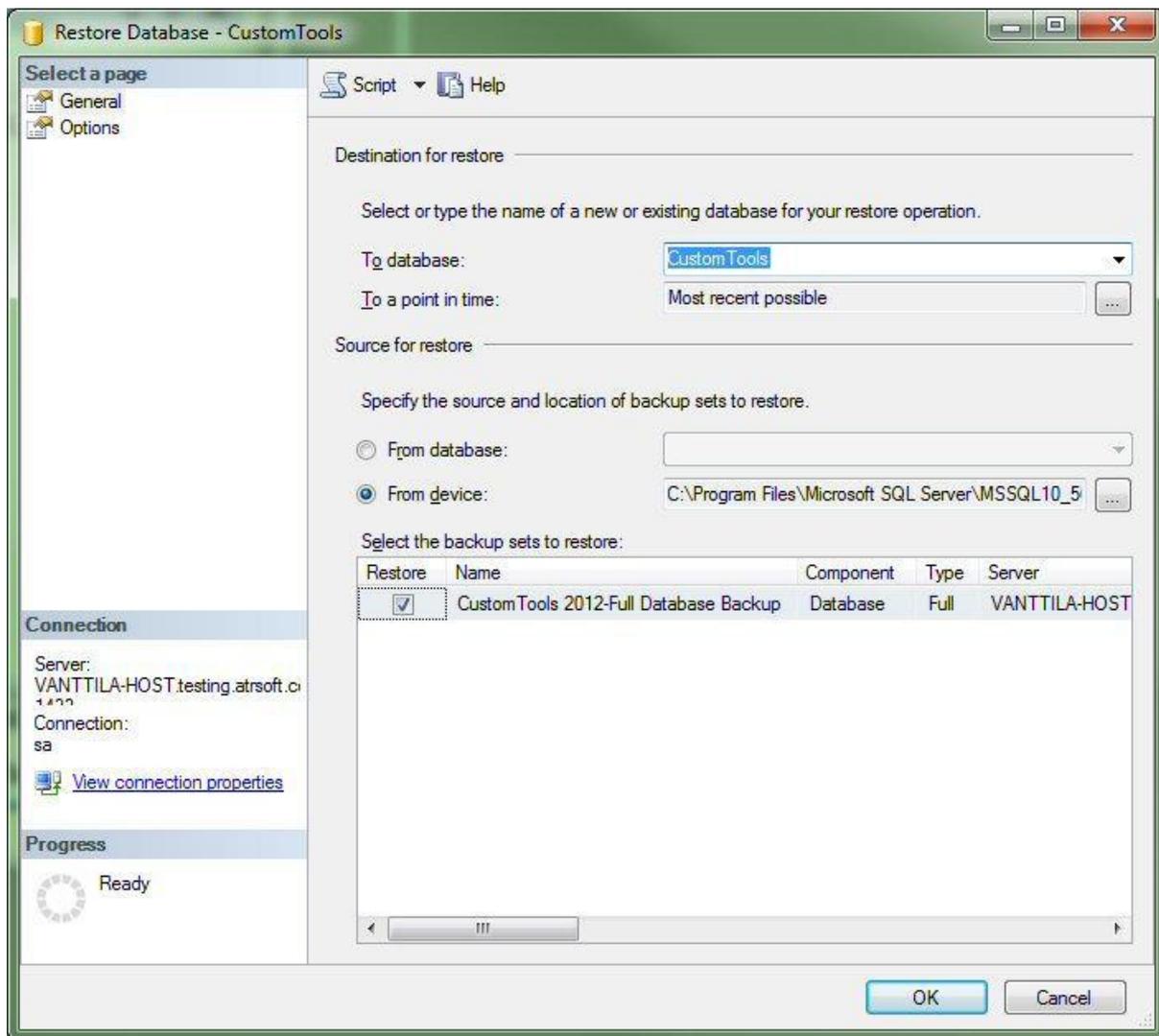
Figure 5. The Restore Database dialog box used to restore a database from a backup set

Setup the database restoring process using the **Restore Database** dialog box shown above:

- **To database.** Determines name of the database to restore the backup set to. The database name you type in here should not already exist in the SQL Server.

- **To a point in time.** Determines the database revision to be restored. Recommended in this case is to use the *Most recent possible* option as it is by default.

- **Source for restore:**
    - o **From database.** Restore database from another database.
    - o **From device.** Restore database from a backup file. In this case, you have created a backup file from your old CustomTools database so for that reason, use the **From device** option.
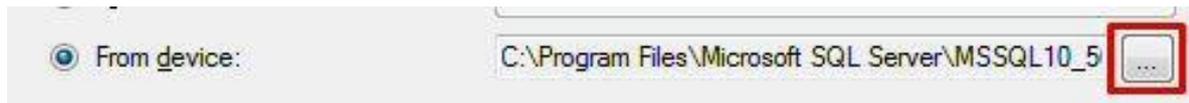
Click the button highlighted in the picture above to open a file browser that you can use to open your CustomTools database backup file created earlier. Once you have opened the backup file, select the **Restore** option associated to your CustomTools database backup set which appear to the **Backup sets to restore** list.

Click **OK** to restore the database backup into your new server's SQL Server.

Take the New Server into Use

You have now successfully installed an SQL Server into your new server and the CustomTools database have been restored to there from the backup set. There are now just few steps left before you can begin to use the new server.

Open UDP Port for SQL Server Browser

This is an optional step which benefit for you is to make easier to establish connection to your new server from CustomTools client workstations, an **UDP** port used by **SQL Server Browser** service should be opened in your server's firewall.
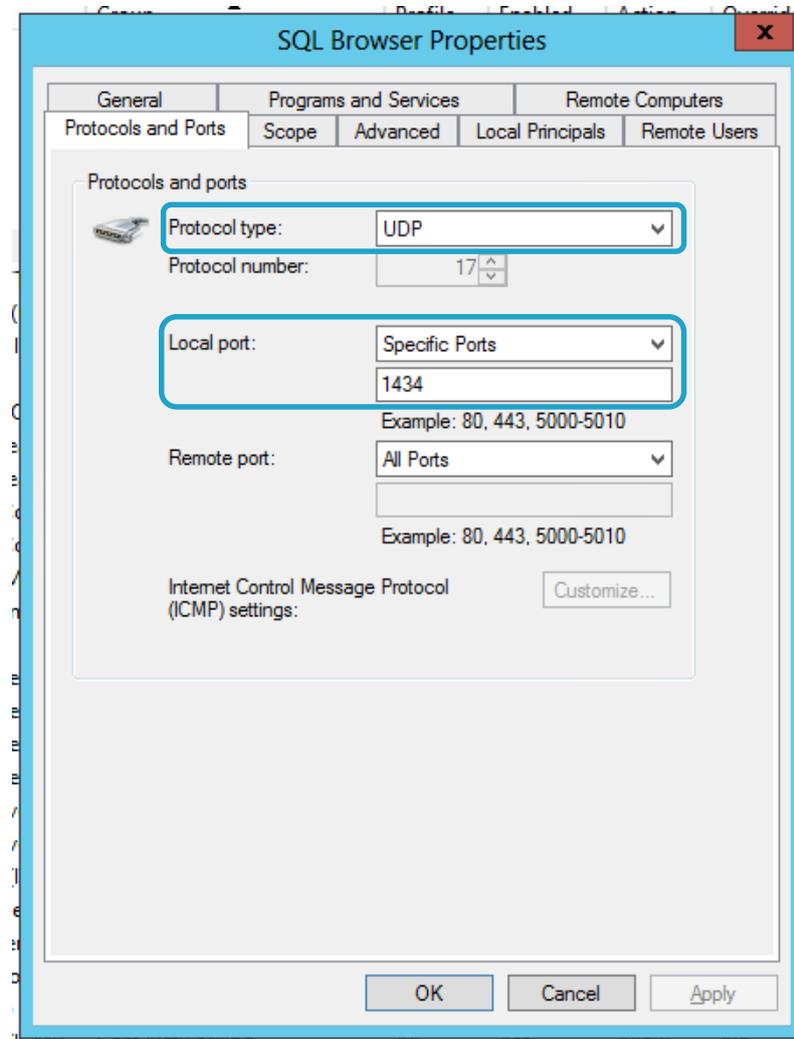
Figure 7. Properties example of an UDP port to be opened for SQL Server Browser service (Windows Firewall)

The **Figure 7** above shows the properties of a Windows firewall **Inbound rule**. You can create an **Inbound rule** by using the guided wizard available in **Windows Firewall with Advanced Security** in your **Windows Server**.

Change Database Server in CustomTools

Run the **CustomTools Administration Tool** program to change the database server used by CustomTools.
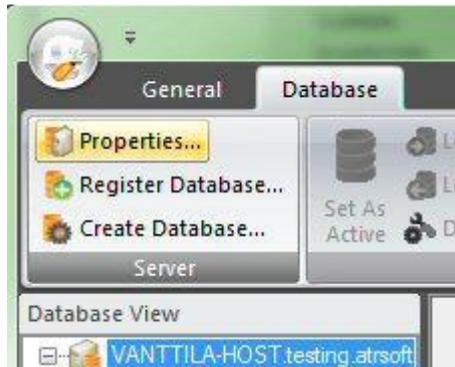
Once you have started the Administration Tool, select the database server node in the **Database View** and click the **Properties** command located in **Server** group of the ribbon. On click of the Properties button you get noticed about effects caused by changing the database server. Click **OK** to close the message box.

The **Register CustomTools Database Server** dialog box appear.

Figure 9. The Register CustomTools Database Server dialog box is used to setup the database server for CustomTools

Refer to the following instructions to setup the new database server for CustomTools:

- **Server name.** Type here the server name of the new SQL Server in format shown using the following format; <HOSTNAME or IP ADDRESS>\<SQL SERVER NAME>, <TCP PORT NUMBER*>.
- **Use Windows authentication.** Select this option to use the Windows user account based authentication mode to login to the SQL Server.

- o   When using the Windows authentication, the Windows users need permissions to the SQL Server in order to be able to use the databases.
- **Server user name.** Type here the SQL Server user name. (Not needed if using Windows authentication)
- **Server user's password.** Password of the SQL Server user. (Not needed if using Windows authentication)

* TCP port number is not needed if the SQL Server uses the default TCP port which is 1433 and, if the UDP port for the SQL Server Browser service is open in your firewall in server side.

Register CustomTools Database

Once the database server is changed, you need to register the CustomTools database in CustomTools and add the SQL Server level message libraries into the new SQL Server. Both of these two operations could be done with help of the **CustomTools Administration Tool**.

Registering CustomTools Database

To register the CustomTools database select the database server node in the **Administration Tool's Database View** and then click the **Register Database** command located in **Server** group of the ribbon.

The **Register Database** dialog box appear.
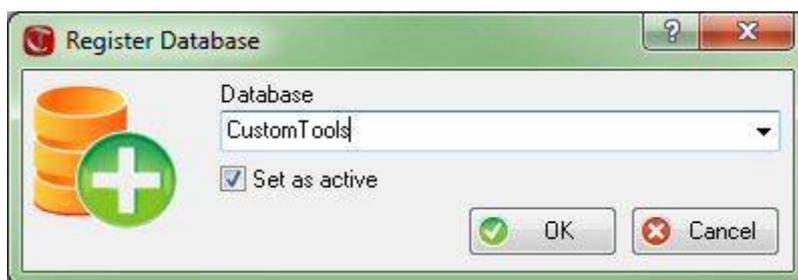
Figure 10. The Register Database dialog box is used to register SQL Server database for CustomTools use

Select the CustomTools database you restored earlier into the SQL Server from the **Database** box and select the **Set as active** option. Then click **OK** to register the database.

Creating a "dummy" Database

Easiest way to add the message libraries to SQL Server is to create one "dummy" database with CustomTools Administration Tool's database creation wizard. To do that, please refer to the following CustomTools help topic:

http://taskpane.customtools.info/en/2013/help/source/CTAdmin/HTML/Wizard_DatabaseCreation.html

For the dummy database, you can go through the database creation wizard using the default settings. Only thing you may want to change is the database name defined in the first page of the database creation wizard.

Take New Database Server to Use in Client Workstations

New database server have to be taken to use also in the client workstations in the CustomTools environment. The database server change could be done with the same instructions as you did it on the Administration Tool but, the way to find the database server settings in other CustomTools products is a bit different. Please refer to the following instructions to find the database server settings from the other CustomTools products:

SOLIDWORKS Add-In

In **SOLIDWORKS**, while having the CustomTools Add-in in use, click **CustomTools, DataServer.**

The **CustomTools: Data Vault** dialog box appear.

Click the **Server Properties** command to open the **Register Database Server** dialog box.

CustomTools Viewer

In **CustomTools Viewer**, click the **DataServer** command located in **Settings** group of the ribbon.

The **CustomTools: Data Vault** dialog box appear.

Click the **Server Properties** command to open the **Register Database Server** dialog box.

Once you have taken the new database server into use in the client workstations, you are done.