



SQL Server Reporting Services: How to Get Started

Presented by:

Geoff Johnson

Database Administrator/Business Intelligence Developer

shelby**systems**

©2013 Shelby Systems, Inc.

Other brand and product names are trademarks or registered trademarks of the respective holders.

Objective

This session will introduce you to the SSRS Report Manager and show you how to start from a blank slate and begin to create your own custom reports.

What You Will Learn

- How to discover the URL for the SSRS Report Manager web interface.
- How to enable a local user account to use and administer the Report Manager.
- How to create or locate a shared data source that multiple reports can use.
- How to use the wizard to create a report in the Report Builder.
- How to use the graphical query builder tool in Report Builder.

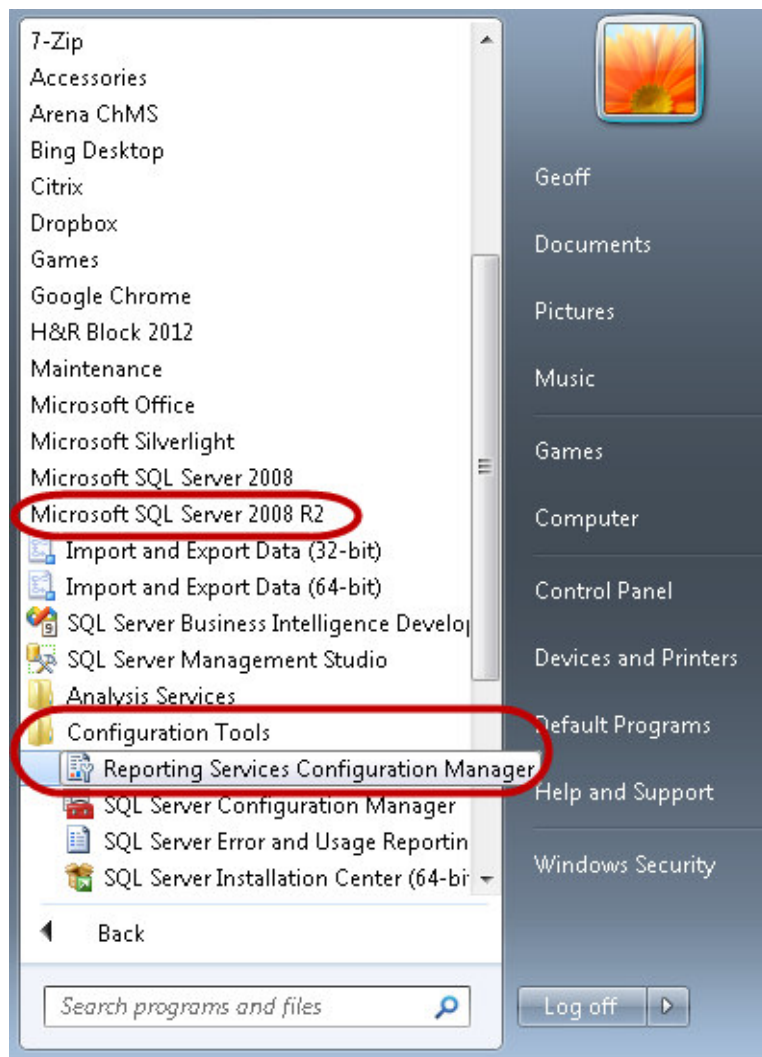
Discover the URL for the SSRS Report Manager

This session will not go into the details of installing SQL Server Reporting Services. It is an optional add-on to SQL Server, and it does require some configuration before it is ready to use. The steps are not difficult, but they should be done by someone familiar with the process.

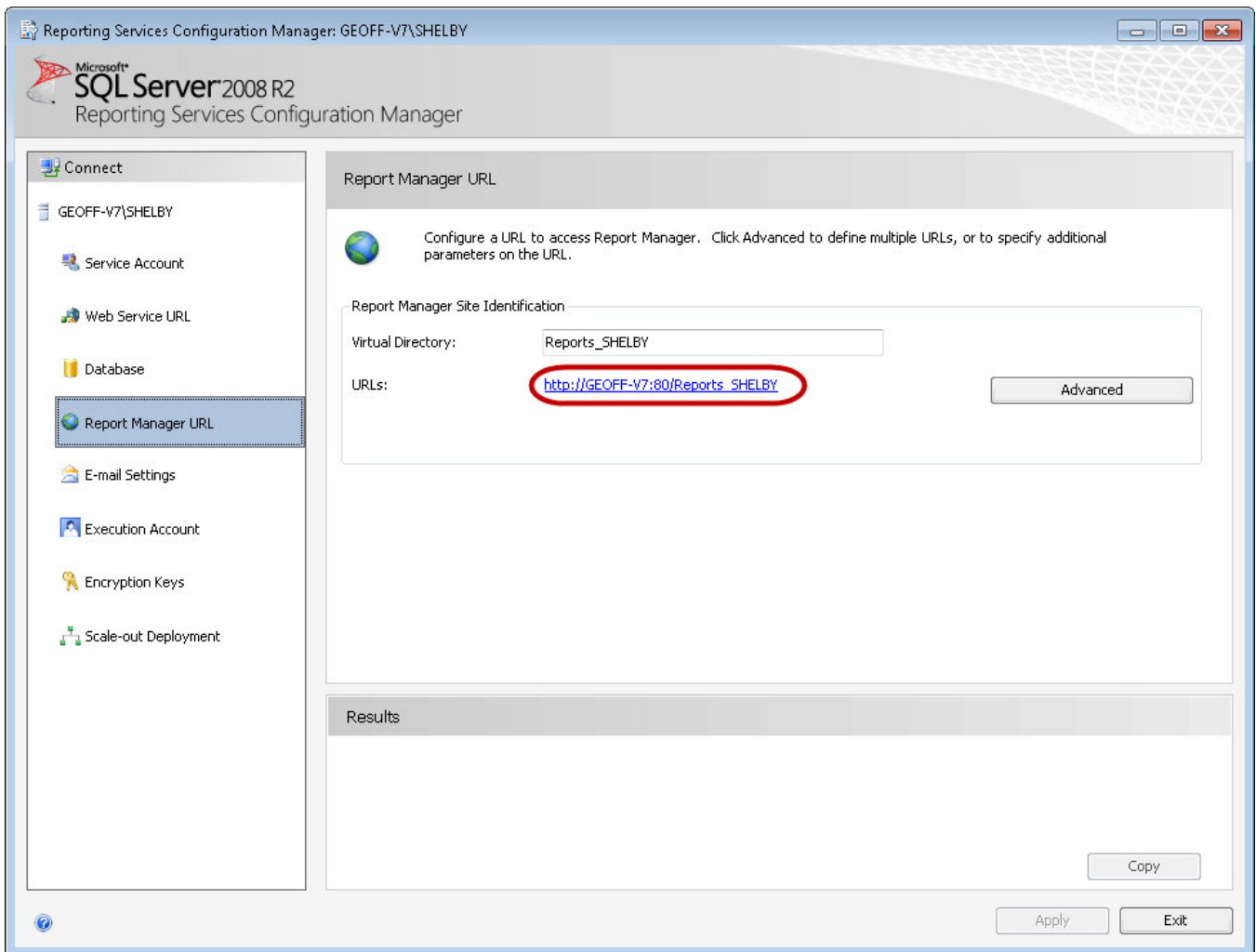
Once Reporting Services is ready to go, you will need to know the URL for the Report Manager website. If your installation follows the default options for a Shelby Systems software solution, the URL will be like this:

`http://<server_name>/Reports_SHELBY`

If you use this URL and the Report Manager does not load, you will need to verify the URL by going to the server and running the Reporting Services Configuration Manager, as shown in the screenshot below.



Once the configuration manager is running, click to the Report Manager URL page. The URL is in blue on that page.



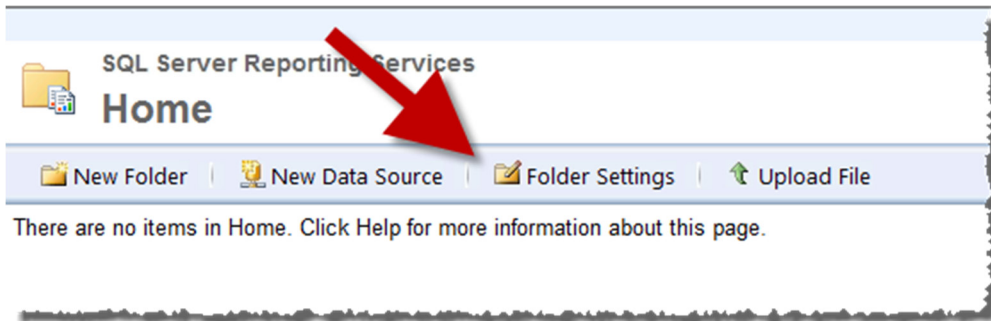
Enable Report Manager for Local User Account

Without the following steps your local Windows user account will not have sufficient permissions to navigate the Report Manager web pages. You have to explicitly give your local account permission by first using the “Run as administrator” option to get to the security settings needed.

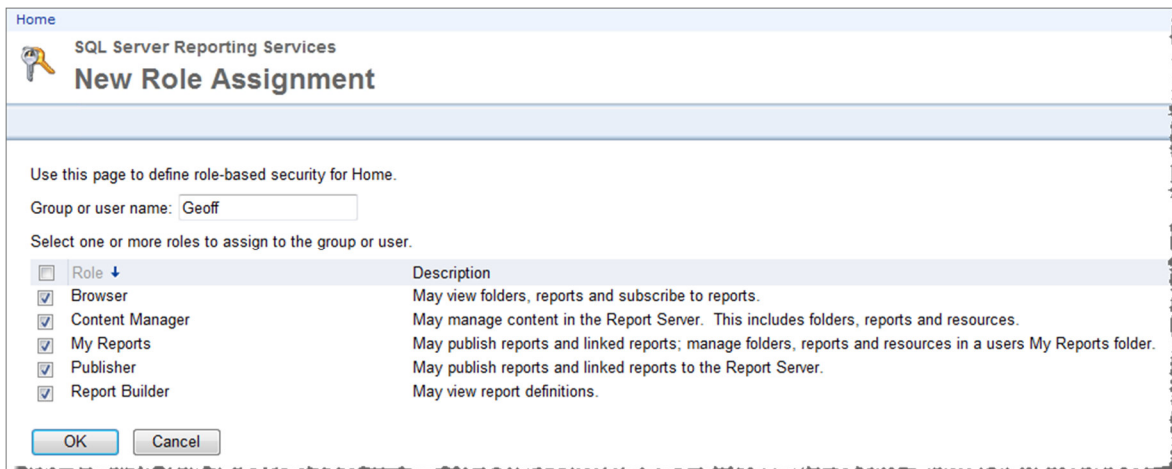
Note: In Windows XP and Windows Vista the “Run as administrator” will not be sufficient; you will need to be logged into Windows as the Administrator, and the Administrator account must have a password assigned to it. The steps for managing the Administrator account password in XP or Vista can be found by searching the Internet for the information.

If at any point during the process described below you are prompted by Internet Explorer to “enable intranet access,” do enable it.

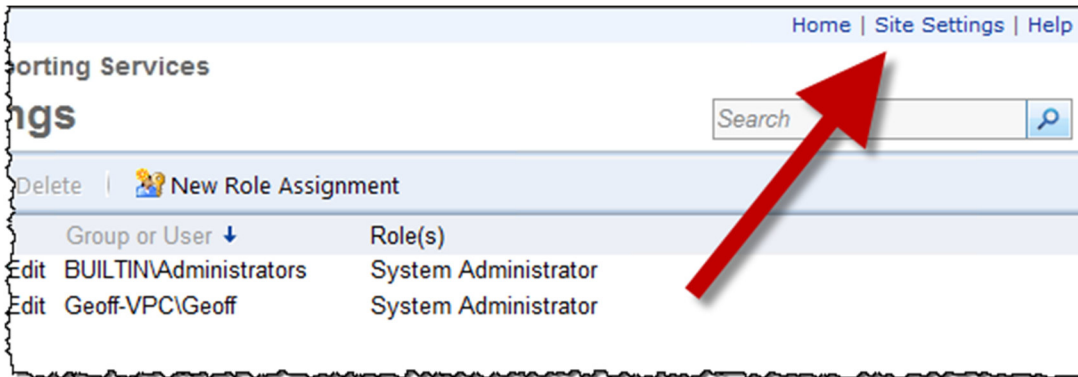
1. Open a web browser using “Run as administrator” option (usually a right-click option).
2. Navigate to the URL of the Report Manager.
3. Click on Folder Settings.



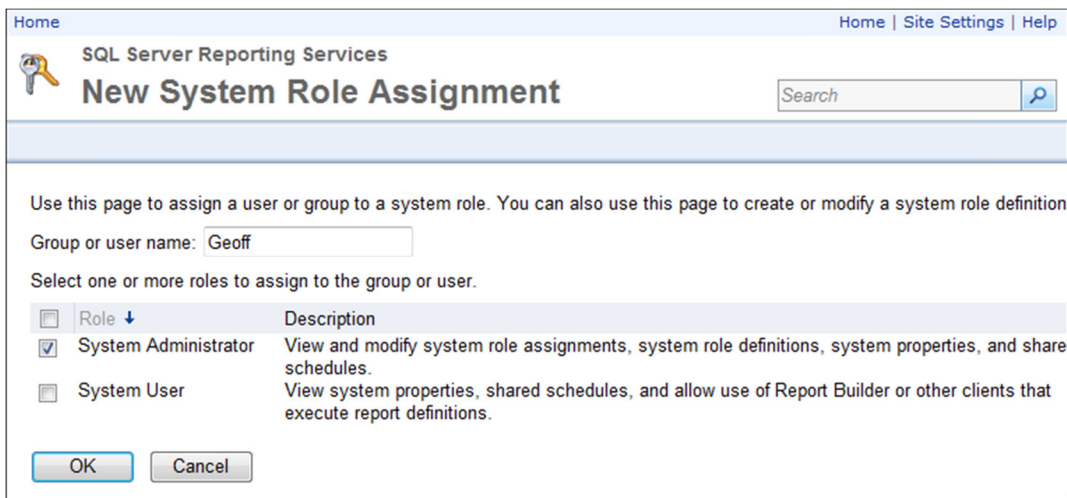
4. Click New Role Assignment.
5. Enter your Windows user account name as the Group or user name.
6. Put a checkmark next to all the roles.



7. Click OK.
8. Click on Site Settings, a small link in the upper right corner of the page.



9. Change to the Security tab.
10. Click New Role Assignment.
11. Enter your Windows user name as the Group or user name.
12. Put a checkmark next to System Administrator.



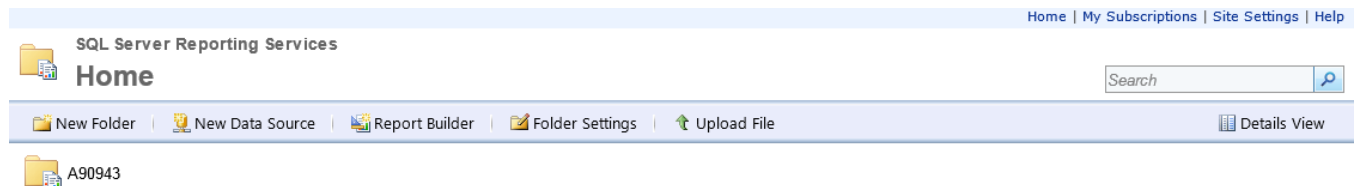
13. Click OK.

Now you can close the web browser and open it normally (not with “Run as administrator”), and you should be able to use and to administer the Report Manager web pages.

Create or Locate a Shared Data Source for Reports

Locate the ArenaDB Data Source

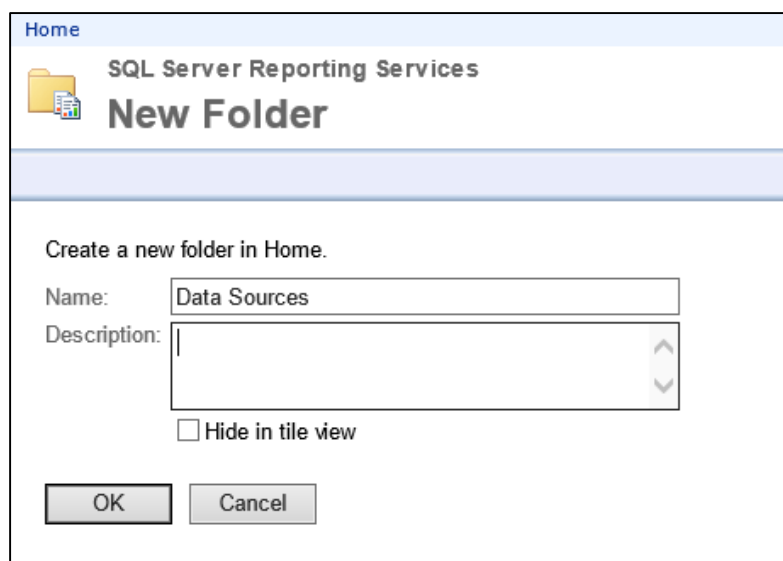
If you are using Arena, you will already have a data source for ArenaDB reports. When you open the Report Manager you will see a folder in the Home folder for Arena reports, as shown in the screenshot below. The folder always begins with the letter A followed by your customer number.



Underneath the AXXXXX folder there is a folder named Arena, and underneath that folder you will see a separate folder for each of the applications in Arena plus a Data Sources folder. There should already be a Data Source set up in this folder for pulling data out of the ArenaDB database. If you want to run Shelby v.5 reports from inside the Arena web interface, you will need to add a ShelbyDB data source as well using the steps outlined in the next segment.

Create a ShelbyDB Data Source

If you are only using Shelby v.5 and not Arena, the Report Manager Home folder will probably be completely empty. The best practice is to put all shared data sources in a Data Sources folder. You can create a Data Sources folder directly in the Home folder by clicking the New Folder link. Just give the folder the name Data Sources and click OK.



Inside the Data Sources folder, click the New Data Source link on the menu bar.

The screenshot below shows the configuration for a ShelbyDB data source, and descriptions of each item are on the next page.

SQL Server Reporting Services
New Data Source

Name:

Description:

Hide in tile view
 Enable this data source

Data source type:

Connection string:

Connect using:

- Credentials supplied by the user running the report
Display the following text to prompt user for a user name and password:

 Use as Windows credentials when connecting to the data source
- Credentials stored securely in the report server
User name:
Password:
 Use as Windows credentials when connecting to the data source
 Impersonate the authenticated user after a connection has been made to the data source
- Windows integrated security
- Credentials are not required

Name: Can be any clear description of the data source. I recommend either the name of the database or the Shelby user account being used to access the ShelbyDB database (i.e. ShelbyREAD, ShelbyCN, etc.).

Description: Can be left blank; provides space for a longer description of the data source.

Hide in tile view: Keep unchecked; this item should be visible in all views.

Enable this data source: Keep checked; this item should be active.

Data source type: Should be *Microsoft SQL Server*, the default value.

Connection string: This is the trickiest item. It consists of two identifiers of types of information and the corresponding values. The first type of information is the *Data Source*, which is the name of the SQL Server instance hosting the database. Normally this is your server name plus a backslash plus the word SHELBY. A semicolon separates the data source value from the next information type, which is the *Initial Catalog*, another way of saying the database name, which will normally be ShelbyDB or ArenaDB. The following is a sample connection string:

Data Source = myserver\Shelby; Initial Catalog = ShelbyDB

Connect using: Unless you want the user to type in a user name and password every time he or she runs the report, select *Credentials stored securely in the report server* and then provide a valid *User name* and *Password*. ShelbyDB has a handful of user account options, listed below. For each password, replace the XXXXX with your five-digit Shelby customer number.

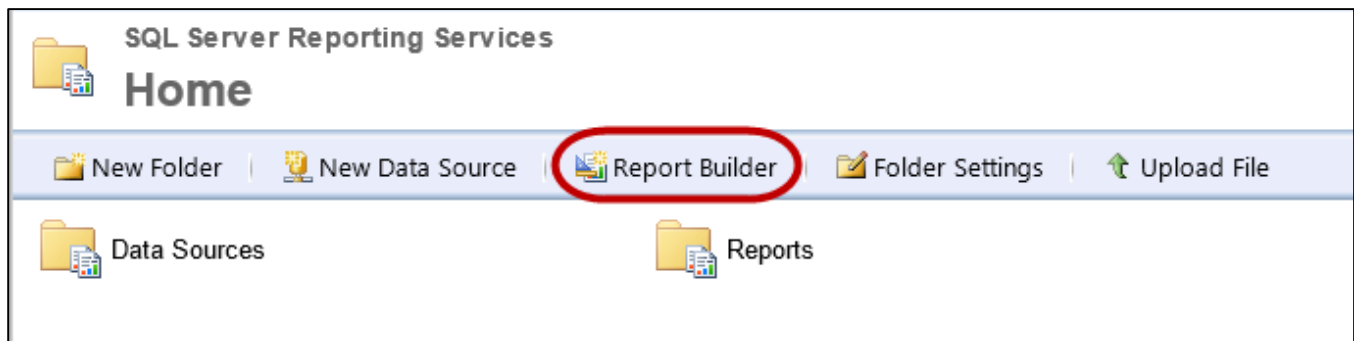
Supported Shelby v.5 Logins			
Login ID	Type	Access to:	Default Password
Sa	Read/Write	Every table	ShelbySystems-2007
ShelbyRead	Read Only	All data except CN, GF, and PR	SsI-01-Rd-XXXXX
ShelbyCN	Read Only	All data except GF and PR	SsI-01-Cn-XXXXX
ShelbyGF	Read Only	All data except CN and PR	SsI-01-Gf-XXXXX
ShelbyPR	Read Only	All data except CN and GF	SsI-01-Pr-XXXXX

After entering the user name and password, click the Test Connection button to verify that all the settings are correct. When they are, the words *Connection created successfully* will appear underneath the test button. Once that appears, click OK to save the new data source.

Prepare a Place to Hold the Reports

If you are creating a report for Arena, the folder structure already exists with places for you to put your reports. If you are creating a report outside the Arena context, you can create your own folder structure and put the report anywhere you wish. Folders can be secured based on the Windows login account, so if you want certain reports to be accessed by only certain people, divide your reports into folders based on security permissions. At a minimum, best practice is to create a basic Reports folder into which all the reports can be stored directly or in subfolders.

Open Report Builder

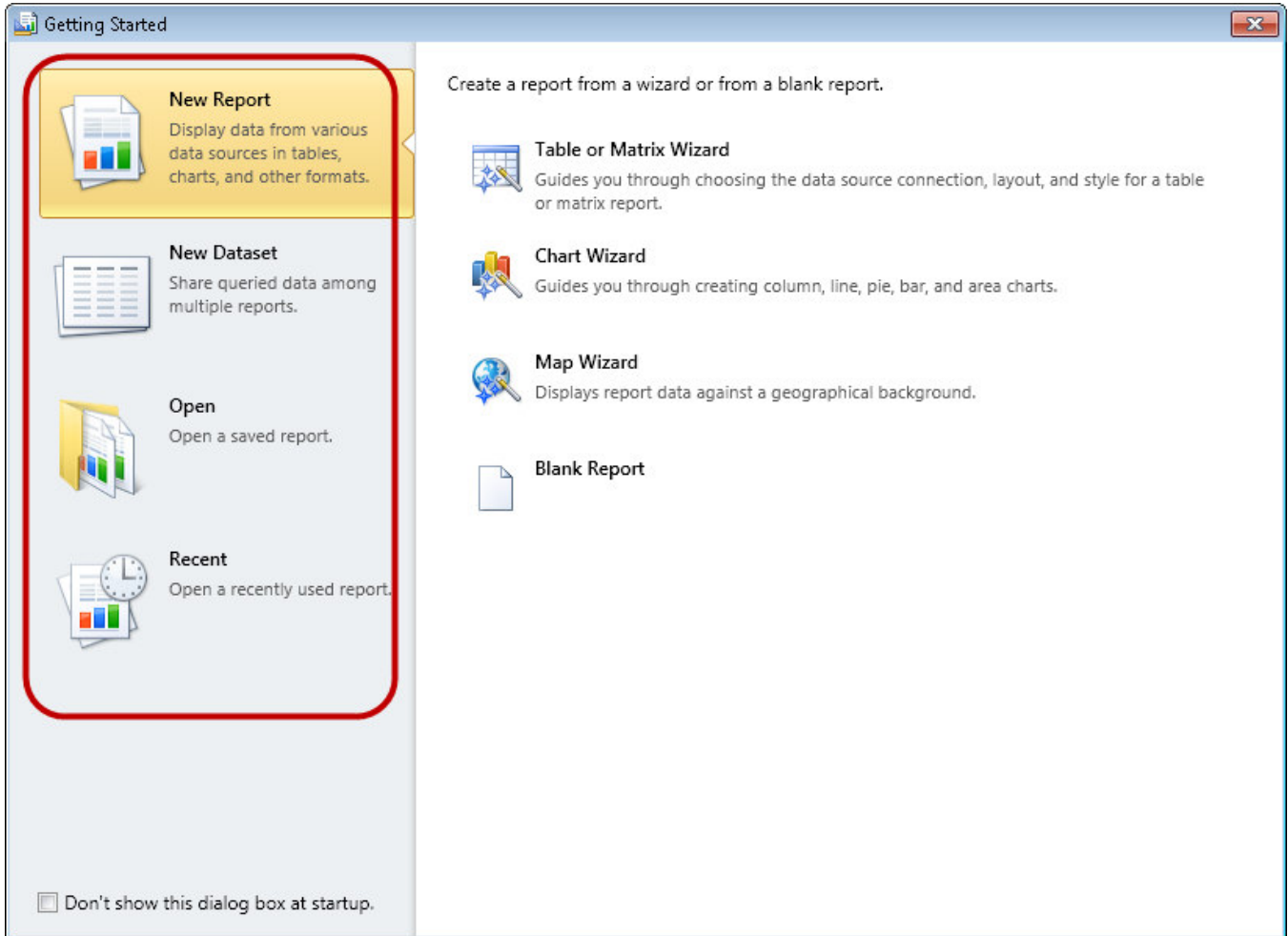


To open Report Builder, simply click the icon on the Report Manager toolbar, circled in the screenshot above.

If you do not have a Report Builder button, there are two possible explanations. One possibility is that you are running SQL Server Express edition. Report Builder is not licensed for Express editions, and therefore you will not have access to this tool. Standard edition is the minimum requirement for using Report Builder. The other possibility is that you do not have permission to access Report Builder. To manage the user permissions, refer to the earlier section on SSRS local user permissions.

Once you click the link to open Report Builder, it will need to download the software first. You may need to give Windows permission to download the software and install it, depending on your local Windows security settings.

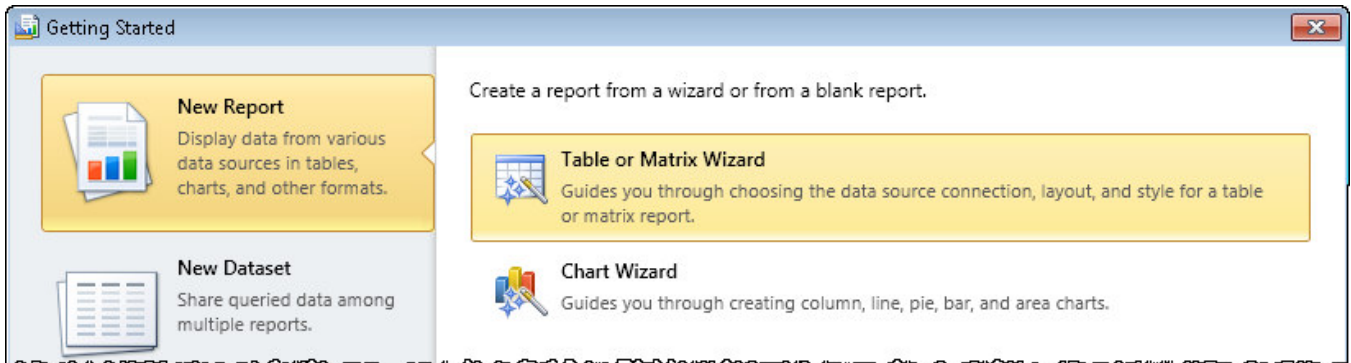
Once Report Builder opens, you will see a “Getting Started” window with several options. You can start a new report, create a new shared dataset (a query that multiple reports can share), open an existing report by browsing the SSRS folders for it, or open an existing report by choosing it from a recently used report list.



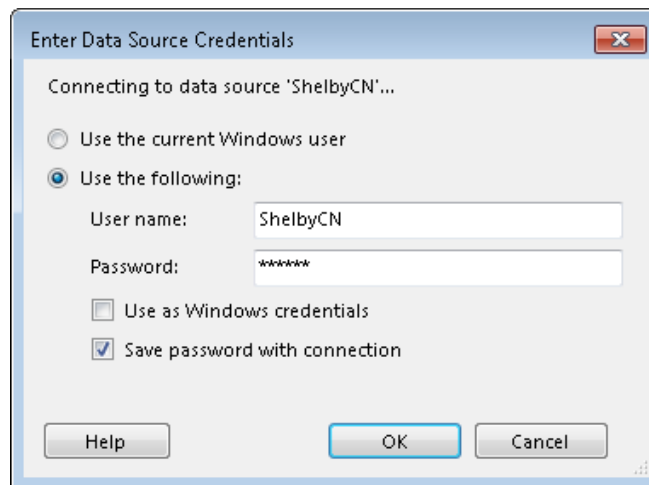
Create a Report Using the Wizard

The example report will answer the question: “What correlation, if any, is there between the reasons for giving and the geographic location of the giver?” The “reasons for giving” will be determined by purpose codes, and the geographic locations will be determined by ZIP codes. The report will show the total giving to purpose codes by the people who live in a common ZIP code.

1. Click on the *Table or Matrix Wizard* link for creating a new report.

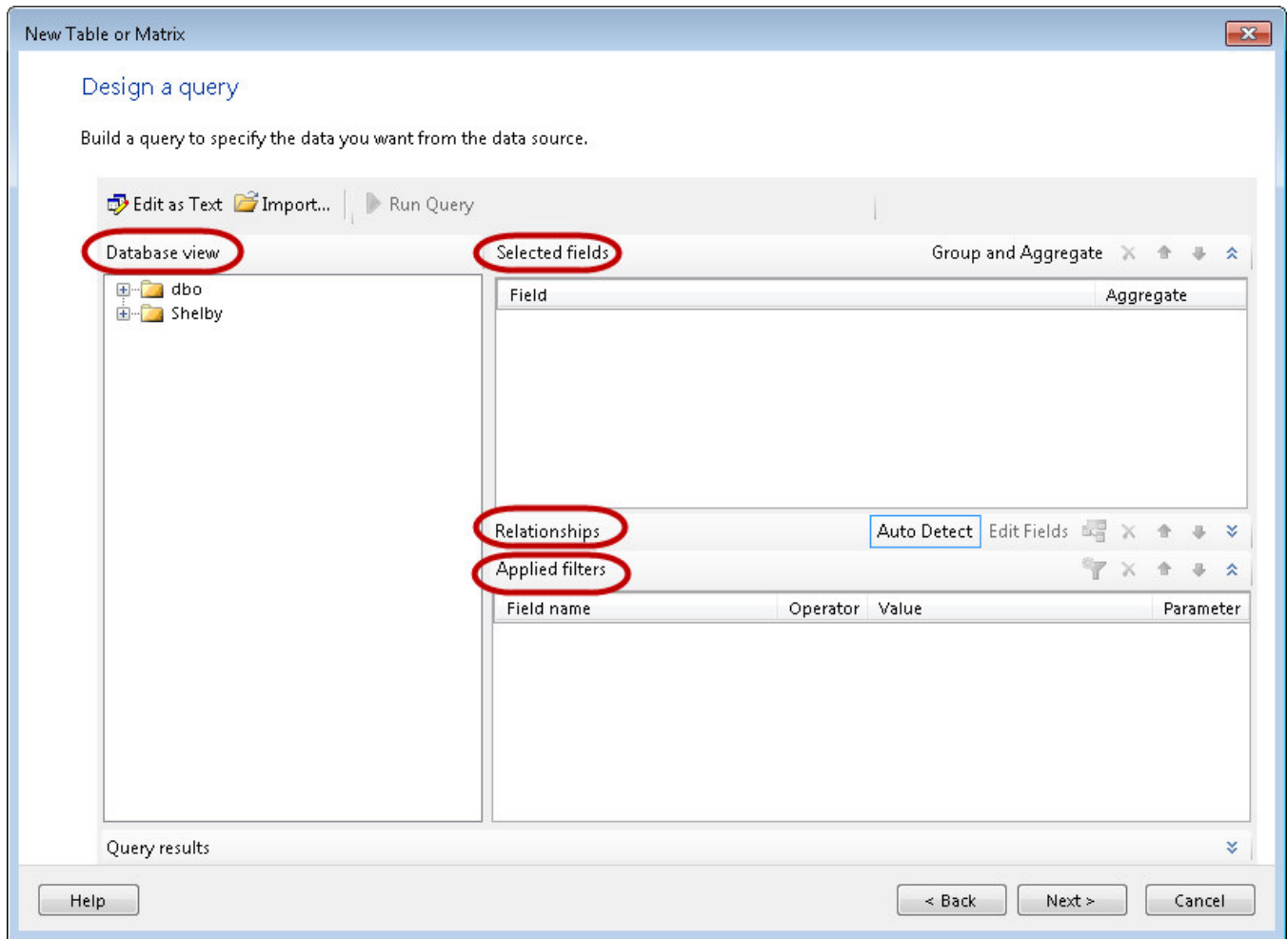


2. Leave the dataset option as *Create a dataset* and click Next.
3. The next step is to choose the data source. The first time you use a new data source, you will need to *Browse* to locate it. Afterward, you will be able to choose the dataset from the list. After selecting the data source, click Next.
4. If prompted for them, provide the credentials again for accessing the data source. Check the box to save the password with the connection to avoid this requestor in the future. Then click OK.



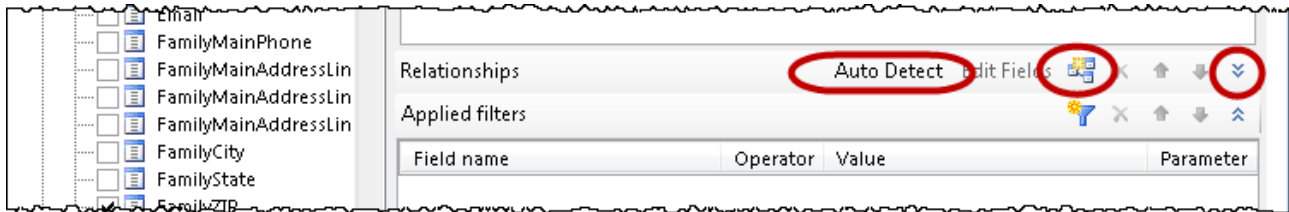
Design the Query Using the Query Builder

The Design a query window is a graphical query editor. It is divided into four parts: the Database View pane, where you select the tables and columns used in the query; the Selected Fields pane, where you can adjust the order of the selected columns and apply aggregate functions if desired; the Relationships pane, where you define the relationships between the tables (this pane starts in its collapsed state); and the Applied Filters pane, where you limit the rows of the query.

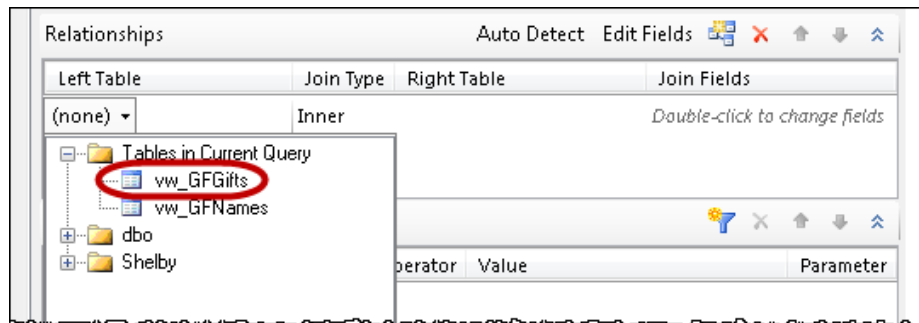


1. In the database view pane, expand the Shelby schema contents by clicking the + symbol to the left of the word. (Shelby v.5 uses its own Shelby schema within the database; Arena uses the default dbo schema.)
2. Expand the Tables or the Views contents by clicking the corresponding + symbol. The example exercise uses two views, so click the + next to the word Views.

3. Expand the specific table or view by clicking the corresponding + symbol, and then put check boxes beside the columns you want to include in the query results. Repeat the process for each table or view that contains columns you want in the results. As you select the columns, they will appear in the selected fields pane. The example will use the following columns:
 - a. vw_GFGifts.PurposeCode
 - b. vw_GFGifts.GiftAmount
 - c. vw_GFNames.FamilyZIP
4. In the selected fields pane, adjust the order of the columns or add aggregate functions as desired. The example exercise does not need any aggregating in the query, so skip this step.
5. In the relationships pane, if the Auto Detect button is in a blue border, click it to turn off the auto detect feature. Then click the Add Relationship button, which is just to the right of the Edit Fields button. And then expand the relationships pane by clicking the down arrow at the far right of the pane. These buttons are shown in the screenshot below.

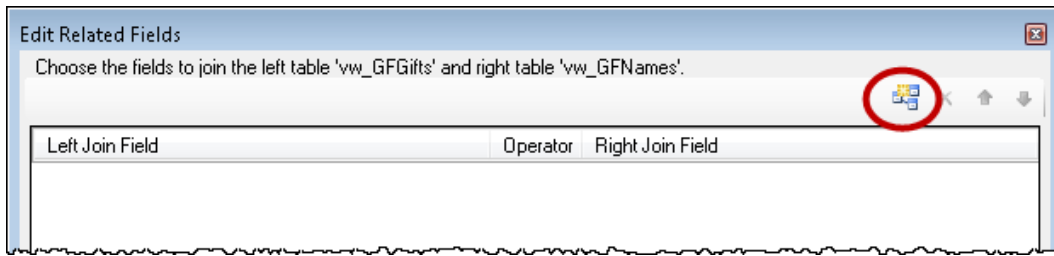


6. Click under the Left Table header to open up a table navigator. Click the table that should be on the left in the join relationship. For the example, click the vw_GFGifts view.



7. Click under the Right Table header and select the table that should be on the right in the join relationship. For the example, click the vw_GFNames view.
8. If necessary, click under the Join Type column to change the join type. For the example, inner join is correct, so skip this step.
9. Double-click under the Join Fields column to open up the join editor.

10. Click the Add Field button on the toolbar, circled in the screenshot below.

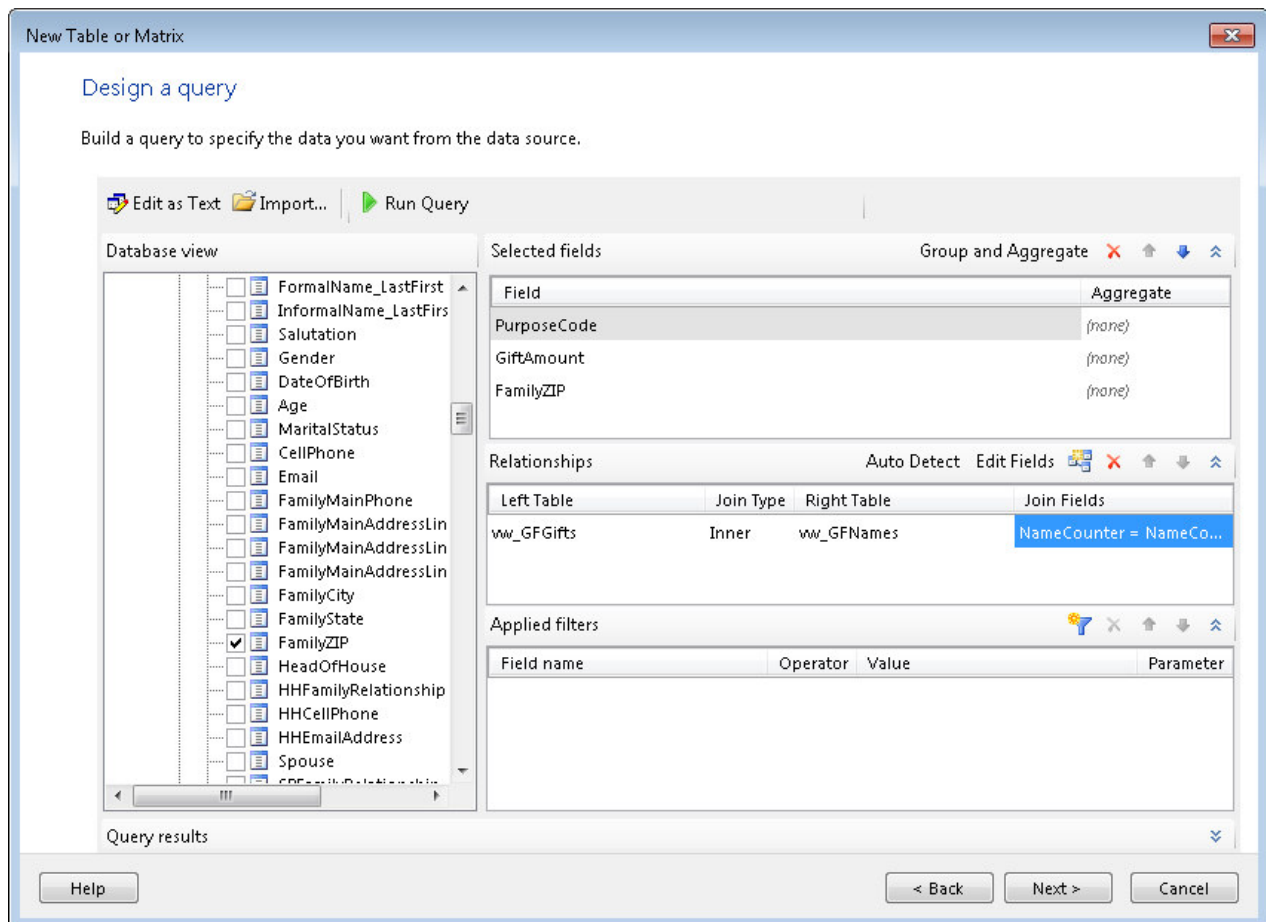


11. Click under the Left Join Field column and select the name of the column that is the foreign key to the left-hand table. For the example, select the NameCounter column name.

12. Click under the Right Join Field column and select the name of the column that is the primary key for the right-hand table. For the example, select the NameCounter column name.

13. Repeat steps 10 – 12 as many times as necessary to define the relationship between the tables or view, and then click OK. For the example, just one join condition is required, so click OK.

14. Add a filter using the similar steps in the applied filters pane. For the example we will not use any filters.

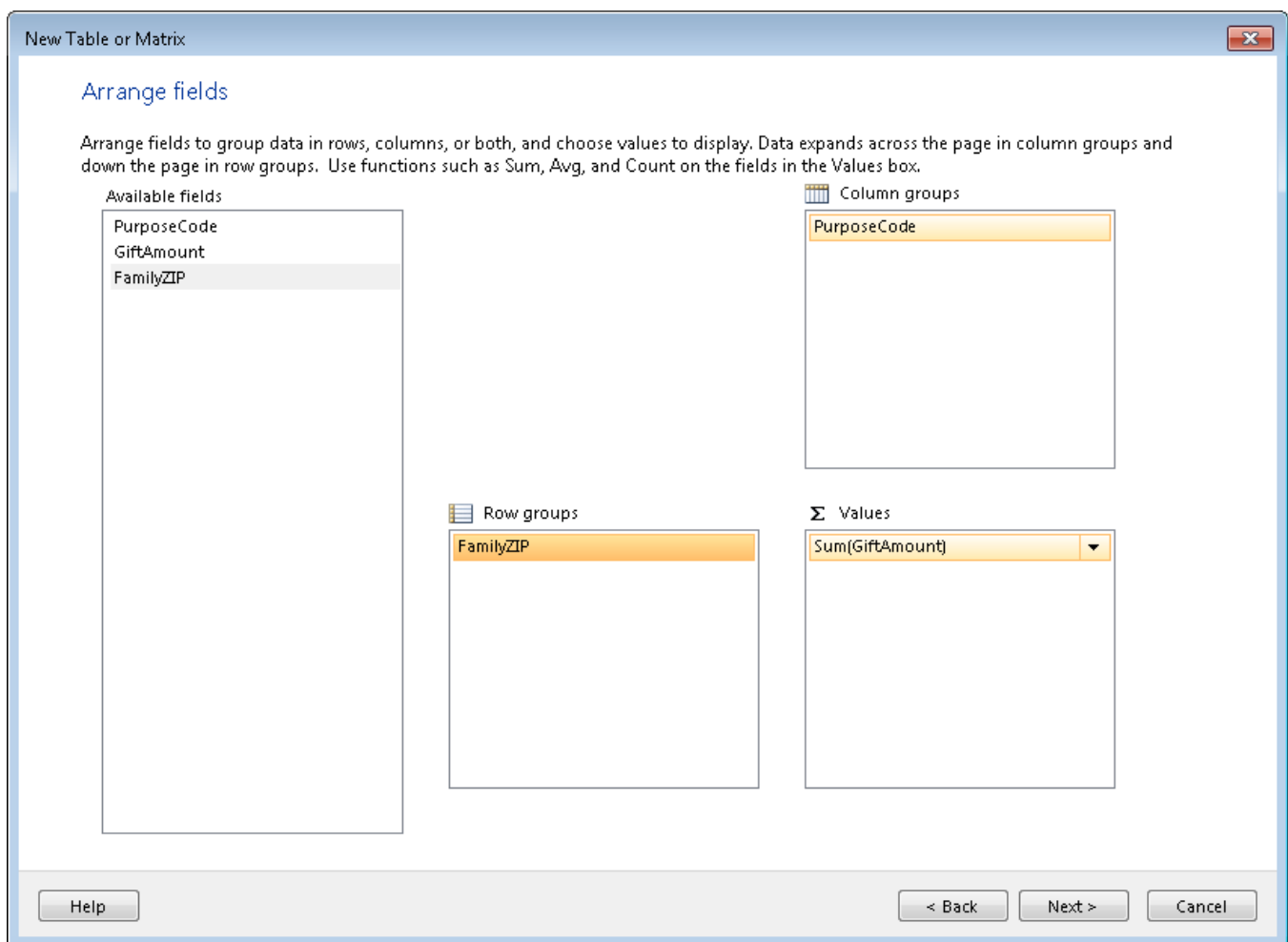


15. Once the query has been fully defined, you can click the Run Query button to preview the results in the query results pane at the bottom of the window, or simply click Next to advance to the next step in the wizard.

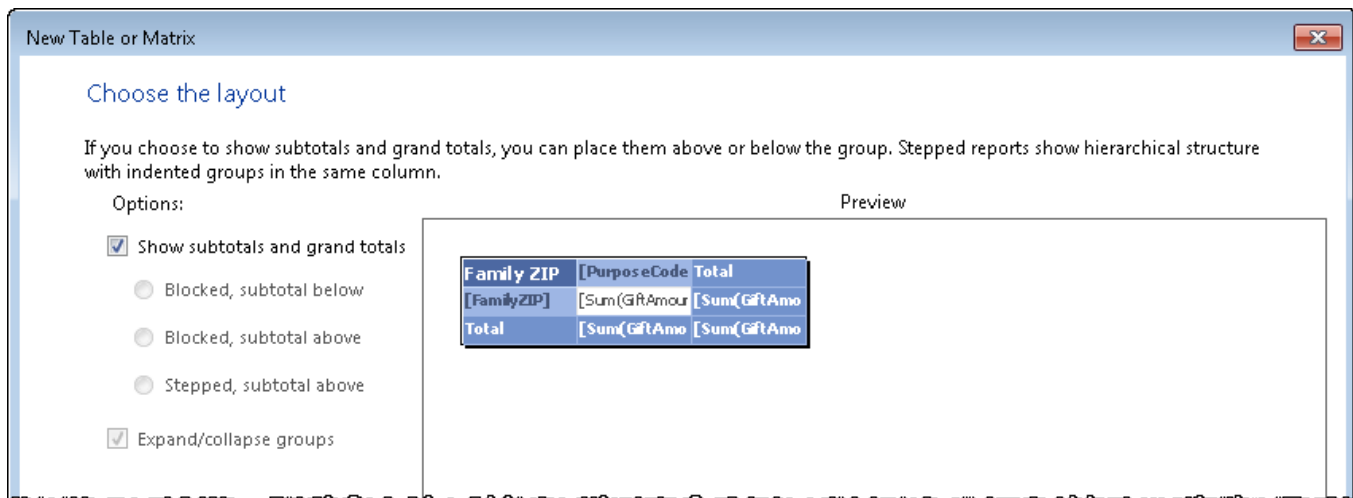
Define the Matrix

The next step in the wizard is to put the available fields into their spaces in the matrix, which is very similar to laying out a PivotTable in Excel.

1. Drag and drop the available fields into the corresponding sections of the matrix output. Only add one field for each grouping level. Then click Next. Later, you can add more fields to the layout. For the example we want PurposeCode to column groups, FamilyZIP to row groups, and GiftAmount to summary values.



2. You can specify whether or not to include totals. If there are multiple levels in the column or row groups, you can also choose whether the subgroup(s) are collapseable for a “drill-down” effect. For the example, make no changes and click Next.



3. You can choose a “style,” which is a pre-defined font and color scheme. There are not many options and all of them are purely cosmetic. Choose whichever you want and click Finish.

Add the Finishing Touches

After the wizard completes, there will probably be a few more steps to take before you are ready to save the report and call it complete. Typically this means adding your own title to the report, adjusting the number formatting, and personalizing the layout in other ways.

1. Click on the report where you see the words “Click to add title.” Enter your own title for the report. For the example, the report could be called *Giving by Purpose and ZIP Code*.
2. Select the four text boxes that have SUM(GiftAmount) in them, just as you would select multiple cells in an Excel spreadsheet. Then go over to the Properties pane locate the Number > Format value box. Type C2 in as the format value, and then press Enter. This formats the values as currency with two decimal places.
3. Click the Run button in the toolbar at the top left to see the report in action.
4. Click the Design button in the same place where the Run button was to return to the design view.
5. Click the Save icon in the very top toolbar to save the report. Click into the Reports folder to save the report there as Giving by Purpose and ZIP Code.



Geoff Johnson
Database Administrator/
Business Intelligence Developer
geoff.johnson@shelbyinc.com



Geoff Johnson has been working with Shelby Systems since 2003 and is currently the Database Administrator/Business Intelligence Developer. As a DBA his primary task is improving the effectiveness of the storage and delivery of the data inside each Shelby software solution. As a BI Developer his primary task is providing quality reports that facilitate the work of our customers and thereby expand the Kingdom of God.

Geoff has a passion for bringing the latest business intelligence innovations from Microsoft to the ministry of the church. He has helped many Shelby Systems customers implement Reporting Services solutions, PivotTable and PowerPivot analytical reports, and other custom report designs ranging in complexity from simple directories to complex GAAP-compliant financial statements.

He has worked closely with several denominational headquarters offices to improve and streamline their data collection and reporting processes.

shelby**systems**[®]