



X°CELSIUS
ENTERPRISE

TechTips

Connecting Xcelsius Dashboards to External Data Sources using: Web Services (Dynamic Web Query)

A step-by-step guide to connecting Xcelsius Enterprise XE dashboards to company databases using Web Services for enterprise-wide reporting.

Chris Bryant
Technical Consultant

Infommersion, Inc.
January 2005

Table of Contents

	<i>Page</i>
▪ Prerequisites for getting the most from this Guide	3
▪ Introduction	3
▪ Integrating Xcelsius Enterprise XE with your data source	4
▪ How Xcelsius works	4
▪ Basic architecture	4
▪ XML integration dynamic refresh	6
▪ Using Web services (dynamic Web query)	6
○ Configuring Xcelsius Enterprise XE for refreshing data	6
○ Format of the XML file	9
○ Benefits	10
▪ XML “gotchas”	10

Prerequisites for getting the most from this Tech Tips Guide

This Tech Tips Guide is technically oriented, and as such the reader should have a solid understanding of the following technologies and terms:

Excel: A basic understanding of Excel – how to navigate and create formulas – is required. It is beneficial, but not required, to have an understanding of Excel 2003 XML Maps.

Flash: No knowledge of Flash programming is necessary

Xcelsius: Experience creating data presentations with Xcelsius is required, including an understanding of the work area components and how they work.

XML: A basic familiarity with XML – how it is structured and how it can be used for communicating between applications – is required.

Additionally, included in some guides is an example of an ASPX script. Therefore, knowledge of ASP .NET and how to connect to a database through OLEDB would be helpful for understanding this example.

A few Xcelsius-specific terms used throughout the guide:

- **Xcelsius Work Area** – The Xcelsius authoring environment – which includes menus, toolbars, components, the object browser, and the canvas – used to design and create dashboards and data presentations.
- **Xcelsius File** - The XLF file created during the development of Xcelsius data presentations. It is the working (and saved) development file used in the work area (authoring environment).
- **Xcelsius Data Presentation** – The output of the work area (authoring environment) for deployment to presentation/dashboard viewers. An Xcelsius data presentation is a Flash file, also known as a SWF file for its file extension (.swf). It can be exported to, and is fully operational in, PowerPoint, Outlook and the Web.

Introduction

A corporate dashboard is critical for monitoring the daily health of an organization. It gives decision-makers concise, visual access to key performance indicators that drive the business.

With Xcelsius|Enterprise XE, the award-winning software from Infomersion, users create and deploy corporate dashboards that make it easier than ever to identify critical data relationships, probe elaborate “what-if” scenarios, and peer into their company’s financial future.

By combining critical business data with the features of Macromedia Flash™, Xcelsius gives users the power to convert data from ordinary Excel spreadsheets and XML-compliant company databases into dynamic dashboards and data presentations for Portals, PowerPoint and the Web. This guide provides a step-by-step guide to connecting Xcelsius|Enterprise XE data presentations to company databases using Web services (dynamic Web query), for enterprise-wide reporting.

Connecting Xcelsius|Enterprise XE Dashboards to External Data Sources

Now that you've discovered the power and possibilities of building highly sophisticated and interactive data presentations with Xcelsius, let's take a closer look at how to connect a dashboard that was created with Xcelsius|Enterprise XE to external data sources – such as corporate databases – via XML.

How Xcelsius Works

Before looking at the various options for connecting an Xcelsius dashboard to an external data source, it is important to understand how Xcelsius|Enterprise XE works with Excel and why connecting to an external data source can be beneficial.

The first step to creating an Xcelsius data presentation or dashboard is to import a previously created Excel spreadsheet into the Xcelsius authoring environment (work area). This import brings in all data, formulas, formatting, mapping, and other features contained in the Excel spreadsheet, and loads it into Xcelsius' own embedded Excel spreadsheet. All the information is still contained in Excel, but now it is in an Excel spreadsheet embedded in the Xcelsius work area (XLF file). You will see this when you select a data range for use with a component. The Excel spreadsheet which was imported is opened in a temporary Excel file. Therefore, after the initial import, there is no need for the original Excel spreadsheet.

After you have created your data presentation in the Xcelsius work area, Xcelsius converts it into a Flash (swf) file. The finished presentation (Flash file) does not contain the full Excel spreadsheet as the Xcelsius work area does. It only contains the range of cells, and their dependencies, that you selected for each component. When a viewer opens the presentation (Flash file), all the necessary data is either contained within the presentation itself, or dynamically retrieved via connectivity to an external data source.

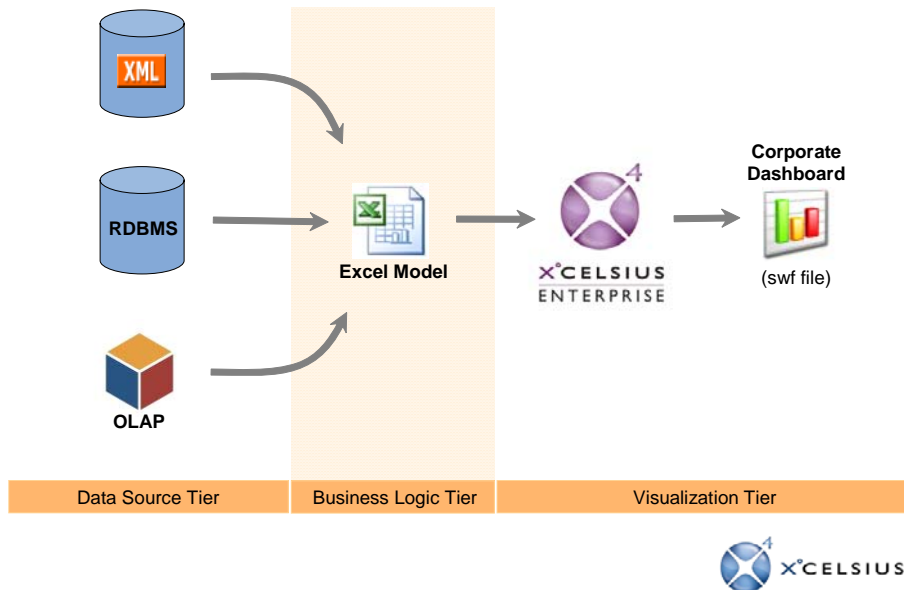
By connecting your data presentation (such as a dashboard) to an external data source, you can cause the data within the presentation to dynamically refresh – thus enabling the dashboard to be continuously connected to the latest data available.

Basic Architecture

Let's take a look at the typical architecture of how and where Xcelsius|Enterprise XE fits into a dashboard reporting environment. Following are two diagrams: design architecture and deployment architecture. The first diagram illustrates the process and technologies used to build the Xcelsius data presentation and generate a Flash file. The second diagram illustrates where the Flash file resides within the dashboard application when it's deployed.

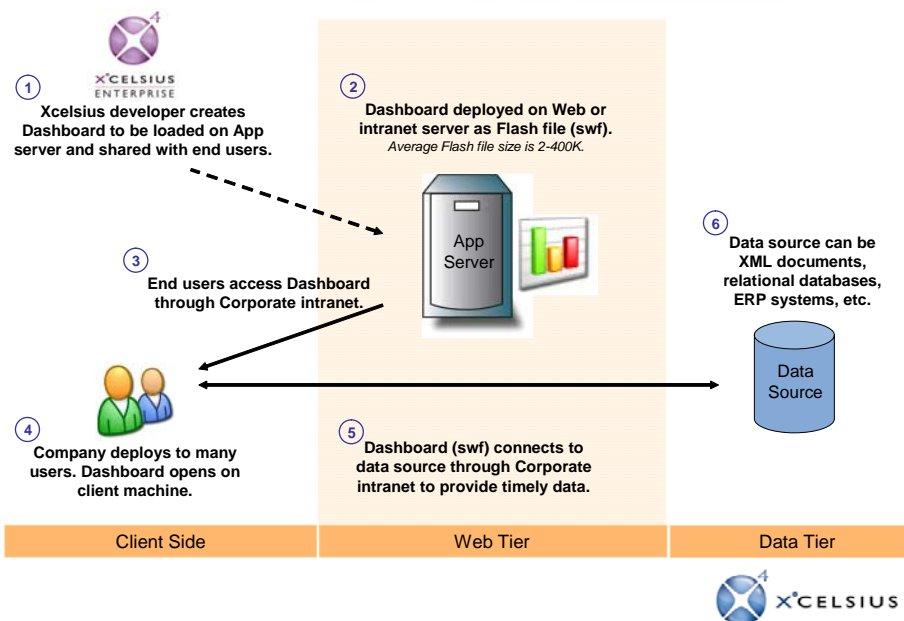
During the design phase, the data can be brought into Excel by many methods: relational pull, manual entry, importing XML, etc. Once the data is in Excel and subsequently imported into Xcelsius, there is no need for the database anymore during the design phase. All data is now self-contained within the data presentation, and the resulting Flash file can be used in static data situations.

Xcelsius Corporate Dashboard: Development



After the completed dashboard has been deployed, connectivity to the back-end data source is required to dynamically refresh the data within the Flash file. This is done independently of the original Excel spreadsheet. If there were links to the data sources created when the data was imported into Excel, those links will be passed along to the Xcelsius dashboard (Flash file), which will then be able to dynamically refresh the data through those links, this is the Excel 2003 XML Maps option. Also, new links to different data sources can be setup within Xcelsius. These options to dynamically refresh data within a data presentation or dashboard will be discussed in the remaining sections of this document.

Xcelsius Corporate Dashboard: Deployment



XML Integration Dynamic Refresh

Now, let's take a closer look at how to connect your Xcelsius|Enterprise XE (Xcelsius) dashboard to external data sources, such as company databases.

When an Xcelsius data presentation is initially created, the data and formulas from the Excel spreadsheet are stored within the Xcelsius work area, and embedded in the presentation (Flash file). So what happens when you want the data within the Xcelsius presentation to be refreshed dynamically? Xcelsius allows you to link data presentations to an XML data source in order to retrieve updated data. This can be accomplished by three different methods: 1) XML Maps in Excel 2003, 2) Web Services (Dynamic Web Query), and 3) the XML Data button.



This Tech Tips Guide focuses on using Web Services (Dynamic Web Query).

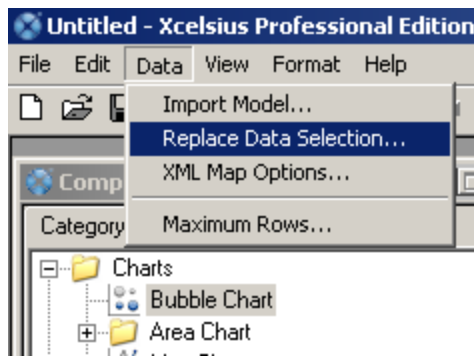
Note: This guide does not describe best practices for organizing your data in Excel, but is intended to help you understand various options for connecting your Xcelsius dashboard to an external data source. For more information on designing your spreadsheets for optimum use with Xcelsius, see "Xcelsius Getting-Started Tutorial: Designing Your Spreadsheets" in the online Xcelsius Learning Center.

Web Services (Dynamic Web Query)

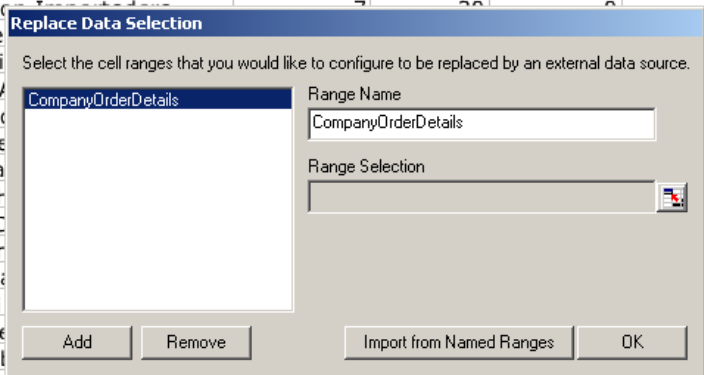
For users who haven't upgraded to Excel 2003 or can create an XML file with any structure, there is another option for dynamically updating the data in an Xcelsius data presentation. This option allows you to specify the range of cells in the Excel spreadsheet you want to refresh, and from which XML file the refresh will occur. The structure of the XML file is very specific (unlike Excel 2003 XML Maps, which allows any formatting because Excel handles the mapping). The structure is a generic Row and Column formatted XML. When you select a range of cells to replace with an XML data source, you will notice the range of cells is purely Rows and Columns of data. The structure of the XML file needs to have the same structure, so the mapping occurs correctly.

Configure Xcelsius for Refreshing Data

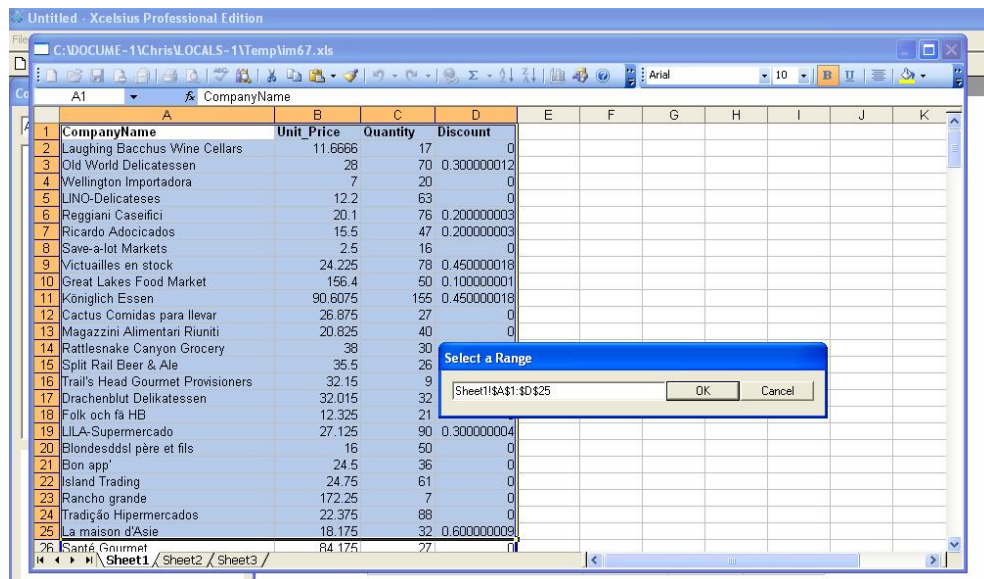
1. For this example, you will need to have some data already saved in an Excel spreadsheet.
2. Open Xcelsius, and import your spreadsheet by clicking on the Excel icon  on the toolbar and then browsing to your desired Excel file. Click Open, then OK.
3. Click the Table component in the Components panel, and add the table to the blank slide (canvas) by dragging and dropping it anywhere on the canvas. Double Click the table to open its Properties panel and define the data to be displayed. Use the selector icon  next to the Display Data input text box to select a range of cells to show in the table. Notice the table on the canvas has updated to show the range of data you selected.
4. The next step is to define which data you want to replace with the data in the XML file. Select Replace Data Selection from the Data menu. The Replace Data Selection dialog box will appear.



Laughina Bacchus Wine Cellars	100	17	0
Old World Delicatessen	28	70	0.3
Wellington Importadora	7	20	0
LINO-Delicatesses	12.2	63	0
Reggiani Caseifici	20.1	76	0.200000003
Ricardo Adocicados	15.5	47	0.200000003
Save-a-lot Markets	2.5	16	0
Victuailles en stock	24.225	78	0.450000018
Great Lakes Food Market	156.4	50	0.100000001
Königlich Essen	90.6075	155	0.450000018
Cactus Comidas para llevar	26.875	27	0
Magazzini Alimentari Riuniti	20.825	40	0
Rattlesnake Canyon Grocery	38	30	0
Split Rail Beer & Ale	35.5	26	0
Trail's Head Gourmet Provisioners	32.15	9	0
Drachenblut Delikatessen	32.015	32	0
Folk och få HB	12.325	21	0
LILA-Supermercado	27.125	90	0.300000004
Blondesdsl père et fils	16	50	0



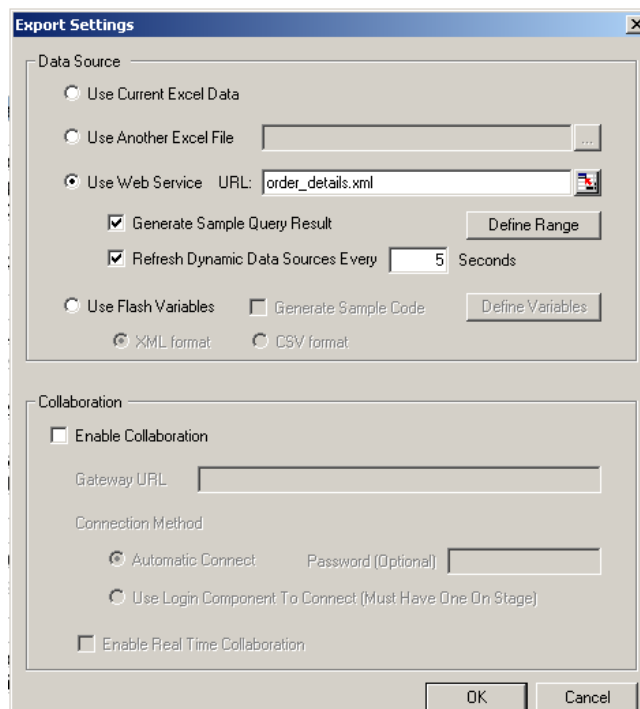
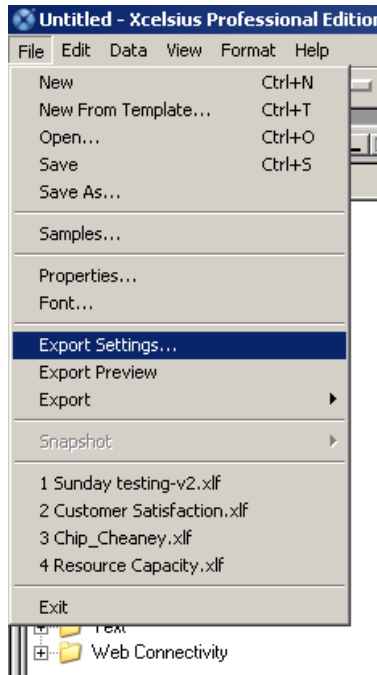
- Click the Add button to define a new Range. Make the range name as descriptive as possible. We named our range “CompanyOrderDetails”. The name is important, it matches the range you want to replace with the data in the XML file. Next click the selector icon next to Range Selection to define the cells in which you want to replace the data. Select the desired cells from the Excel spreadsheet that opens, and click OK.



- Now that we have defined the range of cells that we want replaced, we need to define where the XML file is located, and how often we would like it to refresh. To do this

Select the Export Settings option under the File menu. This will open the Export settings dialog box. Currently the Data Source option is set to Use Current Excel Data. Change this to the Use Web Service option. You will notice that this option is expecting a URL as the source. The URL that we will specify will be the name of our XML file. We are using Order_Details.xml. You don't need the path for this example as we are going to generate an XML file to use. When building your own data presentations, this URL can be any path the computer has access to, or it can be a URL that connects to an XML file on another Web server.

7. Next, select (check) the Generate Sample Query Result and the Refresh Dynamic Data Sources Every check boxes. Set the refresh interval to 5 seconds, and Click OK.



8. We have defined the data range to be replaced, the source of the replacement data, and the refresh frequency. Now it is time to generate the Flash file and test the example. Select the File > Export > Macromedia (SWF) option. Name the SWF file the same name you gave to the XML file. (Note: It is only for this example that we use the same name, normally the XML file can be named any compatible filename.)
9. When the SWF file was created, Xcelsius also created an XML file with the same name. The XML file is located in the same directory as the SWF file. If you open up the XML file you will see a basic row and column format (just below this example is more detail to the format of the XML file). Open the Flash file in Internet Explorer, and open the created XML file in notepad. Make a change to one of the data items in the XML file, and save. You will notice that within 5 seconds the data within the SWF file has changed to reflect the update.

Format of the XML file

Below is a range of cells highlighted in Excel:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	CompanyName	Quantity											
2	Laughing Bacchus Wine Cellars	17											
3	Old World Delicatessen	70											
4	Wellington Importadora	20											
5	LINO-Delicatesses	63											
6	Reggiani Caseifici	76											
7	Ricardo Adocicados	47											
8	Save-a-lot Markets	16											
9	Victuailles en stock	78											
10	Great Lakes Food Market	50											
11	Königlich Essen	155											
12	Cactus Comidas para llevar	27											
13	Magazzini Alimentari Riuniti	40											
14	Rattlesnake Canyon Grocery	30											
15	Split Rail Beer & Ale	26											
16	Trail's Head Gourmet Provisioners	9											
17	Drachenblut Delikatessen	32											
18	Folk och få HB	21											
19	LILA-Supermercado	90											
20	Blondies&dsl père et fils	50											
21	Bon app'	36											
22	Island Trading	61											
23	Rancho grande	7											
24	Tradição Hipermercados	88											

The XML structure for this range of cells should look like this:

```

<data>
<variable name="CompanyOrderDetails">
<row>
<column>CompanyName</column>
<column>Quantity</column>
</row>
<row>
<column>Laughing Bacchus Wine Cellars</column>
<column>17</column>
</row>
<row>
<column>Old World Delicatessen</column>
<column>70</column>
</row>
<row>
<column>Wellington Importadora</column>
<column>20</column>
</row>
<row>

```

```
<column>LINO-Delicateses</column>
<column>63</column>
</row>
<row>
<column>Reggiani Caseifici</column>
<column>76</column>
</row>
</variable>
</data>
```

Notice the simple Row and Column format. The other important feature of this format is the variable tag. The variable name must match to the name of the Range in Xcelsius. We defined our Range to be called CompanyOrderDetails. Notice the variable name in the XML example is set to the same name that was defined for the range. Of course, in practice you won't use Xcelsius to create the XML data source. When the data source is created it must have the format above and specifically the exact name of the Variable range, and must use the tags row and column for all entries. One practice that works very well is to create a XSLT (Style Sheet) that will transform the data from an XML format to the one that is necessary for use with Xcelsius.

Benefits

- Can replace a small subset of the data using Range Replace.
- Don't have to pull back all the data like the Excel mapping feature.
- Refreshes on an interval instead of relying on the user to push a refresh button.

XML Gotchas

This section will explain other formatting requirements of the XML files:

Data Typing in XML

Xcelsius does not support Data Types within an XML file. It requires all data in XML to be in string format. The only exception to this rule is that date-based data must be in its numeric representation. When pulling data from a database these requirements are usually not an issue, but if another application creates the XML or a user is converting text-based data to a XML, these rules need to be taken into account.

Note: this applies to using the Web Services and the XML Data button connectivity options described in this and other Tech Tip Guides, it does not apply to the Excel 2003 Map option.

Non-Table-Based XML in Excel 2003 Maps

If the data being pulled into Excel Maps is in a non-table-based format, the entire root level of the XML file should be pulled into the spreadsheet instead of picking individual elements. A non-table-based format would be an XML file that lists each customer followed by the Orders/Invoices that belong to the customer. By pulling in the full root level, we allow Excel to keep the data in the structure of the XML file. By pulling elements individually Excel might make some changes so it can display the data the best way it sees fit.

Multiple Range selections in a single XML

For both the Web Services and the XML Data Button connectivity options, it is possible to define multiple ranges of data to be replaced from the same XML file. In this case the structure will be exactly the same, but just repeated within the XML file. Inside the data element there will be two blocks of data each with their own variable names which match the names of the ranges defined in Xcelsius. If there are more ranges than blocks in the XML file, the whole file will not work properly. But, it is possible to have more data in the XML than is needed for the ranges.

----- END -----