BIDS Add-In &
Reporting Services Extensions

Installation, Configuration and Getting Started

Release 3.0.0

Installation

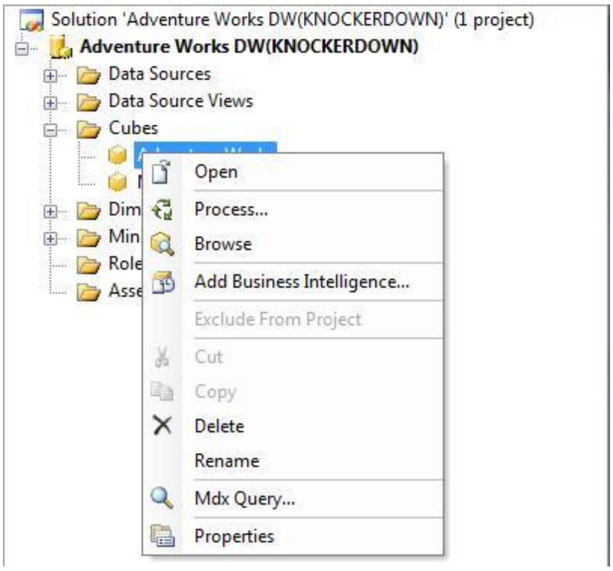
Download the install package from the location provided and run Setup.exe to install.

NOTE: When installing on VISTA or Windows Server 2008 you will need to disable UAC during the install process. UAC can be switch back on after the installation.

The install procedure will install the following:

IQ for BIDS add-in

Allows you to use iT-Workplace's Investigator query builder in the BIDS (Business Intelligence Development Studio) development environment. The query builder can be accessed from the "Mdx Query..." menu option added to the context menu when you select a cube in a BIDS.



IQ – Analysis Services Pivoted data extension

Intelligencia Query Analysis Services Pivoted data extension allows you to create reports in SQL Server Reporting Services that utilize any Mdx expression without the restrictions imposed by the standard Microsoft SQL Server Analysis Services data source.

The tool consists of a custom data extension and a custom query designer that work within the standard Reporting Services tools surfaced through Visual Studio, SQL Server Management Studio and the web.

The custom data extension allows any valid Mdx query to be rendered in reporting services without the current restrictions imposed by the RS Analysis Services data source. In other words you can place hierarchies other than measures on columns including nested dimensions etc.

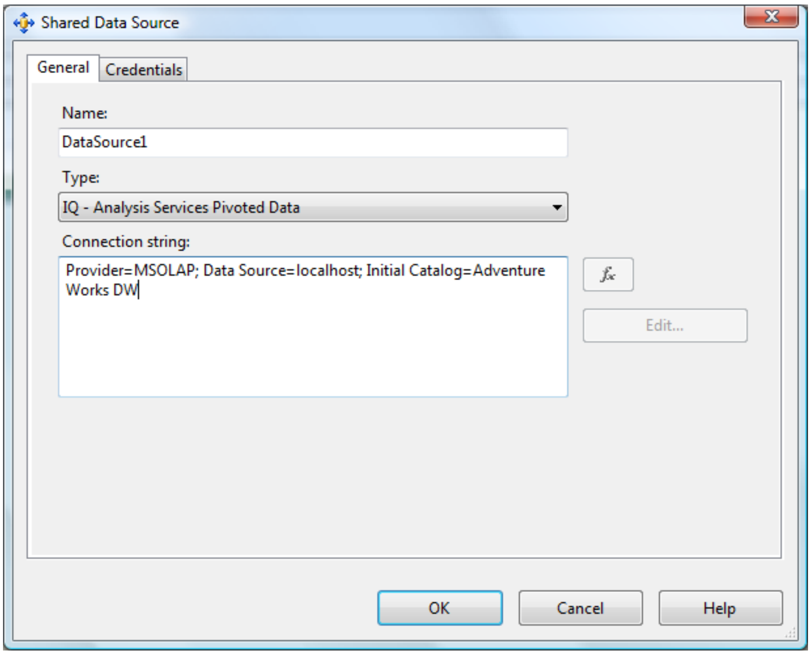
The custom query designer provides a state-of-the-art UI for building Mdx queries and will handle creation of parameterized Mdx for reports.

NOTE: Setting the data extension up on a production server is described later in this document under the heading "Deploying to a Report Server".

Creating a Data Source

Create a new Reporting Services project. Create a new shared data source with a source type of "IQ – Analysis Services Pivoted Data". You will need to provide a standard connection string with the minimum of Provider, Data Source and Inital Catalog specified.

Note: If you get the error message "The server is not started or too busy" you will need to use the fully qualified provider specification "Provider = MSOLAP.3" rather than the default level of MSOLAP. There was a problem some time ago in the Microsoft install procedures that allocated MSOLAP an older version of the provider.



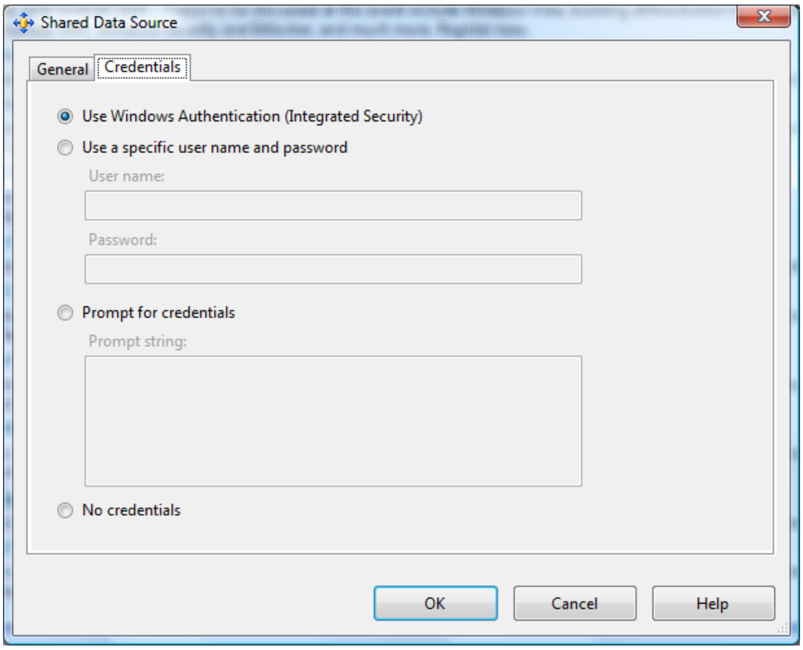
Shared Data Source

Name: DataSource1

Type: IQ - Analysis Services Pivoted Data

Connection string: Provider=MSOLAP; Data Source=localhost; Initial Catalog=Adventure Works DW

On the credentials tab you will probably want to select "Use Windows Authentication (Integrated Security)" unless you are using SQL authentication.



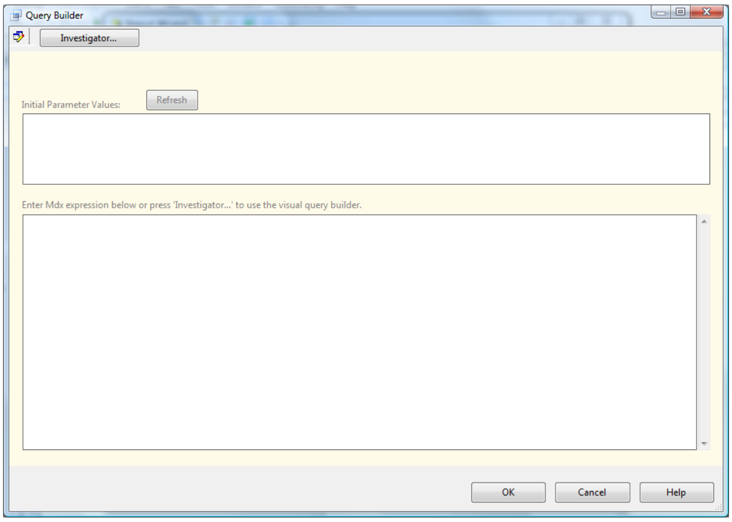
● Use Windows Authentication (Integrated Security)
○ Use a specific user name and password
○ Prompt for credentials
○ No credentials

Creating Reports

IQ provides 2 ways to create a report. You can build a “Standard Report” using the normal visual studio report designer interface or you can use the “Formatted Report” tool which will creates a complete rdl document.

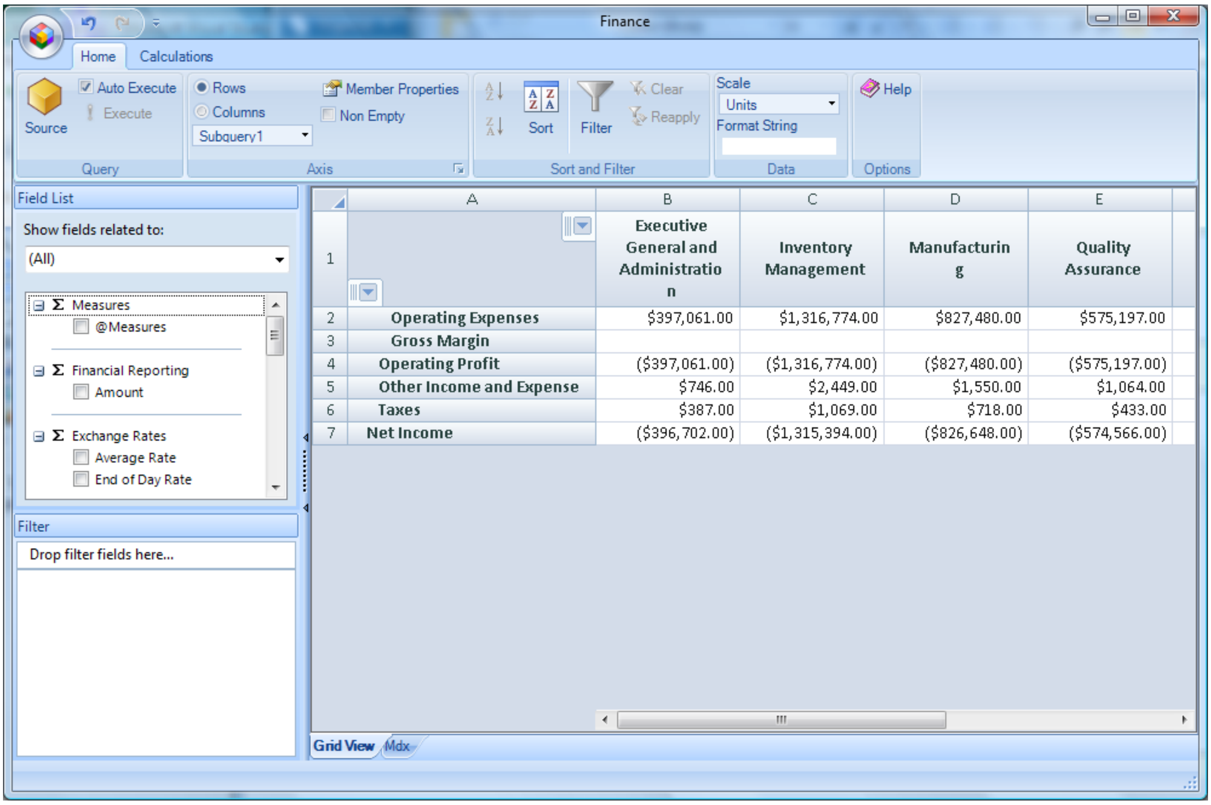
Creating a Standard Report

To create a new report start the report wizard as you normally would selecting your data source. On the second page of the report wizard select “Query Builder...” and the custom query builder window will be displayed:



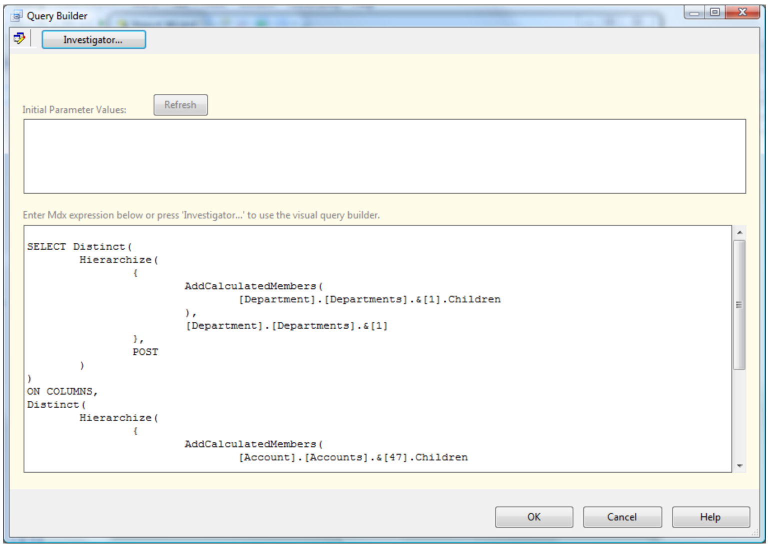
Enter Mdx expression below or press 'Investigator...' to use the visual query builder.

Press the “Investigator...” button to launch the query builder window. Select a cube and then use the drag and drop interface to create a query. Note that there are no restrictions on how you structure the query, you can place any hierarchy/hierarchies on the column axis.



	A	B	C	D	E
1		Executive General and Administration	Inventory Management	Manufacturing	Quality Assurance
2	Operating Expenses	$397,061.00	$1,316,774.00	$827,480.00	$575,197.00
3	Gross Margin				
4	Operating Profit	($397,061.00)	($1,316,774.00)	($827,480.00)	($575,197.00)
5	Other Income and Expense	$746.00	$2,449.00	$1,550.00	$1,064.00
6	Taxes	$387.00	$1,069.00	$718.00	$433.00
7	Net Income	($396,702.00)	($1,315,394.00)	($826,648.00)	($574,566.00)

When you exit from the query builder the Mdx expression generated will be displayed.

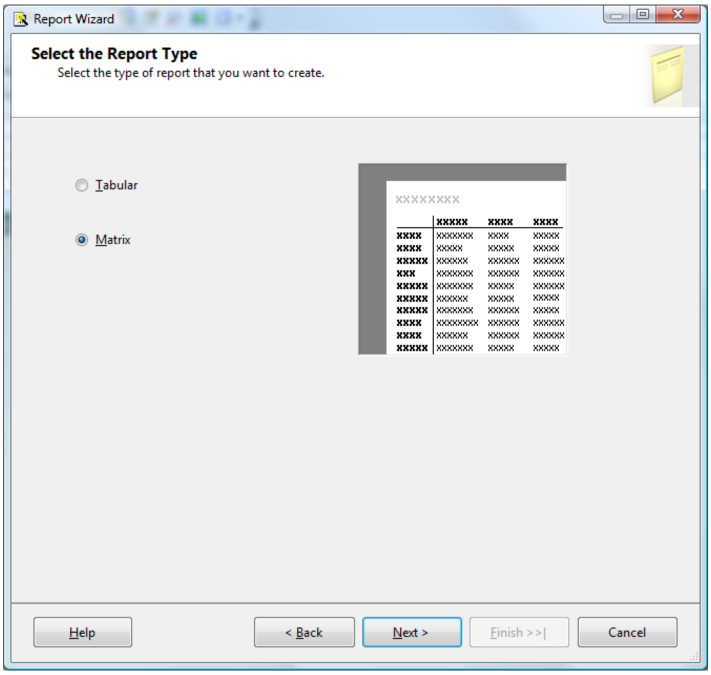


Enter Mdx expression below or press 'Investigator...' to use the visual query builder.

```
SELECT Distinct(
        Hierarchize(
                {
                        AddCalculatedMembers(
                                [Department].[Departments].&[1].Children
                        ),
                        [Department].[Departments].&[1]
                },
                POST
        )
)
ON COLUMNS,
Distinct(
        Hierarchize(
                {
                        AddCalculatedMembers(
                                [Account].[Accounts].&[47].Children
```

Press OK to exit from the Query Builder window and then press “Next>”

Select a Matrix report in the “Select Report Type” window and press “Next>”



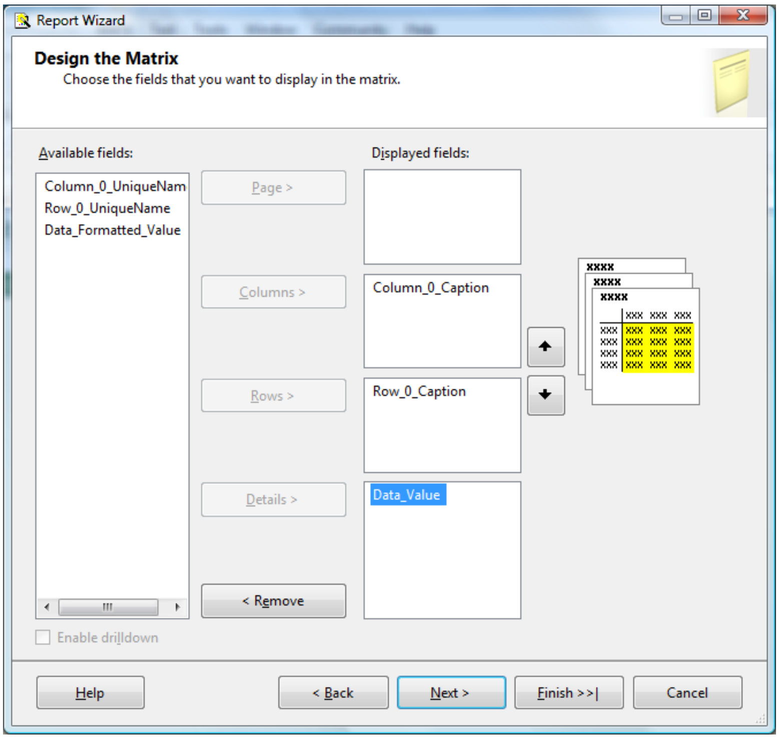
Select the Report Type

Select the type of report that you want to create.

○ Tabular

● Matrix

The list of available fields in the “Design the Matrix” window will show entries for Caption and Unique name for each hierarchy on the row and column axes along with unformatted and formatted data values. Drag these into an appropriate layout.



Design the Matrix

Choose the fields that you want to display in the matrix.

Available fields:

Column_0_UniqueNam
Row_0_UniqueName
Data_Formatted_Value

Displayed fields:

Page >

Columns > Column_0_Caption

Rows > Row_0_Caption

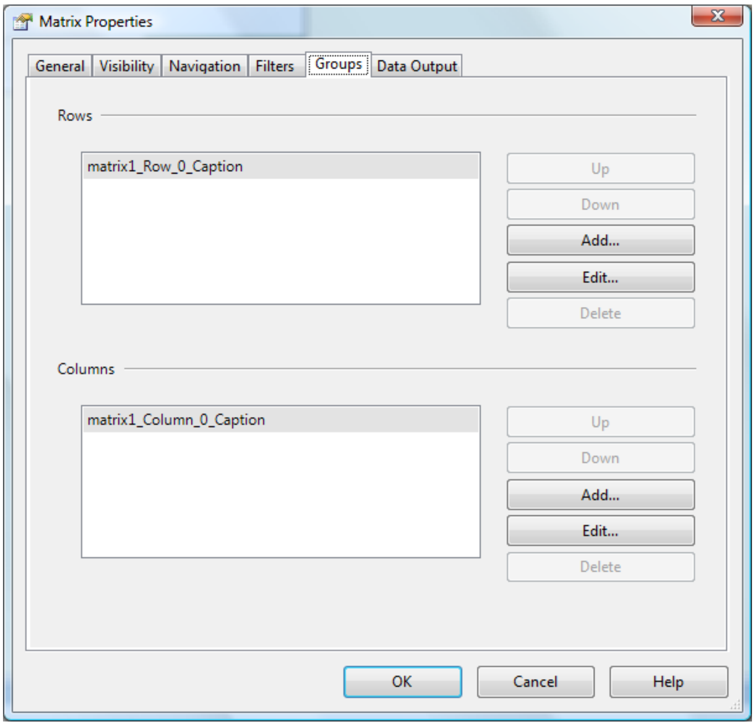
Details > Data_Value

☐ Enable drilldown

You can now finish the wizard as normal:

Report Layout

RS will apply a default sort sequence when you create a Matrix report. To get rid of this right click on the matrix in the "Layout" tab of the designer and select "Properties" then select the "Groups" tab:



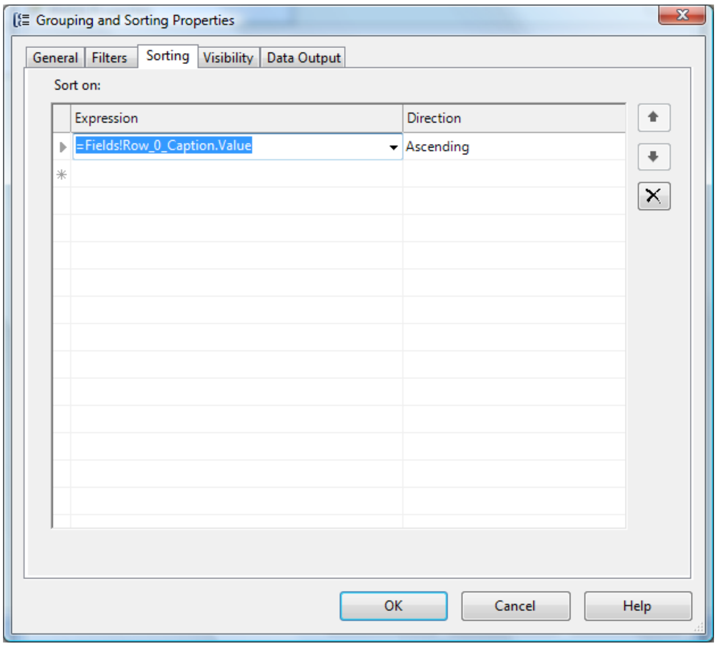
Rows

matrix1_Row_0_Caption

Columns

matrix1_Column_0_Caption

Next select the "Edit..." button for Rows and/or columns and in the "Grouping and Sorting Properties" dialog select the "Sorting" tab.

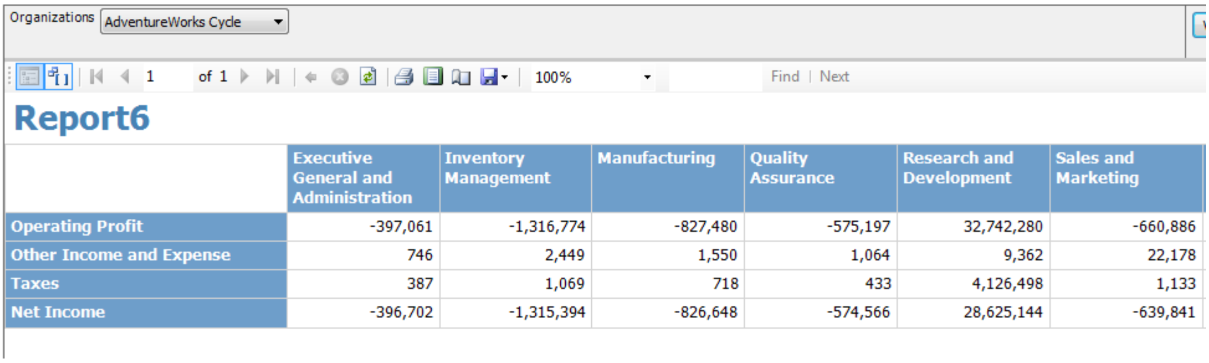


Sort on:

Expression	Direction
=Fields!Row_0_Caption.Value	Ascending

You can select the sort row and delete it.

With a minimal amount of addition tweaking your report should look something like this:



Organizations: AdventureWorks Cycle

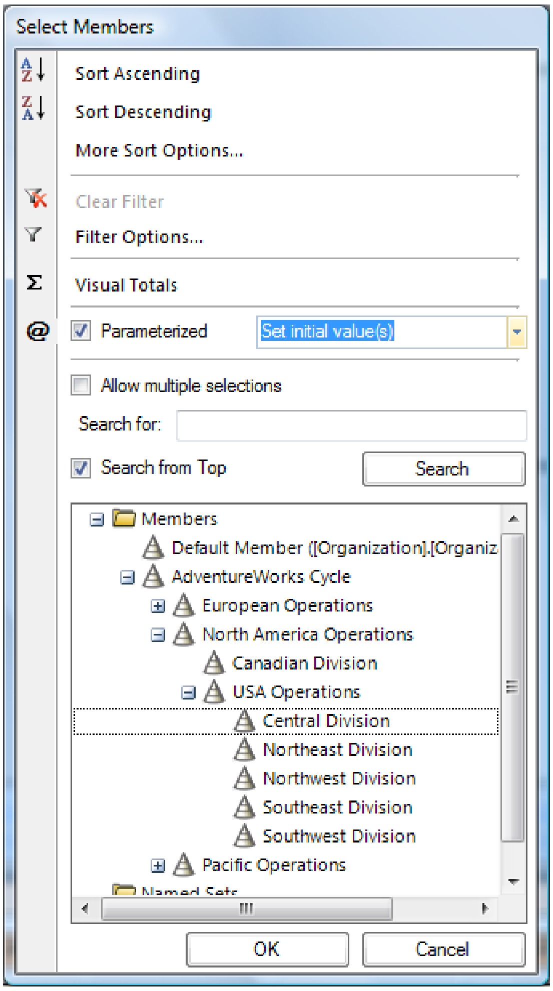
Report6

	Executive General and Administration	Inventory Management	Manufacturing	Quality Assurance	Research and Development	Sales and Marketing
Operating Profit	-397,061	-1,316,774	-827,480	-575,197	32,742,280	-660,886
Other Income and Expense	746	2,449	1,550	1,064	9,362	22,178
Taxes	387	1,069	718	433	4,126,498	1,133
Net Income	-396,702	-1,315,394	-826,648	-574,566	28,625,144	-639,841

Parameterized Reports

In the Investigator you can create report parameters in the following way:

From the Select Members window check the Parameterized option which is indicated by the “@” icon.

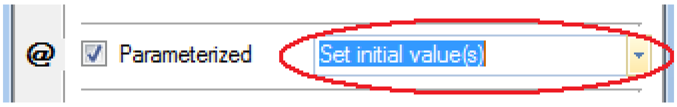


☑ Parameterized: Set initial value(s)

☐ Allow multiple selections

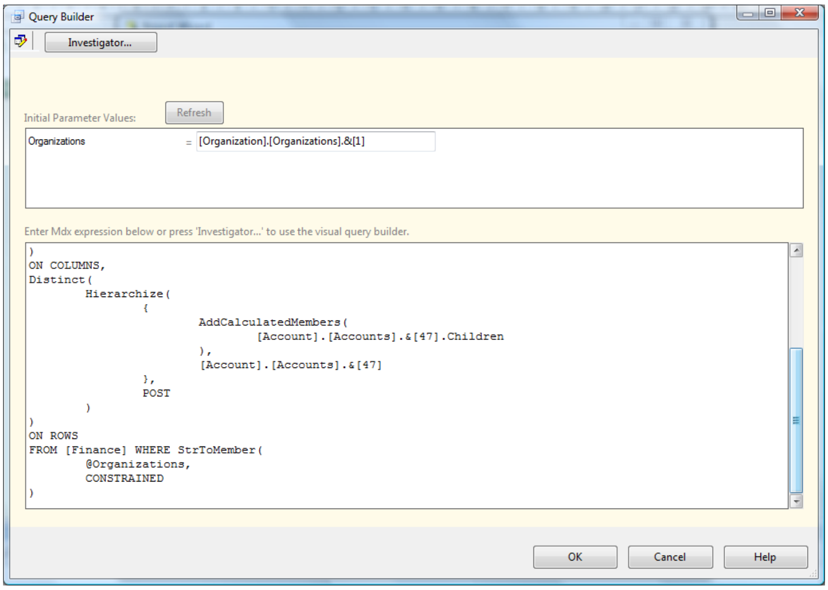
☑ Search from Top

Make sure that the drop down list for parameter selection is set to the “Set Initial value(s)” option. This allows you to select a value that will be used to create a dataset that is used for the rest of the report design process in Visual Studio.



Now you can select an initial value for the parameter.

When you exit from the Investigator the query will contain a reference to the parameter and the “Initial Parameter Value” window will contain the selected value. Note that this initial value is NOT the default value used when the report is rendered; you will need to set this up manually later in the parameter definition process. The initial value is used to generate a valid cellset that can be used to complete the rest of the wizard.



Initial Parameter Values:

Organizations = [Organization].[Organizations].&[1]

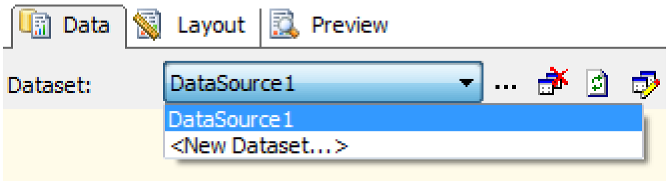
Enter Mdx expression below or press 'Investigator...' to use the visual query builder.

```
)
ON COLUMNS,
Distinct(
        Hierarchize(
                {
                        AddCalculatedMembers(
                                [Account].[Accounts].&[47].Children
                        ),
                        [Account].[Accounts].&[47]
                },
                POST
        )
)
ON ROWS
FROM [Finance] WHERE StrToMember(
        @Organizations,
        CONSTRAINED
)
```

Creating a dataset to populate report parameters

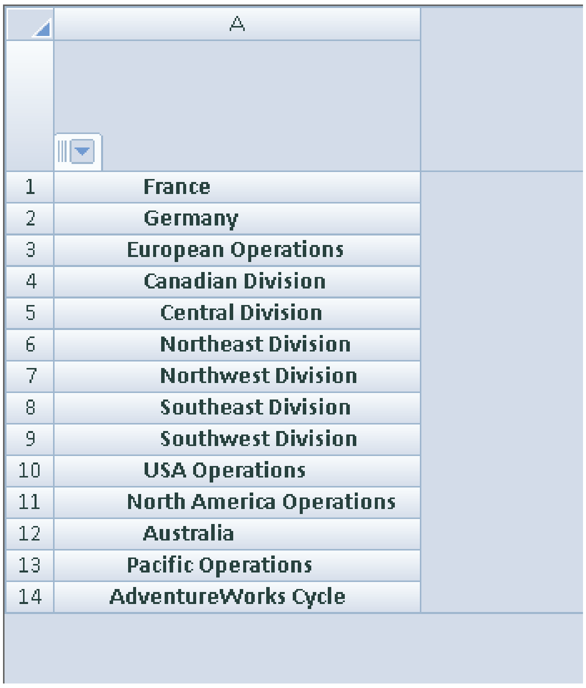
The wizard will not automatically create a dataset to populate the report parameters so you will need to do this yourself.

In the data window create a new dataset:



You can use your IQ data source for this:

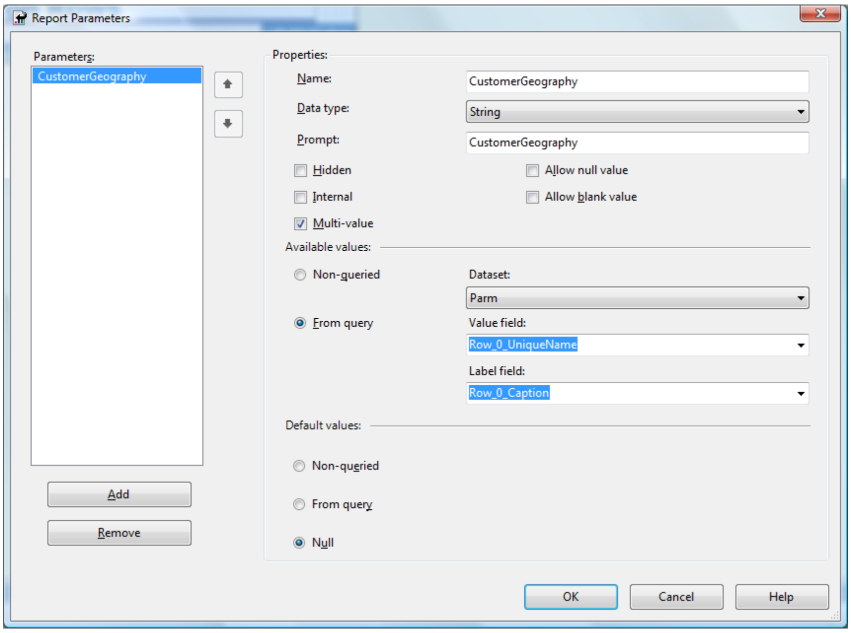
Using the investigator create a query that returns the captions for your parameter selection as rows in the grid. Your query should look something like the following:



	A
1	France
2	Germany
3	European Operations
4	Canadian Division
5	Central Division
6	Northeast Division
7	Northwest Division
8	Southeast Division
9	Southwest Division
10	USA Operations
11	North America Operations
12	Australia
13	Pacific Operations
14	AdventureWorks Cycle

Exit from the Investigator and select the Layout tab in the report designer. Now right mouse click and select “Report Parameters”.

The data set can be selected as shown using the “Row_0_Caption” and “Row_0_UniqueName” attributes for the “Label Field” and “Value Field” attributes.



Report Parameters

Parameters:
CustomerGeography

Properties:
Name: CustomerGeography
Data type: String
Prompt: CustomerGeography
☐ Hidden
☐ Allow null value
☐ Internal
☐ Allow blank value
☑ Multi-value

Available values:
○ Non-queried
● From query
Dataset: Parm
Value field: Row_0_UniqueName
Label field: Row_0_Caption

Default values:
○ Non-queried
○ From query
● Null

Enhancing Your Reports

The IQ ASPivot Dataset will automatically create the fields “Column_IsTotal” and “Row_IsTotal”. The value of these fields will be set to True for each member in a cellset that represents a total within the report.

The algorithm that sets these flags assumes the following:

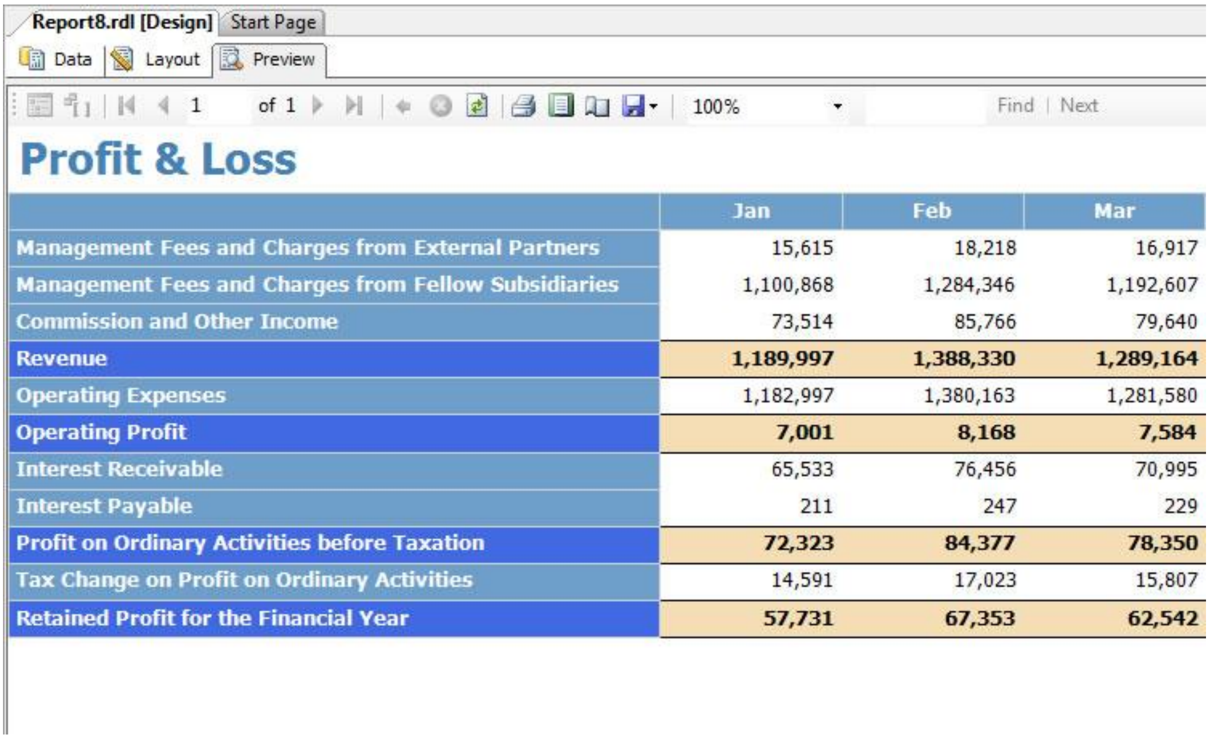
1. The flag is set for the innermost hierarchy on the axis only.
2. The flag assumes that the members are ordered in POST hierarchize order, i.e. with the totals at the bottom of each group on rows or to the right of each group on columns.
3. The flag will be set to true if a member has children and one or more of those children is displayed.

By using these flags in custom formatting expressions you can create attractive reports in which the aggregate values are highlighted as shown below.

In this example the Background colour attribute for the cells in the report was set to:

```
=IIf(Fields!Row_IsTotal.Value=True, "Wheat",
       IIf(Fields!Column_IsTotal.Value, "Wheat", "Transparent")
 )
```

Note: By nesting the references to the row and column total indicators within a compound Iif statement you can set attributes for both rows and columns.



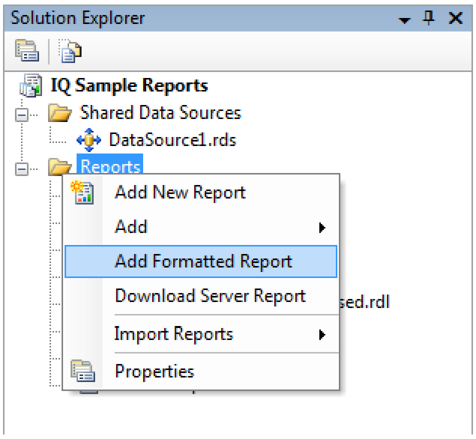
Profit & Loss

	Jan	Feb	Mar
Management Fees and Charges from External Partners	15,615	18,218	16,917
Management Fees and Charges from Fellow Subsidiaries	1,100,868	1,284,346	1,192,607
Commission and Other Income	73,514	85,766	79,640
Revenue	**1,189,997**	**1,388,330**	**1,289,164**
Operating Expenses	1,182,997	1,380,163	1,281,580
Operating Profit	**7,001**	**8,168**	**7,584**
Interest Receivable	65,533	76,456	70,995
Interest Payable	211	247	229
Profit on Ordinary Activities before Taxation	**72,323**	**84,377**	**78,350**
Tax Change on Profit on Ordinary Activities	14,591	17,023	15,807
Retained Profit for the Financial Year	**57,731**	**67,353**	**62,542**

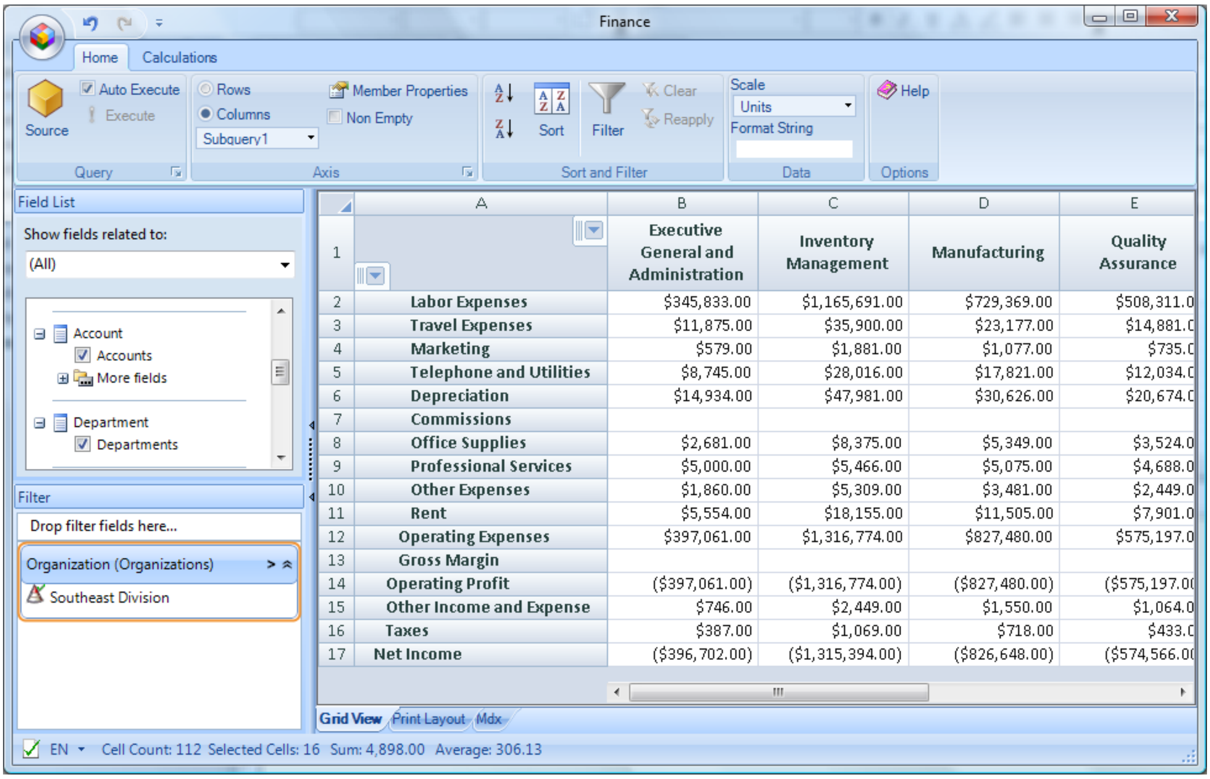
Creating a Formatted Report

Formatted reports allow you to generate attractive, production ready reports in a few mouse clicks without having to spend significant amounts of time hooking up Reporting Services table attributes.

To create a formatted report click on the “Add Formatted Report” option which is available from the context menu when you right mouse click on the Reports folder in your Visual Studio project.



The Investigator window will be launched in which you can build your query:

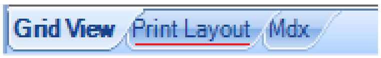


	A	B	C	D	E
1		Executive General and Administration	Inventory Management	Manufacturing	Quality Assurance
2	Labor Expenses	$345,833.00	$1,165,691.00	$729,369.00	$508,311.0
3	Travel Expenses	$11,875.00	$35,900.00	$23,177.00	$14,881.0
4	Marketing	$579.00	$1,881.00	$1,077.00	$735.0
5	Telephone and Utilities	$8,745.00	$28,016.00	$17,821.00	$12,034.0
6	Depreciation	$14,934.00	$47,981.00	$30,626.00	$20,674.0
7	Commissions				
8	Office Supplies	$2,681.00	$8,375.00	$5,349.00	$3,524.0
9	Professional Services	$5,000.00	$5,466.00	$5,075.00	$4,688.0
10	Other Expenses	$1,860.00	$5,309.00	$3,481.00	$2,449.0
11	Rent	$5,554.00	$18,155.00	$11,505.00	$7,901.0
12	Operating Expenses	$397,061.00	$1,316,774.00	$827,480.00	$575,197.0
13	Gross Margin				
14	Operating Profit	($397,061.00)	($1,316,774.00)	($827,480.00)	($575,197.00
15	Other Income and Expense	$746.00	$2,449.00	$1,550.00	$1,064.0
16	Taxes	$387.00	$1,069.00	$718.00	$433.0
17	Net Income	($396,702.00)	($1,315,394.00)	($826,648.00)	($574,566.00

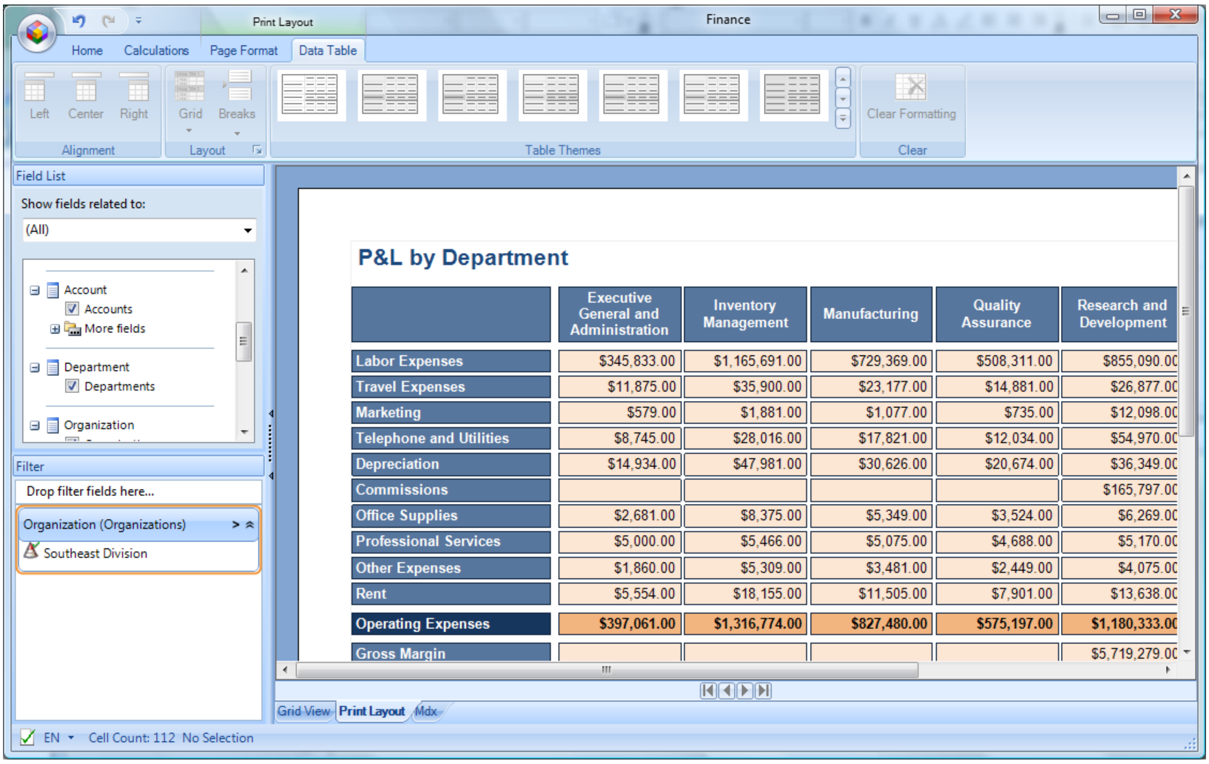
Cell Count: 112 Selected Cells: 16 Sum: 4,898.00 Average: 306.13

You can add parameters to queries in the same way as for standard reports (see previous section). Note that in formatted reports the parameter filter list is used to create the parameter dataset which controls the list of values displayed to the user in the finished report.

When you are happy with your query you should switch to “Print Layout” mode by clicking on the tab at the bottom of the query window.



You can use the Print Layout features to format your report within a few mouse clicks (please refer to the online help for full details of how to use the print layout features in the Investigator):



P&L by Department

	Executive General and Administration	Inventory Management	Manufacturing	Quality Assurance	Research and Development
Labor Expenses	$345,833.00	$1,165,691.00	$729,369.00	$508,311.00	$855,090.00
Travel Expenses	$11,875.00	$35,900.00	$23,177.00	$14,881.00	$26,877.00
Marketing	$579.00	$1,881.00	$1,077.00	$735.00	$12,098.00
Telephone and Utilities	$8,745.00	$28,016.00	$17,821.00	$12,034.00	$54,970.00
Depreciation	$14,934.00	$47,981.00	$30,626.00	$20,674.00	$36,349.00
Commissions					$165,797.00
Office Supplies	$2,681.00	$8,375.00	$5,349.00	$3,524.00	$6,269.00
Professional Services	$5,000.00	$5,466.00	$5,075.00	$4,688.00	$5,170.00
Other Expenses	$1,860.00	$5,309.00	$3,481.00	$2,449.00	$4,075.00
Rent	$5,554.00	$18,155.00	$11,505.00	$7,901.00	$13,638.00
Operating Expenses	**$397,061.00**	**$1,316,774.00**	**$827,480.00**	**$575,197.00**	**$1,180,333.00**
Gross Margin					$5,719,279.00

When you are happy with your report appearance you should exit from the investigator and you will see that an rdl file name “New Report.rdl” has been added to your project; rename this report as required.

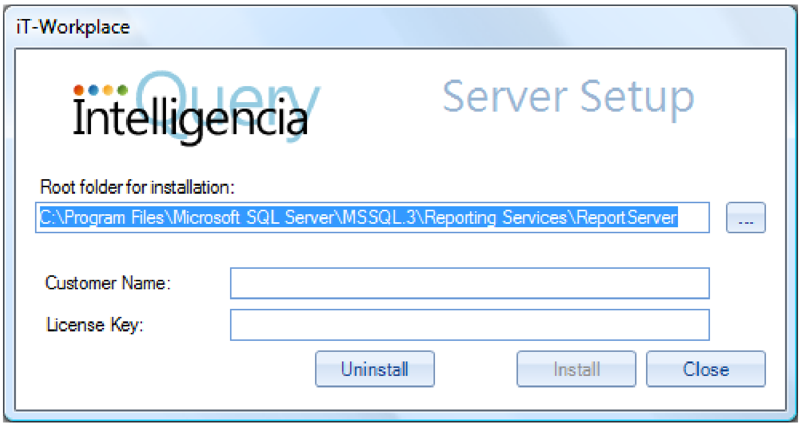
You can modify formatted reports using the Visual Studio report designer tools to add company logos and other report items if required.

Deploying to the Report Server

To deploy IQ to your report server instance you will need to locate and run the deployment tool IQServerInstall.exe. This is located in the installation root for IQ normally located at:

C:\Program Files\iT-Workplace Limited\Intelligencia Query (3.x.x)\IQServerInstall.exe

When you run this tool the following window will be displayed:



Root folder for installation: C:\Program Files\Microsoft SQL Server\MSSQL.3\Reporting Services\ReportServer

Make sure that the "Root folder for installation" points to the reporting services instance that you wish to update. This can be located on a different machine so long as you have full authority to update the folder. Note that the ReportServer folder contains the RSReportServer.config and RSSrvPolicy.config files.

Enter your customer name and license key for "Intelligencia Query Server". The UI will check the license key and will not enable the Install button unless it is correct. Note that there are separate client and server licenses so please make sure you are using the correct key.

For a demonstration installation leave the Customer Name and License Key blank. When running in demonstration mode reports will be rendered but the cell contents will be displayed as #LICENSE!

When you click on "Install" the tool will copy the necessary dlls into the bin folder for the report server and will modify the RSReportServer.config and RSSrvPolicy.config files.

You will need to stop and start the SQL Server Reporting Services windows service to ensure that the config files are re-read.

Manual deployment

If you need to deploy IQ manually you can use the following procedure:

1 – Deploy the IQ dlls

Locate the IQ installation. By default this will be in the folder:

```
C:\Program Files\iT-Workplace Limited\Intelligencia Query for BIDS\
```

Copy the following files to the bin folder of your Reporting Services instance:

DocModel.dll
Interop.ADODB.dll
Interop.ADOMD.dll
ITW.Investigator.Plugin.Api.dll
ITW.IQ.ASPivot.dll
ITW.Reporting.ObjectModel.dll
ITWFormControls.dll
ITWOlapControls.dll

If you are working with SQL 2005 and do NOT have SQL 2008 installed you will also need to copy:

Microsoft.AnalysisServices.dll

Note: Microsoft.AnalysisServices.dll is the version 10 (SQL 2008) file required as a dependency of ITWOlapControls.dll.

Typically the bin folder for Reporting Services will be:

```
C:\Program Files\Microsoft SQL Server\MSSQL.3\Reporting Services\Report Server\bin
```

2 – Modify the server policy configuration

Reporting Services needs to be configured to recognise the IQ ASPivot data extension and dependent dlls as trusted assemblies. This is carried out by modifying the rssrvpolicy.config file located in the report server root folder normally located at:

```
C:\Program Files\Microsoft SQL Server\MSSQL.3\Reporting Services\Report Server
```

Locate the codegroup element "$CodeGen$/*" and add the following:

```
<CodeGroup
        class="UnionCodeGroup"
        version="1"
        PermissionSetName="FullTrust">
        <IMembershipCondition
                class="UrlMembershipCondition"
                version="1"
                Url="$CodeGen$/*"
        />
</CodeGroup>
<CodeGroup
        class="UnionCodeGroup"
        version="1"
        PermissionSetName="FullTrust"
        Name="ASPivot"
        Description="Intelligencia Query - ASPivot Data Extension">
        <IMembershipCondition
                class="UrlMembershipCondition"
                version="1"
                Url="C:\Program Files\Microsoft SQL Server\MSSQL.3\Reporting
        Services\ReportServer\bin\ITW.IQ.ASPivot.dll"
        />
</CodeGroup>
<CodeGroup
        class="UnionCodeGroup"
        version="1"
        PermissionSetName="FullTrust"
        Name="ITWOlapControls"
        Description="Intelligencia Query - ASPivot Data Extension">
        <IMembershipCondition
```

```
                class="UrlMembershipCondition"
                version="1"
                Url="C:\Program Files\Microsoft SQL Server\MSSQL.3\Reporting
        Services\ReportServer\bin\ITWOlapControls.dll"
        />
</CodeGroup>
<CodeGroup>
        class="UnionCodeGroup"
        version="1"
        PermissionSetName="FullTrust"
        Name="DocModel"
        Description="Intelligencia Query - ASPivot Data Extension">
        <IMembershipCondition
                class="UrlMembershipCondition"
                version="1"
                Url="C:\Program Files\Microsoft SQL Server\MSSQL.3\Reporting
        Services\ReportServer\bin\DocModel.dll"
        />
</CodeGroup>
<CodeGroup>
        class="UnionCodeGroup"
        version="1"
        PermissionSetName="FullTrust"
...
```

Modify the path of the Url element to reflect the location of your report server.

3 – Modify the report server configuration

Last we need to enable the Intelligencia Query ASPivot data extension and implement the license. To do this located the configuration file rsreportserver.config which is located in the report server root folder normally located at:

```
C:\Program Files\Microsoft SQL Server\MSSQL.3\Reporting Services\Report Server
```

Locate the data extensions section of the configuration and add the following element:

```
<Data>
        <Extension Name="SQL"
Type="Microsoft.ReportingServices.DataExtensions.SqlConnectionWrapper,Microsoft.ReportingServi
ces.DataExtensions"/>
        <Extension Name="OLEDB"
Type="Microsoft.ReportingServices.DataExtensions.OleDbConnectionWrapper,Microsoft.ReportingSer
vices.DataExtensions"/>
        <Extension Name="OLEDB-MD"
Type="Microsoft.ReportingServices.DataExtensions.AdoMdConnection,Microsoft.ReportingServices.D
ataExtensions"/>
        <Extension Name="ORACLE"
Type="Microsoft.ReportingServices.DataExtensions.OracleClientConnectionWrapper,Microsoft.Repor
tingServices.DataExtensions"/>
        <Extension Name="ODBC"
Type="Microsoft.ReportingServices.DataExtensions.OdbcConnectionWrapper,Microsoft.ReportingServ
ices.DataExtensions"/>
        <Extension Name="XML"
Type="Microsoft.ReportingServices.DataExtensions.XmlDPConnection,Microsoft.ReportingServices.D
ataExtensions"/>
        <Extension Name="SAPBW"
Type="Microsoft.ReportingServices.DataExtensions.SapBw.SapBwConnection,Microsoft.ReportingServ
ices.DataExtensions.SapBw"/>
        <Extension Name="ESSBASE"
Type="Microsoft.ReportingServices.DataExtensions.Essbase.EssbaseConnection,Microsoft.Reporting
Services.DataExtensions.Essbase"/>
        <!-- <Extension Name="SSIS"
Type="Microsoft.SqlServer.Dts.DtsClient.DtsConnection,Microsoft.SqlServer.Dts.DtsClient,
Version=9.0.242.0, Culture=neutral, PublicKeyToken=89845dcd8080cc91"/> -->
        <!-- <Extension Name="SAP"
Type="Microsoft.Adapter.SAP.SAPConnection,Microsoft.Adapter.SAP.SAPProvider, Version=1.0.0.0,
Culture=neutral, PublicKeyToken=31bf3856ad364e35"/> -->
        <Extension Name="IQASPivot" Type="ITW.IQ.ASPivot.ASPivotConnection,ITW.IQ.ASPivot">
                <Configuration>
                        <License>
```

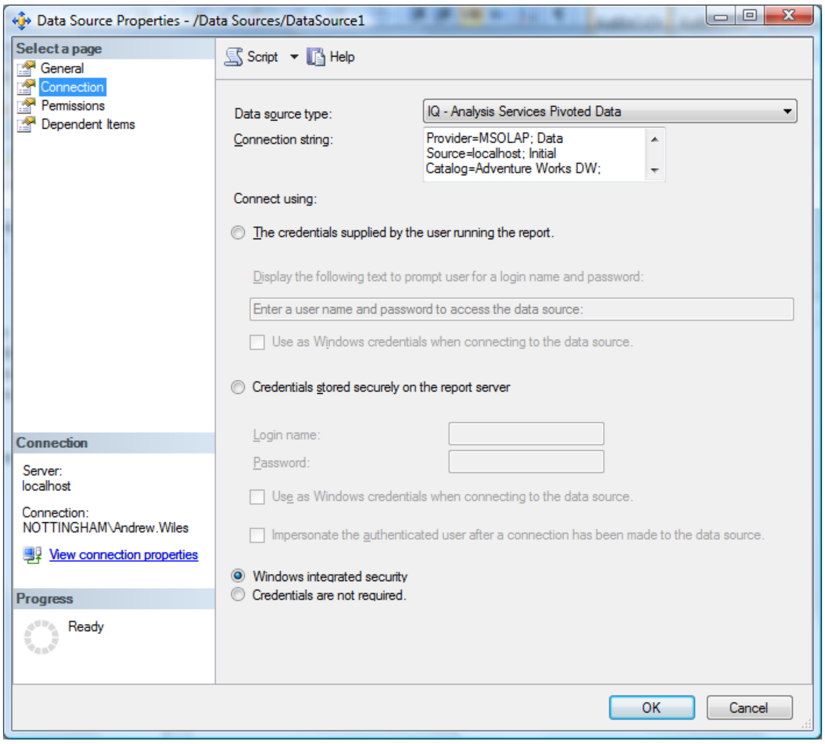
```
                                <CustomerName><<Your Customer Name>></CustomerName>
                                <LicenseKey><<Your License Key>></LicenseKey>
                        </License>
                </Configuration>
        </Extension>
</Data>
```

The customer name and license key will have been provided in your license confirmation email.

Note: If you do not have a valid customer name and license key the implementation will work but data values will be returned as the string “#LICENSE!”

Testing the Configuration

Deploy a test report along with its connection in the normal way. Open SQL Server Management Studio and attach to the report server. Open the properties for the connection and check that the data extension has been correctly registered.



Data Source Properties - /Data Sources/DataSource1

Data source type: IQ - Analysis Services Pivoted Data

Connection string: Provider=MSOLAP; Data Source=localhost; Initial Catalog=Adventure Works DW;

Connect using:

○ The credentials supplied by the user running the report.

Display the following text to prompt user for a login name and password:

Enter a user name and password to access the data source:

☐ Use as Windows credentials when connecting to the data source.

○ Credentials stored securely on the report server

Login 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.

Connection

Server: localhost

Connection: NOTTINGHAM\Andrew.Wiles

If the data source has not registered correctly you will not see the data source type “IQ – Analysis Services Pivoted Data” in the list of available source types.

Modify the connection properties to set the login credentials that are appropriate for your installation.

Lastly open a report and check that it renders correctly.