Working with Structured Data in Microsoft Office SharePoint Server 2007 (Part 4): SharePoint Designer

Applies to: Microsoft Office SharePoint Server 2007, Microsoft Office SharePoint Designer 2007

Explore different options you have to work with structured data in a high volume while you need to perform complex queries and actions against such data ranging from authoring, approval and landing information on Web Part pages, all the way down to physical storage. This blog post is part 4 of a blog post series that I am planning to write on this topic (31 printed pages).

Reza Alirezaei , Microsoft Office Server 2007 MVP
5/13/2009

* For comments please see http://blogs.devhorizon.com/reza/?p=972

Content:

Introduction

Office SharePoint Designer 2007 allows you to smoothly integrate your information into a SharePoint site by giving you the ability to combine HTML design elements with the XSLT data transformation, all in a single design interface. It also offers an amazing support for the Data Form Web Part (a.k.a DFWP) that can be utilized to build rich, powerful forms much easier than ever.

Data Form Web Parts can take advantage of sending additional input parameters either in query string or Web Part connections. This is extremely helpful to drive dynamic changes in the content that is surfaced on your SharePoint pages and experience an interactive interface using a composite no-code solution.

Important
In WSS 2.0, the Data Form Web Part was known as the Data View Web Part (DVWP) or Data View. Data view is a representation of data from a data source. In WSS 3.0 , this Web Part was enhanced to not only allow you to view data , but provide you with the ability to insert, update and delete data from a variety of data sources, such as SharePoint data sources , database connections, Web services , RSS feeds and other type of XML files.

Create SQL data views

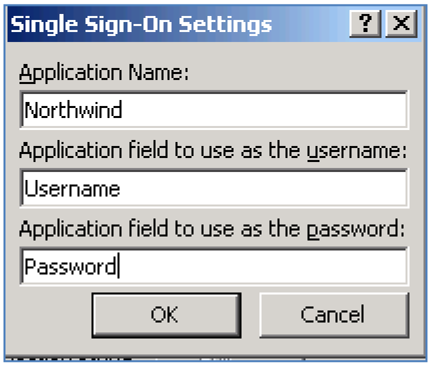
In this section, you will learn how to create a Data Form Web Part that is connected to Microsoft SQL Server 2005 data. You will be also shown the flexibility of the Data Form Web Part to connect to multiple data repositories, display and aggregate such data consistently. Using the Northwind database (See Part1) is simple to begin the process of customizing a Dashboard view using DFWP. Data is surfaced real-time from the Northwind database to show how the Data View presentation will look as it is being designed.

1. Start SharePoint Designer.
2. Select the **OpenSite** option in the **File** menu.

3. Enter **http://litwaredem** in the **Site name** text box.
4. Click in the **File** menu , **New** and then from **Page** tab choose **ASPX**.
5. Save untitled.aspx as **Products.aspx.**
6. Select the **Insert Data View** option from the **Data View** menu.
7. Expand **Database Connections** in the **Data Source Library** pane and click **Connect to a database...**
8. Click on the **Configure Database Connection....**

Important
Database connections in DFWP can only use Sql authentication or SSO. Windows authentication works only when the database resides on the same machine as your Windows SharePoint Services site. In this blog post, we will use SSO to connect to the Northwind database.

9. Enter **LITWAREDEMO (**Replace LITWAREDEMO with your SQL Server name**)** in the **Server Name** box.
10. Choose **Microsoft .NET Framework Data Provider for SQL Server** in the **Provider Name** dropdown**.**
11. Under **Authentication**, click **Use Single Sign-On authentication**, and then click **Settings.**
12. In the **Single Sign-On Settings** dialog box, do the following:
    - In the **Application Name** box, type **Northwind.**
    - In the **Application field to use as the user name** box, type Username.
    - In the **Application field to use as the password** box, type Password.
    - Click on the Ok button.



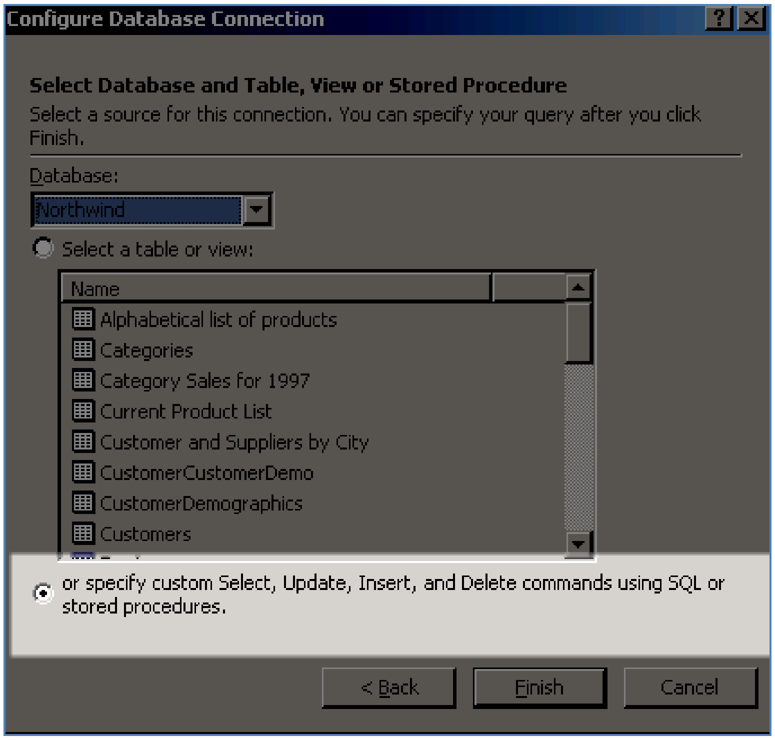
Application Name: Northwind

Application field to use as the username: Username

Application field to use as the password: Password

13. Click Next.
14. Choose **Northwind** from the **Database** dropdown box.
15. From **select a table or view,** click on **Products** and then click on the **Finish** button.

Note that in DFWP, Instead of working with tables or view directly, you could also specify a SQL statement or stored procedure in Configure **Database Connection** dialog box.



Configure Database Connection

Select Database and Table, View or Stored Procedure

Select a source for this connection. You can specify your query after you click Finish.

Database: Northwind

○ Select a table or view:

Name
Alphabetical list of products
Categories
Category Sales for 1997
Current Product List
Customer and Suppliers by City
CustomerCustomerDemo
CustomerDemographics
Customers

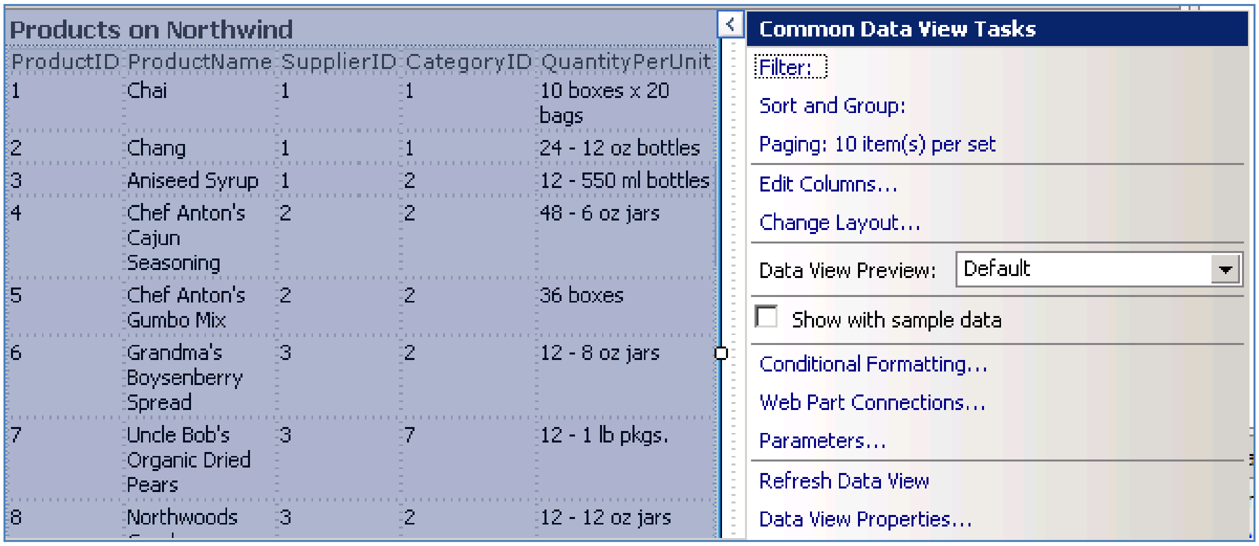
● or specify custom Select, Update, Insert, and Delete commands using SQL or stored procedures.

16. Click on the **OK** button to close **Data Source Properties** dialog box.

Add formula and conditional formatting

In previous section , we successfully created the Products view and now we want to surface it onto a page , add a Formula Column to show the total inventory value (UnitPrice * UnitsInStock) and a conditional formatting that turns a Product row into red for low inventory.

1. Expand **Database Connections** in the **Data Source Library** pane.
2. Click on **Products on Northwind** and select the **Show Data** option.
3. Drag the selected fields in to the Data View on the design surface.
4. SharePoint designer inserts **DataFormWebPart** onto the page to contain Northwind products.
   Right-click the **CategoryID** and select **Delete → Delete Columns.**
   Change **ProductID** text to **ID.**
5. Change **ProductName** text to **Product Name.**
6. Change **ProductName** text to **Supplier Info.**
7. Change **QuantityPerUnit** text to **Quantity Per Unit.**
8. Click on the fly out button for the Data View Web Part.



Products on Northwind

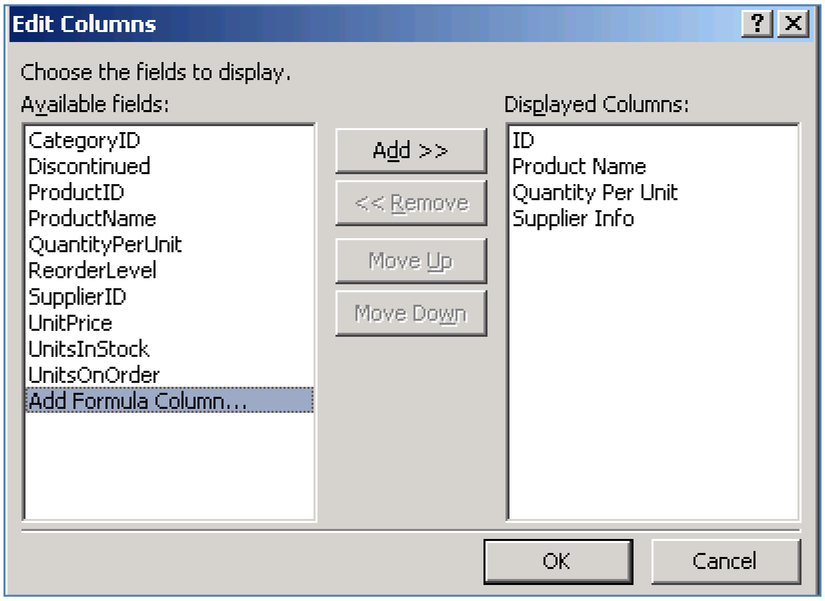
ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit
1	Chai	1	1	10 boxes x 20 bags
2	Chang	1	1	24 - 12 oz bottles
3	Aniseed Syrup	1	2	12 - 550 ml bottles
4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars
5	Chef Anton's Gumbo Mix	2	2	36 boxes
6	Grandma's Boysenberry Spread	3	2	12 - 8 oz jars
7	Uncle Bob's Organic Dried Pears	3	7	12 - 1 lb pkgs.
8	Northwoods	3	2	12 - 12 oz jars

Paging: 10 item(s) per set

Data View Preview: Default

☐ Show with sample data

9. Click on the **Paging** link and select the **Limit total number of items displayed to** option.
10. Set the limit to **5.**
11. Click the **Ok** button.
12. Click on the **Edit Columns...., ** then from **Displayed Columns** choose **Supplier Info.**
13. Click on **Move Down** .
14. Select **Add Formula Column…**
15. click **Add >>** to launch the XPath Expression Builder.



Edit Columns

Choose the fields to display.

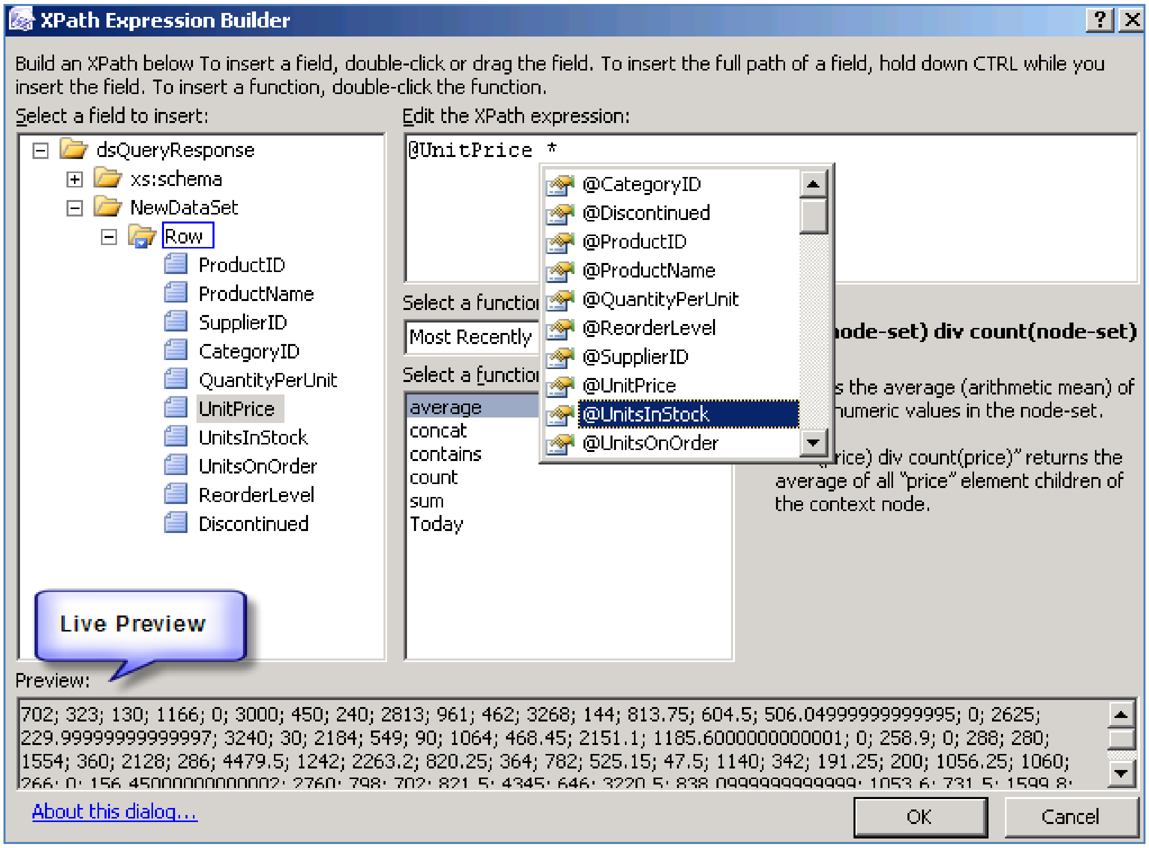
Available fields:

CategoryID
Discontinued
ProductID
ProductName
QuantityPerUnit
ReorderLevel
SupplierID
UnitPrice
UnitsInStock
UnitsOnOrder
Add Formula Column...

Displayed Columns:

ID
Product Name
Quantity Per Unit
Supplier Info

16. Double click or drag and drop **UnitsPrice** on the data tree.
17. Click to the right of **UnitPrice** in the Edit XPath Expression. Type “*” and then use the data aware IntelliSense to select **UnitsInStock**. Notice the live preview area towards the bottom of the dialog.



Build an XPath below To insert a field, double-click or drag the field. To insert the full path of a field, hold down CTRL while you insert the field. To insert a function, double-click the function.

Edit the XPath expression: `@UnitPrice *`

node-set) div count(node-set)

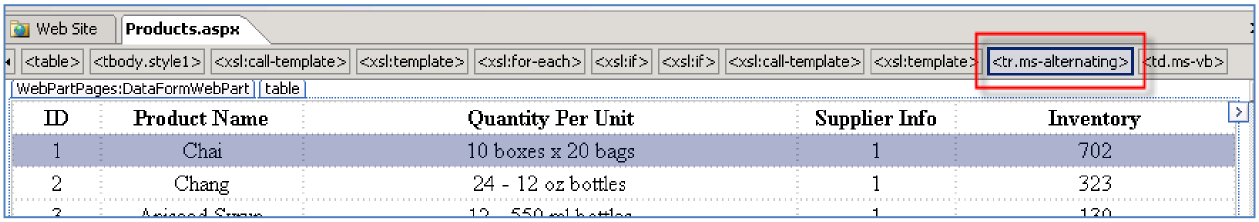
s the average (arithmetic mean) of numeric values in the node-set.

rice) div count(price)" returns the average of all "price" element children of the context node.

Preview:

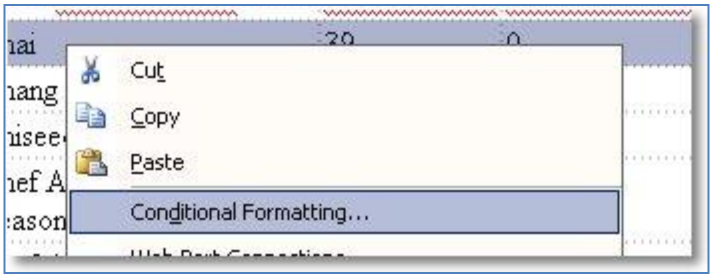
```
702; 323; 130; 1166; 0; 3000; 450; 240; 2813; 961; 462; 3268; 144; 813.75; 604.5; 506.04999999999995; 0; 2625;
229.99999999999997; 3240; 30; 2184; 549; 90; 1064; 468.45; 2151.1; 1185.6000000000001; 0; 258.9; 0; 288; 280;
1554; 360; 2128; 286; 4479.5; 1242; 2263.2; 820.25; 364; 782; 525.15; 47.5; 1140; 342; 191.25; 200; 1056.25; 1060;
266; 0; 156.45000000000002; 2760; 798; 702; 821.5; 4345; 646; 3220.5; 838.0999999999999; 1053.6; 731.5; 1599.8;
```

18. Click **OK** in the XPath Expression Builder dialog.
19. Click on the **Supplier Info** and click the **Move Down** button.
20. Click **OK** Edit Columns dialog.
21. Switch to "Split View" by clicking the **Split** tab at the lower left of the design surface. In the "Design View" half, highlight the text that says "**UnitPrice * UnitsInStock**". Notice the granular selection within the XSLT in "Code View". Delete this text and type the words "Inventory Value."
22. Return to "Design View" by clicking the **Design** tab at the lower left of the design surface.
23. Add conditional formatting to highlight low inventory .
24. Click next to one of the data values in the first row of your view. Using the Quick Tag Selector, select the whole row by clicking the TR button.

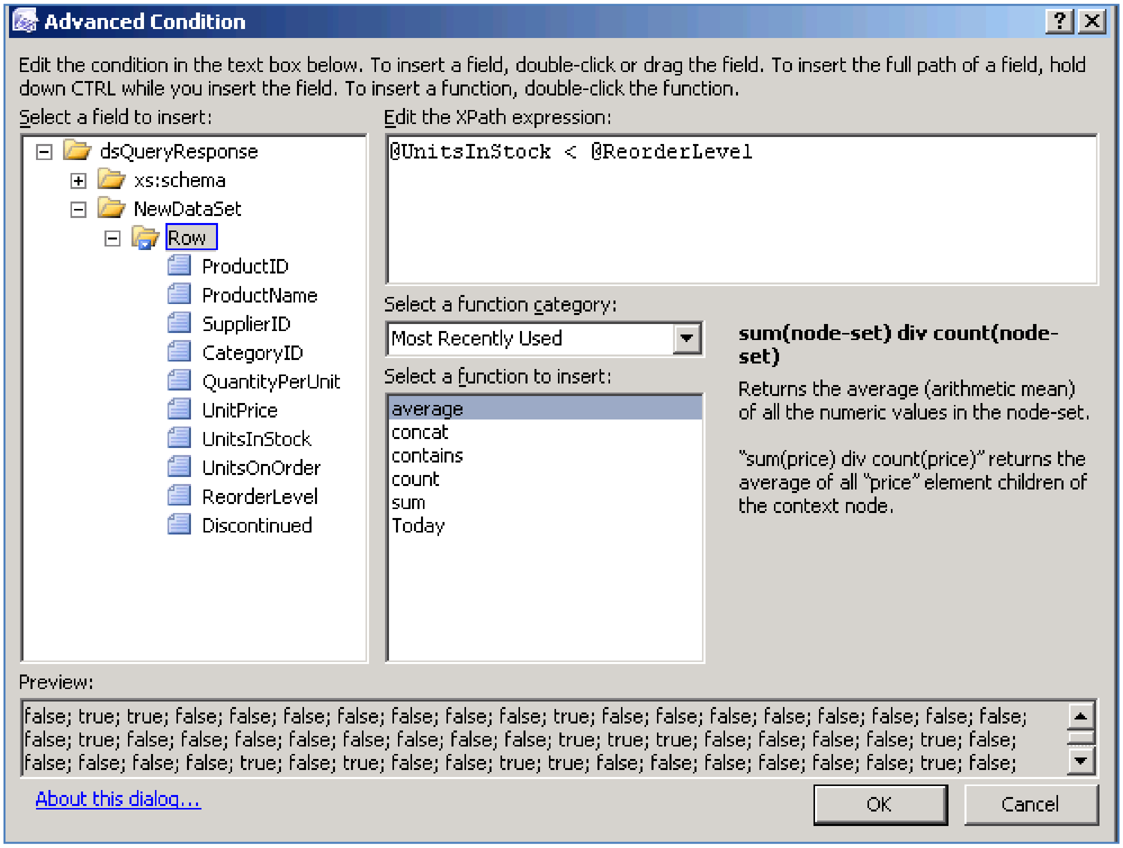

Web Site
Products.aspx
<table> <tbody.style1> <xsl:call-template> <xsl:template> <xsl:for-each> <xsl:if> <xsl:if> <xsl:call-template> <xsl:template> <tr.ms-alternating> <td.ms-vb>
WebPartPages:DataFormWebPart | table

ID	Product Name	Quantity Per Unit	Supplier Info	Inventory
1	Chai	10 boxes x 20 bags	1	702
2	Chang	24 - 12 oz bottles	1	323
3	Aniseed Syrup	12 - 550 ml bottles	1	130

25. Right click on the highlighted row and select **Conditional Formatting…**



26. In the Conditional Formatting task pane, click **Create** and select **Apply Formatting.**
27. Click on the **Click here to add a new clause…** area to open the **Condition Criteria** dialog.
28. Click on the **Advanced…** button to open the **Advanced Condition** dialog.
29. In the **Edit the XPath expression** section type: UnitsInStock < ReorderLevel.



Edit the condition in the text box below. To insert a field, double-click or drag the field. To insert the full path of a field, hold down CTRL while you insert the field. To insert a function, double-click the function.

Edit the XPath expression:

```
@UnitsInStock < @ReorderLevel
```

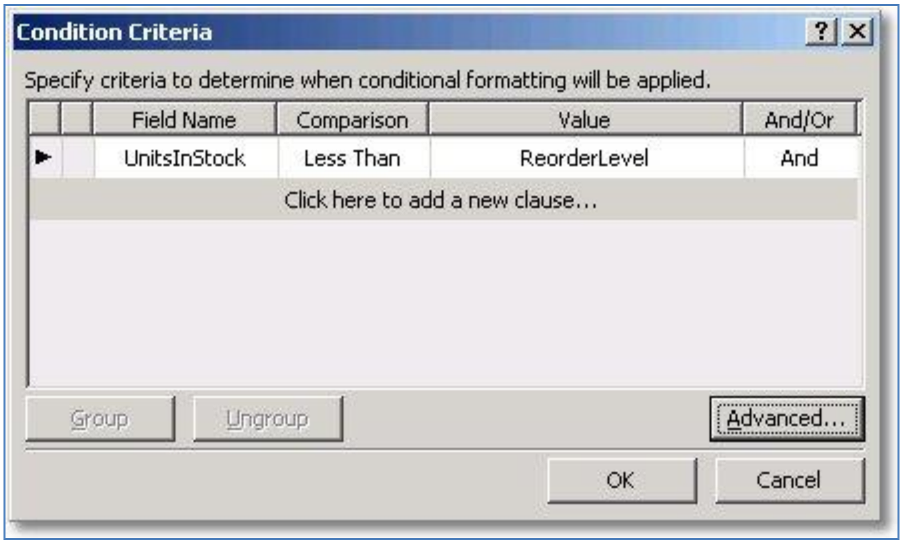
sum(node-set) div count(node-set)

Returns the average (arithmetic mean) of all the numeric values in the node-set.

"sum(price) div count(price)" returns the average of all "price" element children of the context node.

```
false; true; true; false; false; false; false; false; false; false; true; false; false; false; false; false; false; false; false;
false; true; false; false; false; false; false; false; false; false; true; true; true; false; false; false; false; true; false;
false; false; false; false; true; false; true; false; false; true; true; false; false; false; false; false; false; true; false;
```

30. Click **OK** in the **Advanced Condition** dialog.

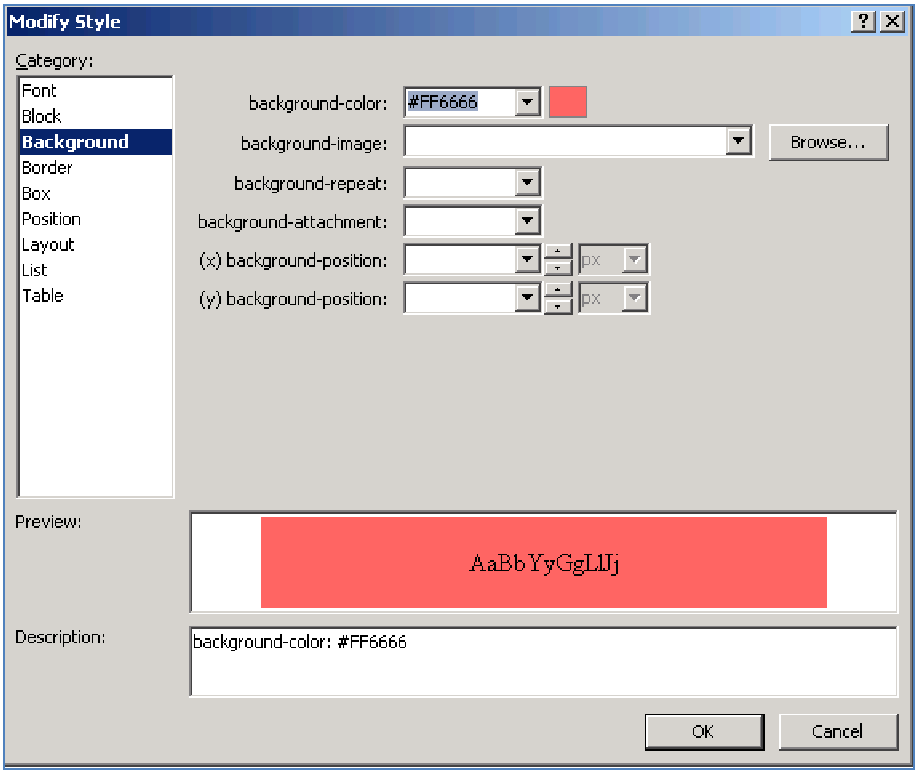


Condition Criteria

Specify criteria to determine when conditional formatting will be applied.

		Field Name	Comparison	Value	And/Or
▶		UnitsInStock	Less Than	ReorderLevel	And

31. Click **OK** in the **Condition Criteria** dialog.
32. The **Modify Style** dialog will appear, in the **Category** list, click **Background.**
33. Set **background-color** to " #FF6666".



Modify Style

background-color: #FF6666 🟥

background-image:

background-repeat:

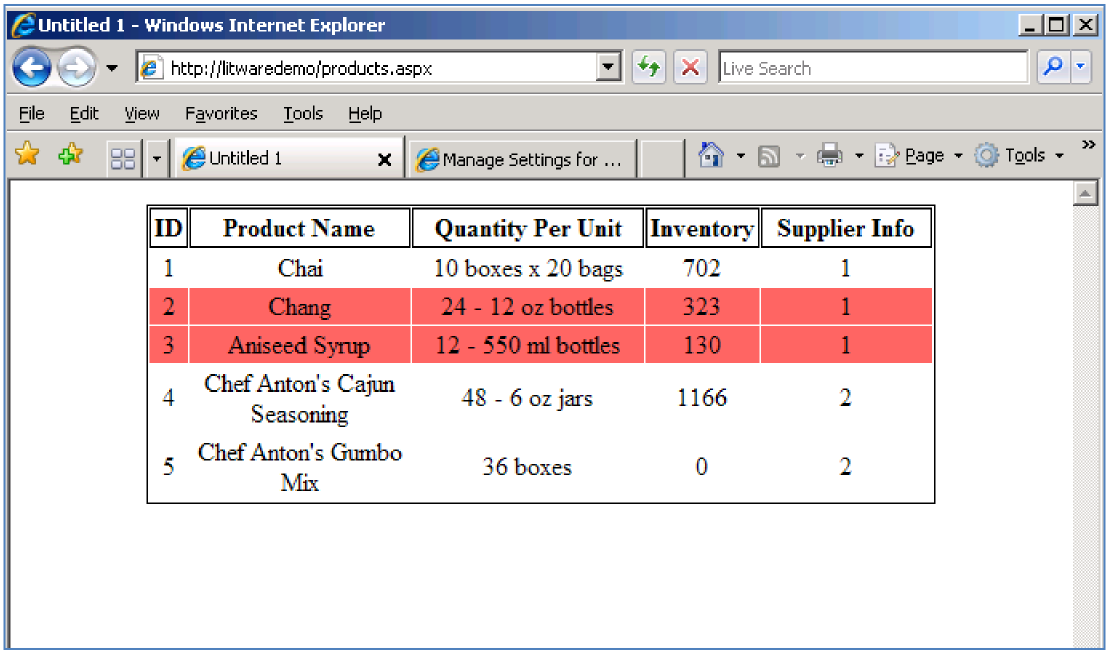
background-attachment:

(x) background-position: px

(y) background-position: px

Description: background-color: #FF6666

34. Click **OK**.
35. Open Internet Explorer and browse to **http://litwaredemo/products.aspx.**

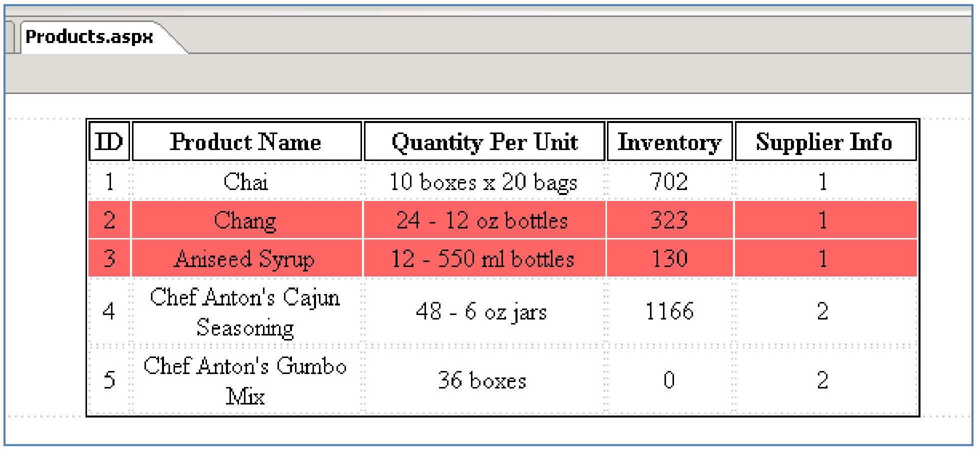

Untitled 1 - Windows Internet Explorer
http://litwaredemo/products.aspx
Live Search
File Edit View Favorites Tools Help
Untitled 1
Manage Settings for ...
Page
Tools

ID	Product Name	Quantity Per Unit	Inventory	Supplier Info
1	Chai	10 boxes x 20 bags	702	1
2	Chang	24 - 12 oz bottles	323	1
3	Aniseed Syrup	12 - 550 ml bottles	130	1
4	Chef Anton's Cajun Seasoning	48 - 6 oz jars	1166	2
5	Chef Anton's Gumbo Mix	36 boxes	0	2

Master/Detail filtering using Web Part connection

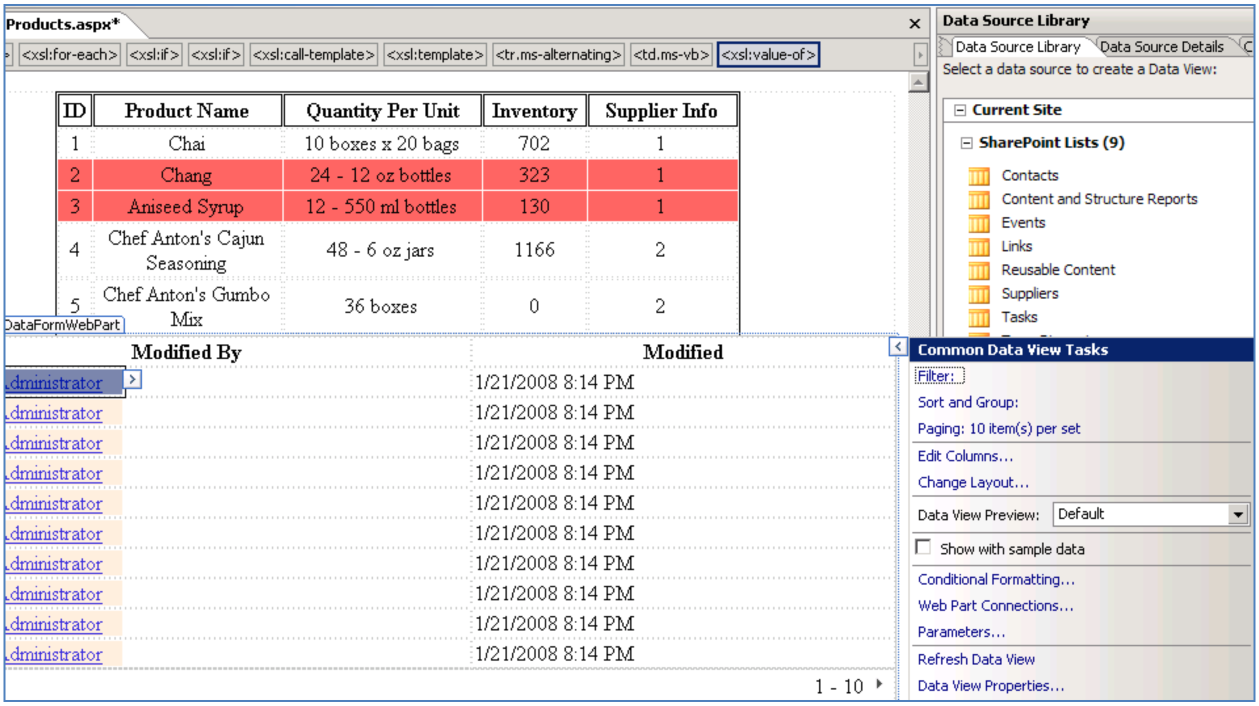
Master/Detail data presentation is a common type of functionality in many applications for implementing one-to-many relationships , such as a report showing all of the products and then allowing users to select a particular product and display its supplier. In this section , we are going to build a master/detail relationship between two Web Parts sitting on the same page.

1. Start SharePoint Designer .
2. Select the **OpenSite** option in the **File** menu.
3. Enter **http://litwaredem** in the **Site name** text box.
4. Click on the **products.aspx** page we created in the previous section.
5. Switch to "Design" by clicking the **Design** tab at the lower left of the design surface.



ID	Product Name	Quantity Per Unit	Inventory	Supplier Info
1	Chai	10 boxes x 20 bags	702	1
2	Chang	24 - 12 oz bottles	323	1
3	Aniseed Syrup	12 - 550 ml bottles	130	1
4	Chef Anton's Cajun Seasoning	48 - 6 oz jars	1166	2
5	Chef Anton's Gumbo Mix	36 boxes	0	2

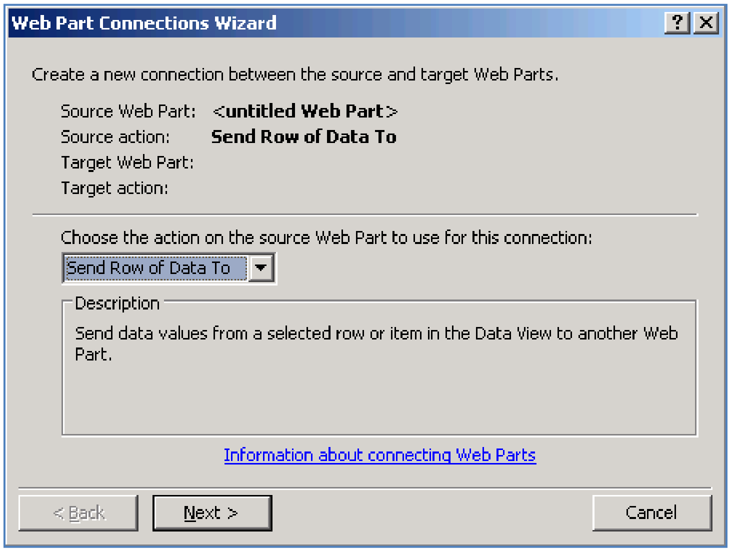
6. Expand **SharePoint Lists** in the **Data Source Library** pane and click on the **Supplier List.**
7. Drag and drop the **Suppliers List** onto the design surface **.**
8. Click on the fly out button for the **Suppliers List** Data Form Web Part.



ID	Product Name	Quantity Per Unit	Inventory	Supplier Info
1	Chai	10 boxes x 20 bags	702	1
2	Chang	24 - 12 oz bottles	323	1
3	Aniseed Syrup	12 - 550 ml bottles	130	1
4	Chef Anton's Cajun Seasoning	48 - 6 oz jars	1166	2
5	Chef Anton's Gumbo Mix	36 boxes	0	2

Modified By	Modified
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM
.dministrator	1/21/2008 8:14 PM

9. Click on the **Edit Columns....,** then from **Displayed Columns** choose **Modified By** and **Modified.**
10. Click on the **Remove** Button.
11. From the **Available fields** list ,choose **SupplierID, CompanyName,ContactName** and **phone.**
12. Click **Ok.**
13. Click on the fly out button for the **Products** Data Form Web Part.
14. Click on the **Web Part Connections.....**
15. Select **Send Row of Data To** from the dropdown box and click **Next .**


Web Part Connections Wizard
? X
Description
Information about connecting Web Parts
< Back
Next >
Cancel

Create a new connection between the source and target Web Parts.

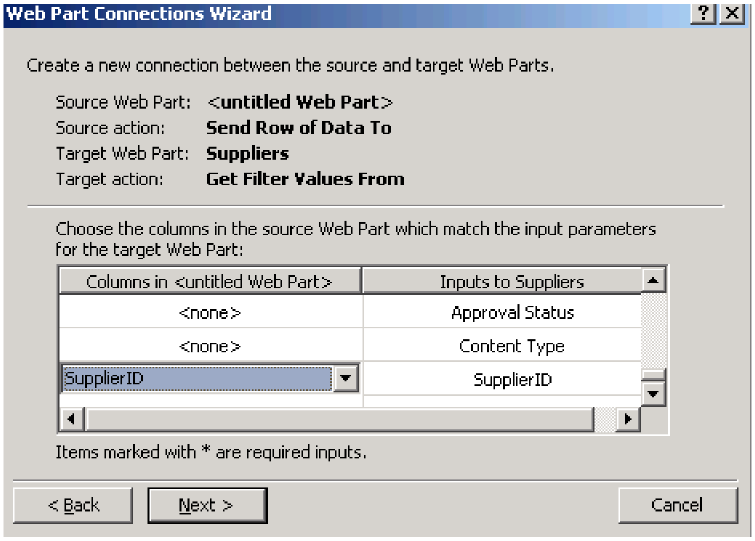
Source Web Part: **<untitled Web Part>**
Source action: **Send Row of Data To**
Target Web Part:
Target action:

Choose the action on the source Web Part to use for this connection:
Send Row of Data To

Send data values from a selected row or item in the Data View to another Web Part.

16. Select **Connect to a Web Part Page on this page.** Click **Next .**
17. **Target Web Part** must be **Suppliers** (because there is no other Web Part to connect to).

18. Leave target action intact **(Get Filters Values From).** Click **Next .**
19. From the **Inputs to Suppliers** columns select **Supplier ID** and map to **SupplierID** in the **Columns in <Untitled Web Part>.** Click **Next.**



Web Part Connections Wizard

Create a new connection between the source and target Web Parts.

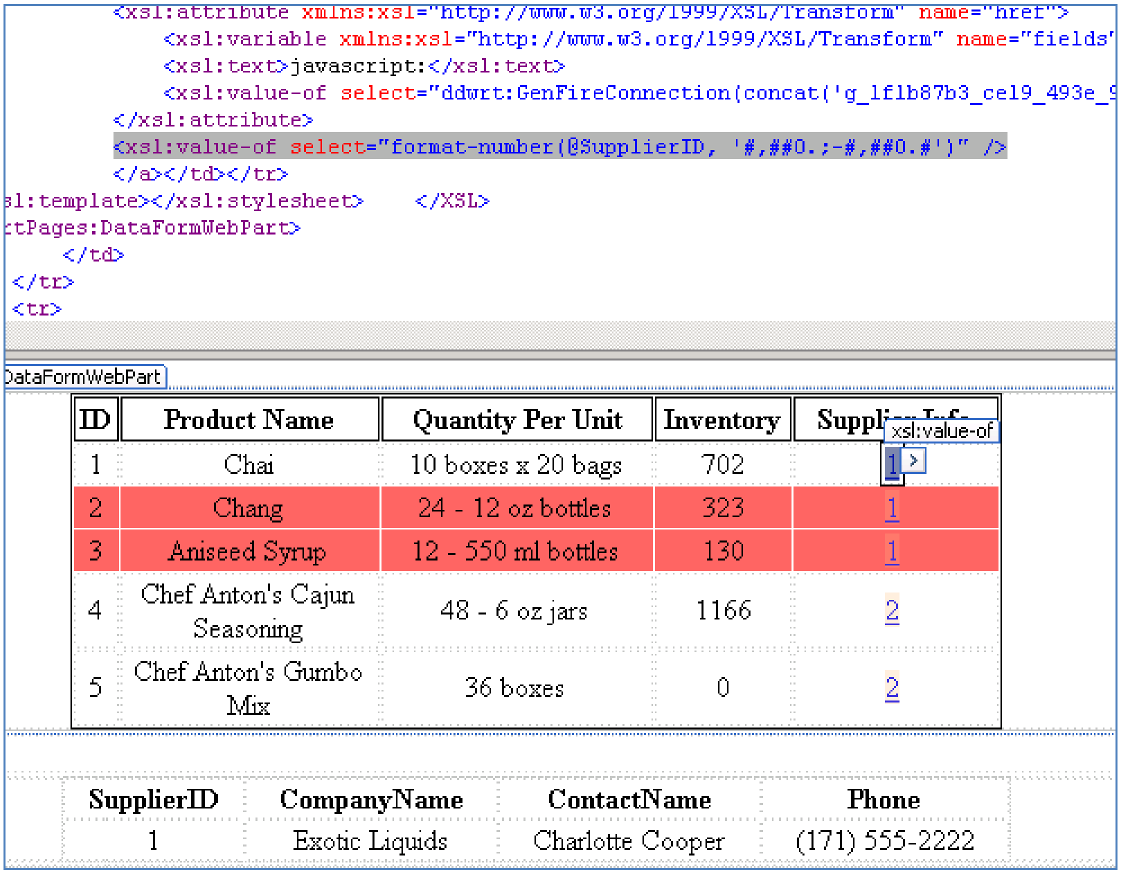
Source Web Part: **<untitled Web Part>**
Source action: **Send Row of Data To**
Target Web Part: **Suppliers**
Target action: **Get Filter Values From**

Choose the columns in the source Web Part which match the input parameters for the target Web Part:

Columns in <untitled Web Part>	Inputs to Suppliers
<none>	Approval Status
<none>	Content Type
SupplierID	SupplierID

Items marked with * are required inputs.

20. From **Create a hyperlink on** dropdown box **,** select **SupplierID** and click **Next.**
21. Click on the **Finish** button.
22. Switch to "Split View" by clicking the **Split** tab at the lower left of the design surface. In the “Design View” half, highlight the one of the columns below **Supplier Info** heading.

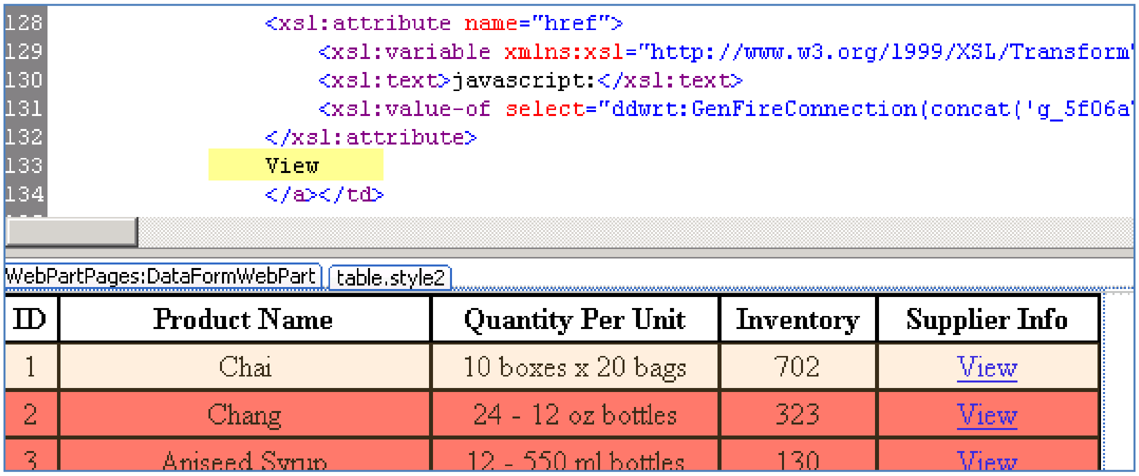


```
          <xsl:attribute xmlns:xsl="http://www.w3.org/1999/XSL/Transform" name="href">
              <xsl:variable xmlns:xsl="http://www.w3.org/1999/XSL/Transform" name="fields'
              <xsl:text>javascript:</xsl:text>
              <xsl:value-of select="ddwrt:GenFireConnection(concat('g_1f1b87b3_ce19_493e_9
          </xsl:attribute>
          <xsl:value-of select="format-number(@SupplierID, '#,##0.;-#,##0.#')" />
          </a></td></tr>
sl:template></xsl:stylesheet>    </XSL>
tPages:DataFormWebPart>
      </td>
 </tr>
 <tr>
```

ID	Product Name	Quantity Per Unit	Inventory	Supplier Info
1	Chai	10 boxes x 20 bags	702	1
2	Chang	24 - 12 oz bottles	323	1
3	Aniseed Syrup	12 - 550 ml bottles	130	1
4	Chef Anton's Cajun Seasoning	48 - 6 oz jars	1166	2
5	Chef Anton's Gumbo Mix	36 boxes	0	2

SupplierID	CompanyName	ContactName	Phone
1	Exotic Liquids	Charlotte Cooper	(171) 555-2222

23. Replace the selected XSLT code with the word **View.**



```
                    <xsl:attribute name="href">
                        <xsl:variable xmlns:xsl="http://www.w3.org/1999/XSL/Transform'
                        <xsl:text>javascript:</xsl:text>
                        <xsl:value-of select="ddwrt:GenFireConnection(concat('g_5f06a'
                    </xsl:attribute>
                    View
                    </a></td>
```

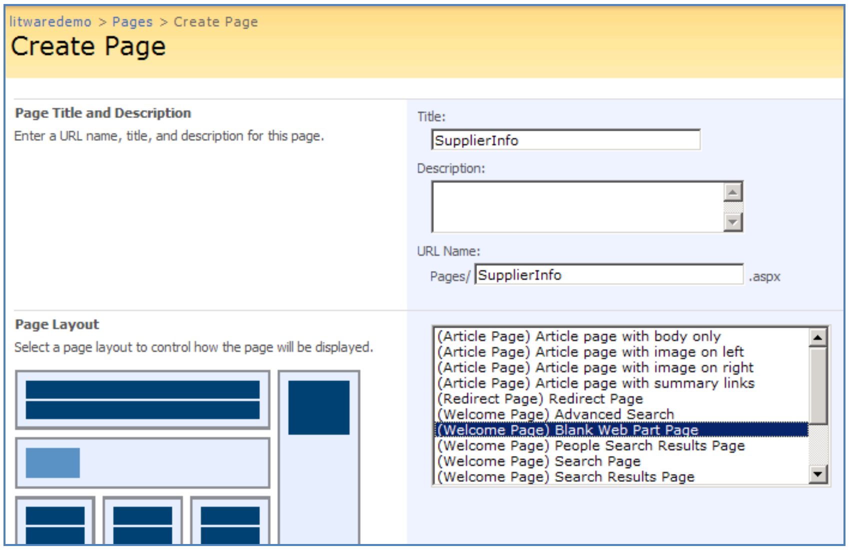
ID	Product Name	Quantity Per Unit	Inventory	Supplier Info
1	Chai	10 boxes x 20 bags	702	View
2	Chang	24 - 12 oz bottles	323	View
3	Aniseed Syrup	12 - 550 ml bottles	130	View

24. Save **Products.aspx** page.
25. Open Internet explorer and browse to **http://litwaredemo/Products.aspx.**
26. Click on the **View** link for each product. Notice how **Suppliers list** will be filtered to show information only about that product's supplier.

Master/Detail filtering across two pages via query string

A common pattern in many web applications is to have the master records on one web page and details shown on another. In this section, we will implement this pattern and build a master/detail filtering with master records coming from Northwind database and detail record(s) filtered in Suppliers List (See Part 1) sitting on another page.

1. browse to **http://litwaredemo.**
2. Click on the **Site Actions** and choose **Create Page.**
3. In the title box, type **SupplierInfo** and choose **(Welcome Page) Blank Web Part Page** as the Page Layout. Click on the **Create** button.



litwaredemo > Pages > Create Page

Create Page

Page Title and Description

Enter a URL name, title, and description for this page.

Title: SupplierInfo

Description:

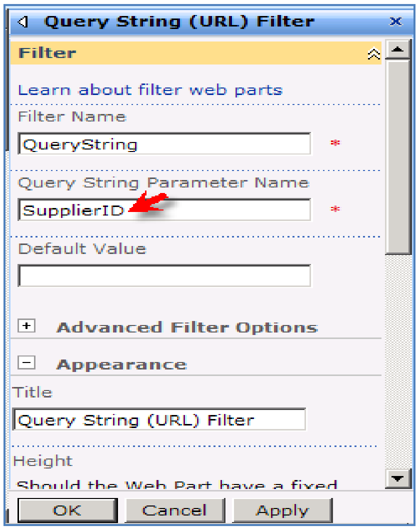
URL Name: Pages/ SupplierInfo .aspx

Page Layout

Select a page layout to control how the page will be displayed.

(Article Page) Article page with body only
(Article Page) Article page with image on left
(Article Page) Article page with image on right
(Article Page) Article page with summary links
(Redirect Page) Redirect Page
(Welcome Page) Advanced Search
(Welcome Page) Blank Web Part Page
(Welcome Page) People Search Results Page
(Welcome Page) Search Page
(Welcome Page) Search Results Page

4. Add the **suppliers** list view Web Part to the center zone.
5. Add the **Query String (URL) Filter** Web Part to the center right zone.
6. Go to **Query String (URL) Filter** Web Part properties and in **Query String Parameter Name** text box, enter **SupplierID.**



Query String (URL) Filter

Filter

Learn about filter web parts

Filter Name: QueryString *

Query String Parameter Name: SupplierID *

Default Value:

Advanced Filter Options

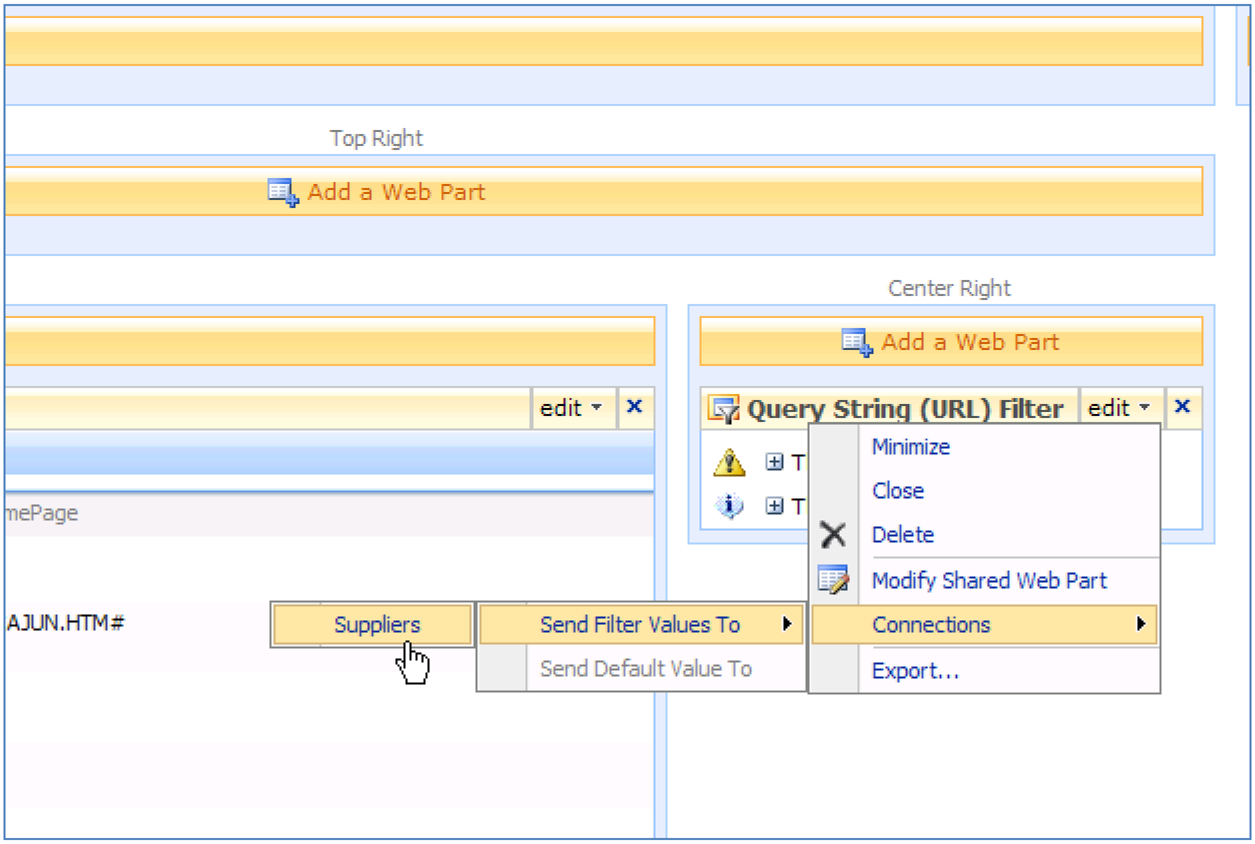
Appearance

Title: Query String (URL) Filter

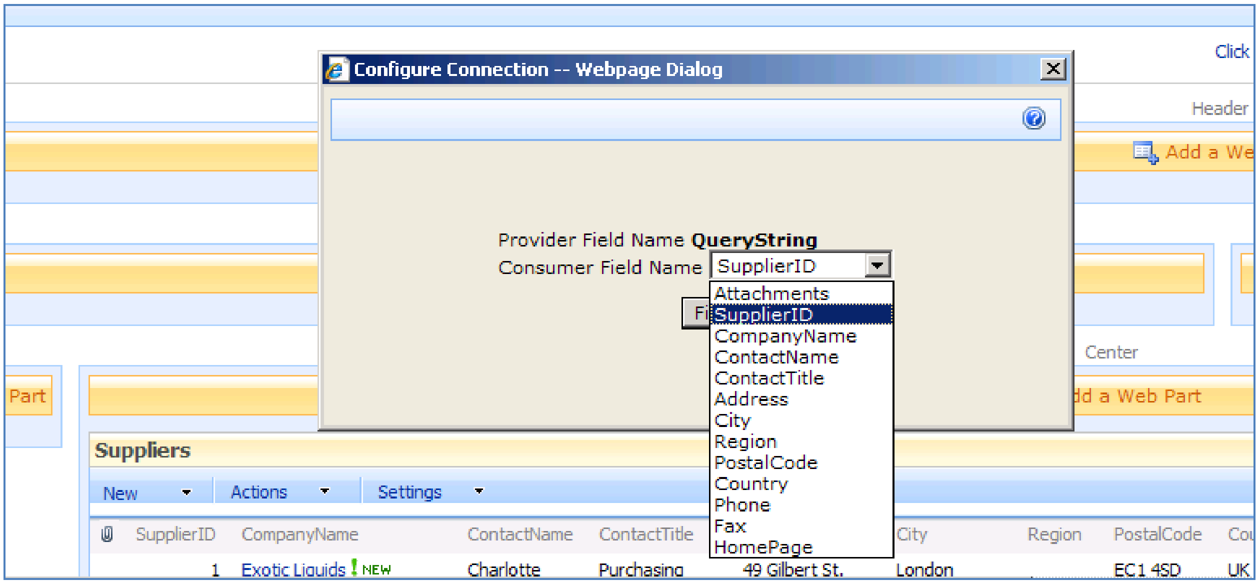
Height

Should the Web Part have a fixed

7. Click **Ok .**
8. Click on the **edit** menu on the **Query String(URL) Filer** Web Part **,** click on **Connections ,Send Filer Value TO** and Choose **Suppliers.**



9. **Configure Connection** dialog box will appear.
10. Choose **SupplierID** from the **Consumer Field Name** dropdown box.

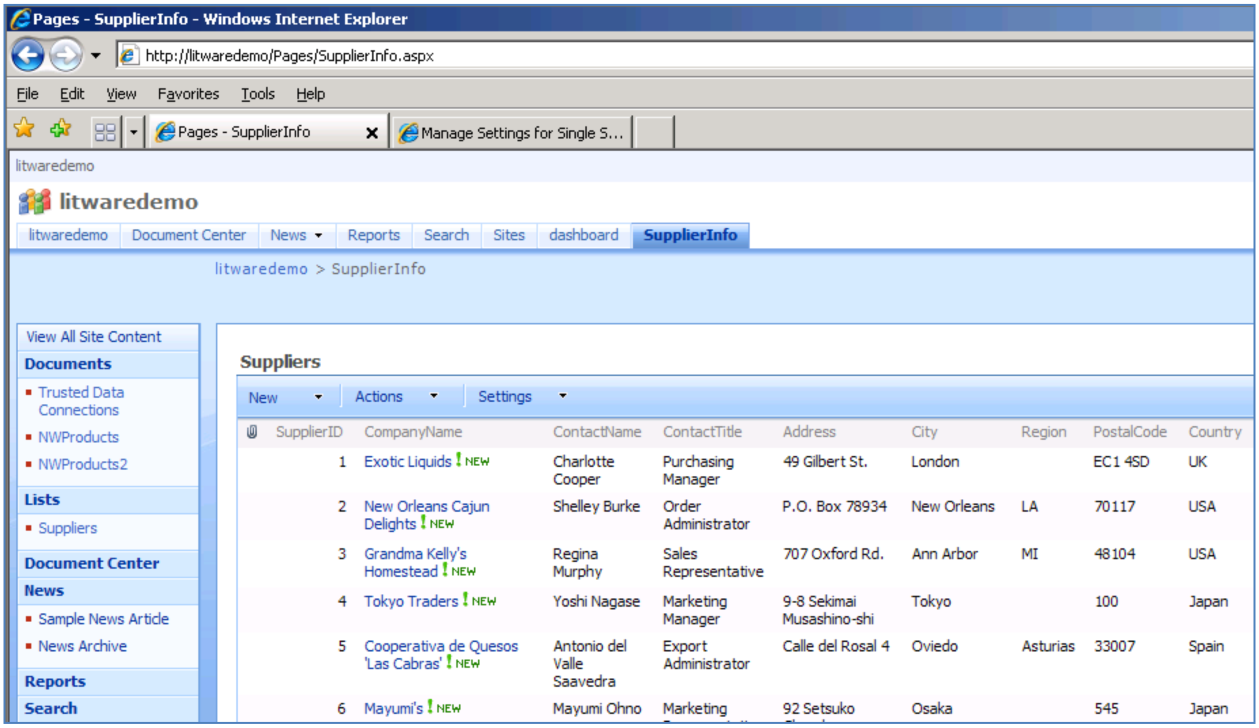


Provider Field Name **QueryString**
Consumer Field Name SupplierID

Suppliers

SupplierID	CompanyName	ContactName	ContactTitle		City	Region	PostalCode	Cou
1	Exotic Liquids ! NEW	Charlotte	Purchasing	49 Gilbert St.	London		EC1 4SD	UK

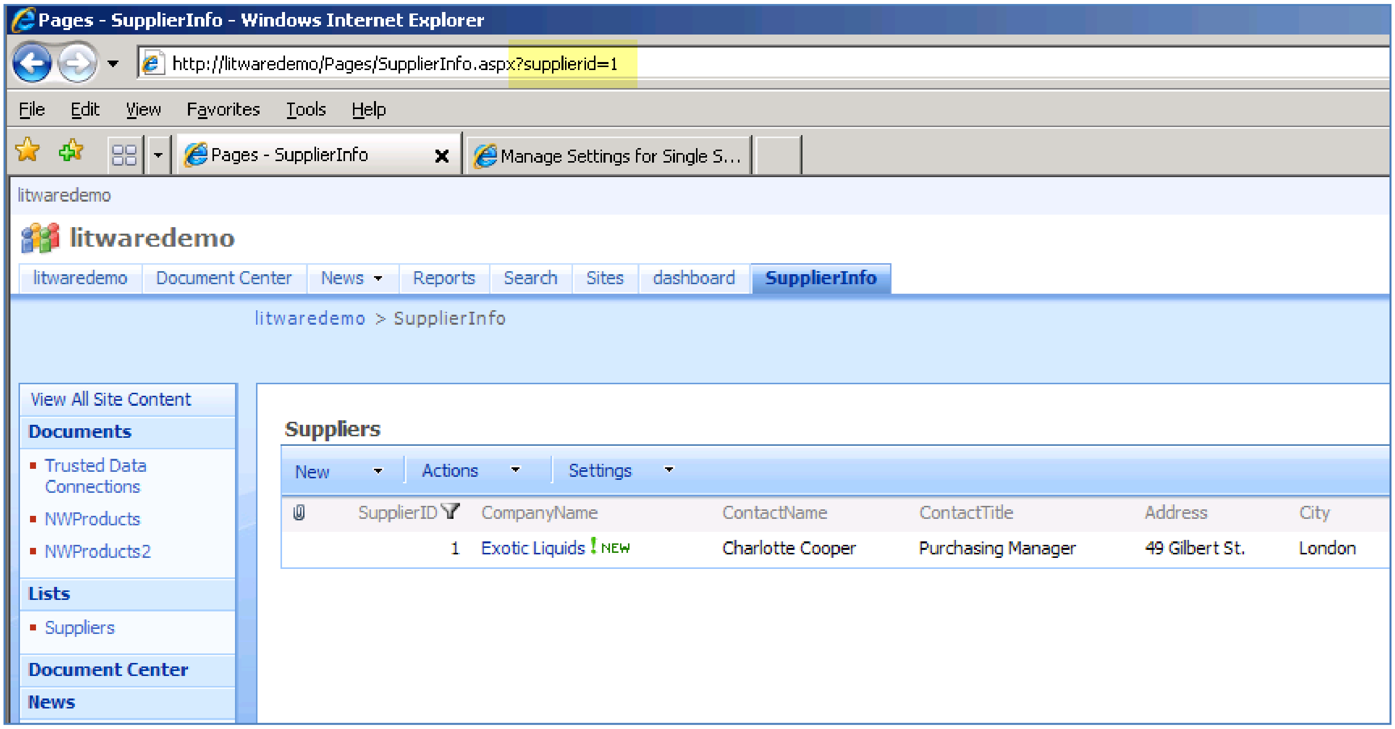
11. Click on the **Publish** button **, SupplierInfo.aspx** without the query string shows all the suppliers.



Suppliers

SupplierID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country
1	Exotic Liquids ! NEW	Charlotte Cooper	Purchasing Manager	49 Gilbert St.	London		EC1 4SD	UK
2	New Orleans Cajun Delights ! NEW	Shelley Burke	Order Administrator	P.O. Box 78934	New Orleans	LA	70117	USA
3	Grandma Kelly's Homestead ! NEW	Regina Murphy	Sales Representative	707 Oxford Rd.	Ann Arbor	MI	48104	USA
4	Tokyo Traders ! NEW	Yoshi Nagase	Marketing Manager	9-8 Sekimai Musashino-shi	Tokyo		100	Japan
5	Cooperativa de Quesos 'Las Cabras' ! NEW	Antonio del Valle Saavedra	Export Administrator	Calle del Rosal 4	Oviedo	Asturias	33007	Spain
6	Mayumi's ! NEW	Mayumi Ohno	Marketing	92 Setsuko	Osaka		545	Japan

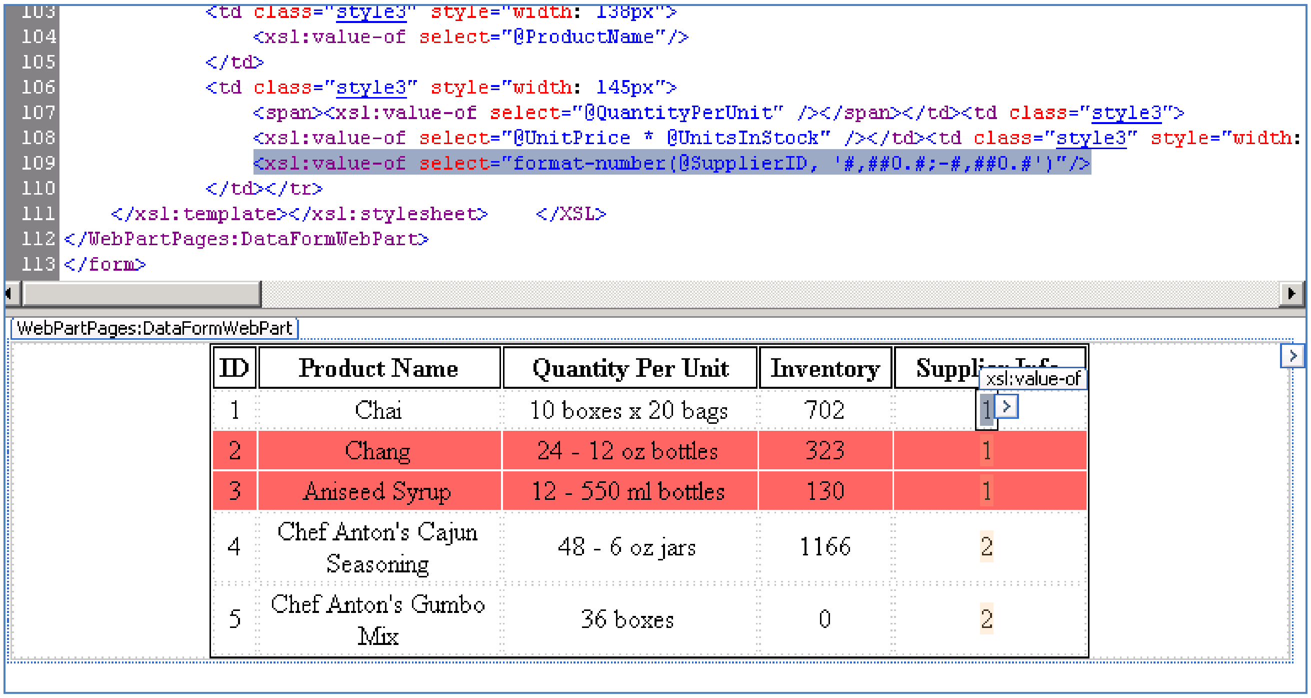
12. Test the filtering on **SupplierInfo.aspx** page by browsing to the page , passing along supplier's **SupplierID** in the query string.



http://litwaredemo/Pages/SupplierInfo.aspx?supplierid=1

SupplierID	CompanyName	ContactName	ContactTitle	Address	City
1	Exotic Liquids ! NEW	Charlotte Cooper	Purchasing Manager	49 Gilbert St.	London

13. With the filtering working properly in the detail page(**SupplierInfo.aspx**), let's get back to the **Products.aspx** and modify the Data Form Web Part to pass the supplier's **SupplierID** in the query string.
14. In the **products.aspx** page **,** we need both static and data-driven values. **/Pages/SupplierInfo.aspx?SupplierID** portion of the link's URL is static, whereas the **SupplierID** portion is data-driven as its value is each row's own **SupplierID.**
36. Switch to "Split View" by clicking the **Split** tab at the lower left of the design surface. In the “Design View” half, Click next to one of the data values of the **Supplier Info column** in the first row of your view.



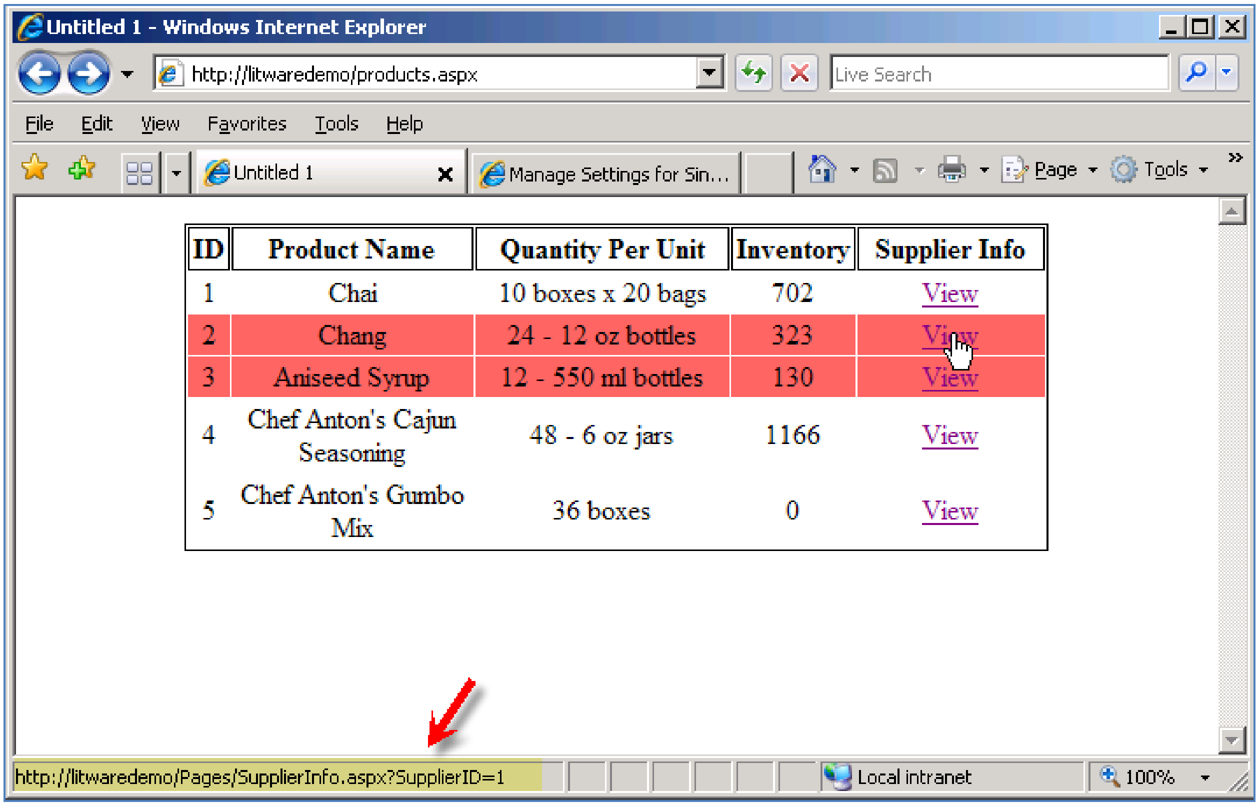
```
            <td class="style3" style="width: 138px">
                <xsl:value-of select="@ProductName"/>
            </td>
            <td class="style3" style="width: 145px">
                <span><xsl:value-of select="@QuantityPerUnit" /></span></td><td class="style3">
                <xsl:value-of select="@UnitPrice * @UnitsInStock" /></td><td class="style3" style="width:
                <xsl:value-of select="format-number(@SupplierID, '#,##0.#;-#,##0.#')"/>
            </td></tr>
    </xsl:template></xsl:stylesheet>    </XSL>
</WebPartPages:DataFormWebPart>
</form>
```

ID	Product Name	Quantity Per Unit	Inventory	Suppl
1	Chai	10 boxes x 20 bags	702	1
2	Chang	24 - 12 oz bottles	323	1
3	Aniseed Syrup	12 - 550 ml bottles	130	1
4	Chef Anton's Cajun Seasoning	48 - 6 oz jars	1166	2
5	Chef Anton's Gumbo Mix	36 boxes	0	2

15. Replace the highlighted XSLT with the following :

```
<a target="_self">
        <xsl:attribute name="href">
        <xsl:variable xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
        name="fields">SupplierID=<xsl:value-of
        select="ddwrt:ConnEncode(string(@SupplierID))" />
        </xsl:variable>
        <xsl:text>/Pages/SupplierInfo.aspx?</xsl:text>
        <xsl:value-of select="$fields"></xsl:value-of>
        </xsl:attribute>
        View
</a>
```

16. Save **Products.aspx** page.
17. Open Internet explorer and browse to **http://litwaredemo/Products.aspx.** Notice how correct query string built based on the **SupplierID** of the product.

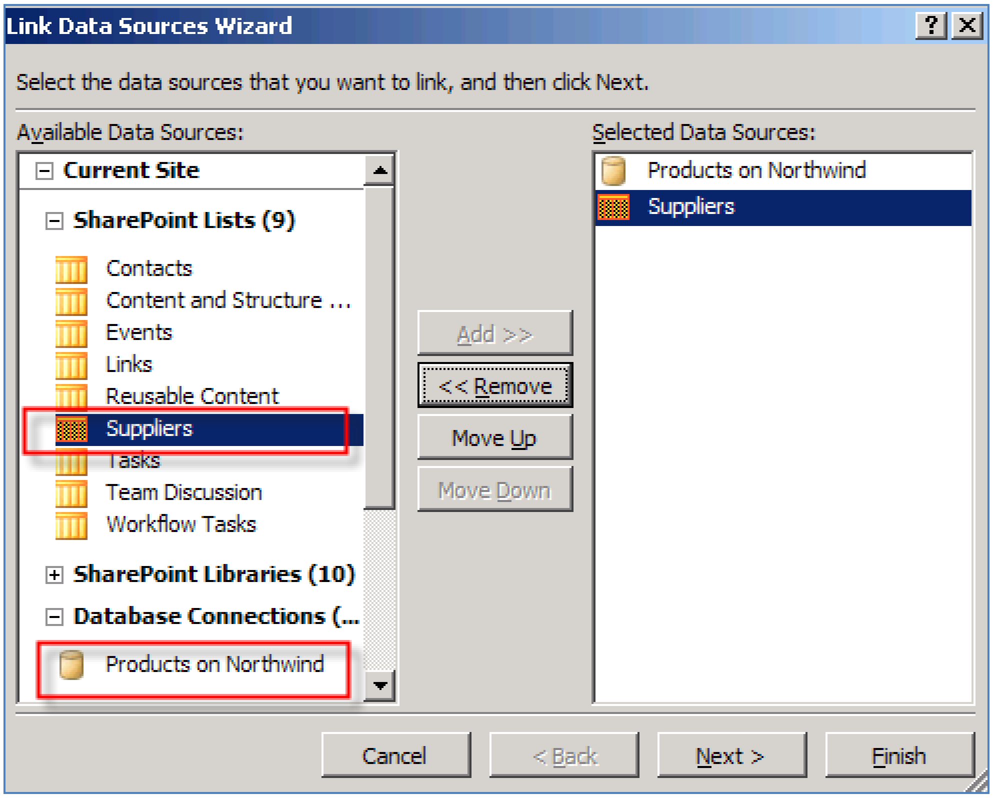

Untitled 1 - Windows Internet Explorer
http://litwaredemo/products.aspx
Live Search
File
Edit
View
Favorites
Tools
Help
Untitled 1
Manage Settings for Sin...
Page
Tools
http://litwaredemo/Pages/SupplierInfo.aspx?SupplierID=1
Local intranet
100%

ID	Product Name	Quantity Per Unit	Inventory	Supplier Info
1	Chai	10 boxes x 20 bags	702	View
2	Chang	24 - 12 oz bottles	323	View
3	Aniseed Syrup	12 - 550 ml bottles	130	View
4	Chef Anton's Cajun Seasoning	48 - 6 oz jars	1166	View
5	Chef Anton's Gumbo Mix	36 boxes	0	View

Linked data sources

Most enterprises store different types of data in various data stores with different data models. The challenge here is how to provide a complete and easy view into all this data sitting in various stores. It's a common practice to use Microsoft Office SharePoint Designer 2007 to link these data sources .By doing so; you can create a single data source that can be treated like any other data source available in Data Source Library. The beauty of process is that You can link data sources of separate types, for example, you can link an SharePoint list to a table which resides in SQL database as long as linked data sources contain data that can be interrelated to each other.

1. Start SharePoint Designer.
2. Select the **OpenSite** option in the **File** menu.
3. Enter **http://litwaredem** in the **Site name** text box.
4. Click in the **File** menu , **New** and then from **Page** tab choose **ASPX**.
5. Save untitled.aspx as **LinkDS.aspx.**
6. Expand **Linked sources** in the **Data Source Library** pane and click **Create a new Linked Source...**
7. In the **Data Source Properties** dialog box, click on the **Configured Linked Source...**
8. **Link Data Sources Wizard** will open.
9. From **Available Data Sources** list, select **Products on Northwind** and **Suppliers** list. Click **Next** .



Link Data Sources Wizard

Select the data sources that you want to link, and then click Next.

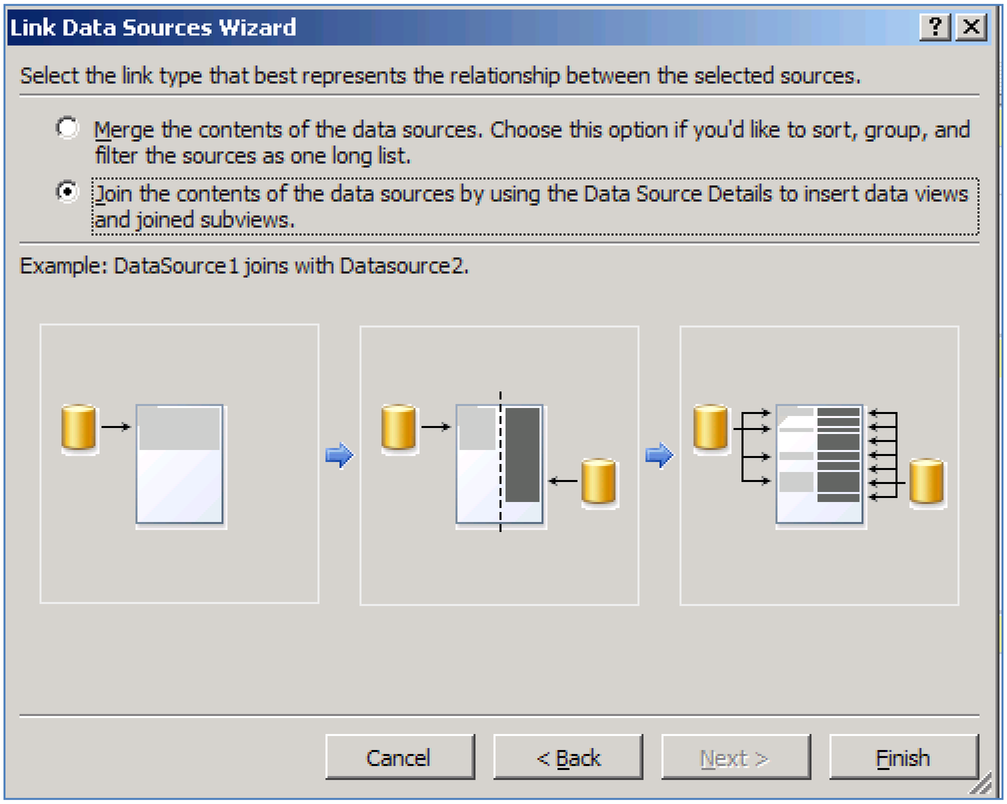
Available Data Sources:

- Current Site
  - SharePoint Lists (9)
    - Contacts
    - Content and Structure ...
    - Events
    - Links
    - Reusable Content
    - Suppliers
    - Tasks
    - Team Discussion
    - Workflow Tasks
  - SharePoint Libraries (10)
  - Database Connections (...
    - Products on Northwind

Selected Data Sources:

- Products on Northwind
- Suppliers

10. Select the section option to join the contents of the data sources. Click **Next.**



Link Data Sources Wizard

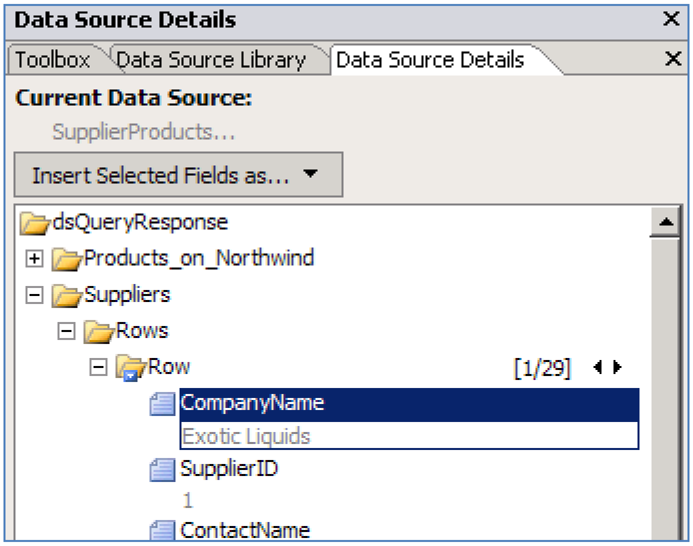
Select the link type that best represents the relationship between the selected sources.

○ Merge the contents of the data sources. Choose this option if you'd like to sort, group, and filter the sources as one long list.

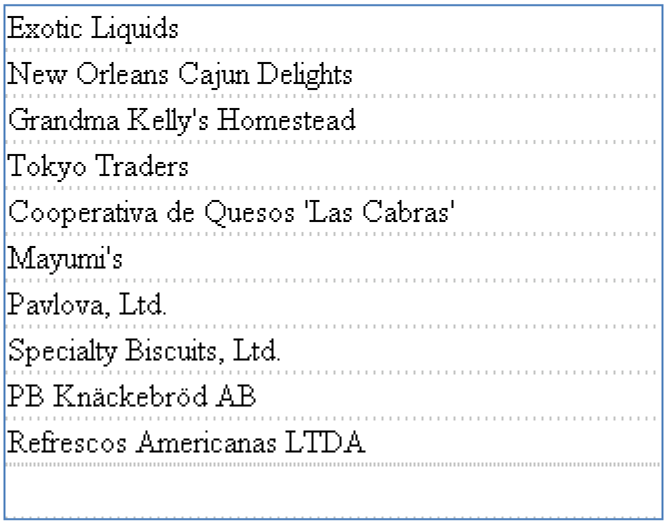
● Join the contents of the data sources by using the Data Source Details to insert data views and joined subviews.

Example: DataSource1 joins with Datasource2.

11. Click **Finish.**
12. Click on the **General** Tab and type a name of the new linked data source , for example **SupplierProductsDS.**
13. Click **Ok.**
14. Click on **SupplierProductsDS** and select the **Show Data** option.
15. In the tree view , highlight **CompanyName** in the **Suppliers** data source.

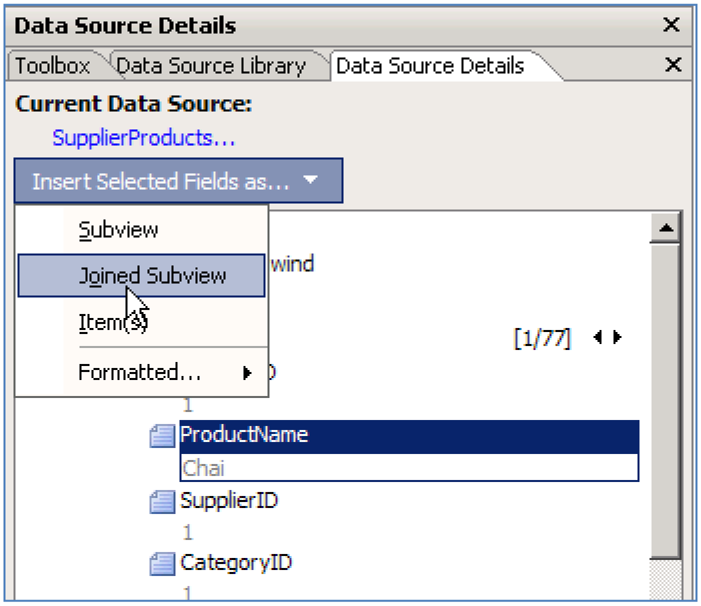


16. Drag and drop this field on to the design surface.
17. Delete the **CompanyName** title. Your design surface should look like this:

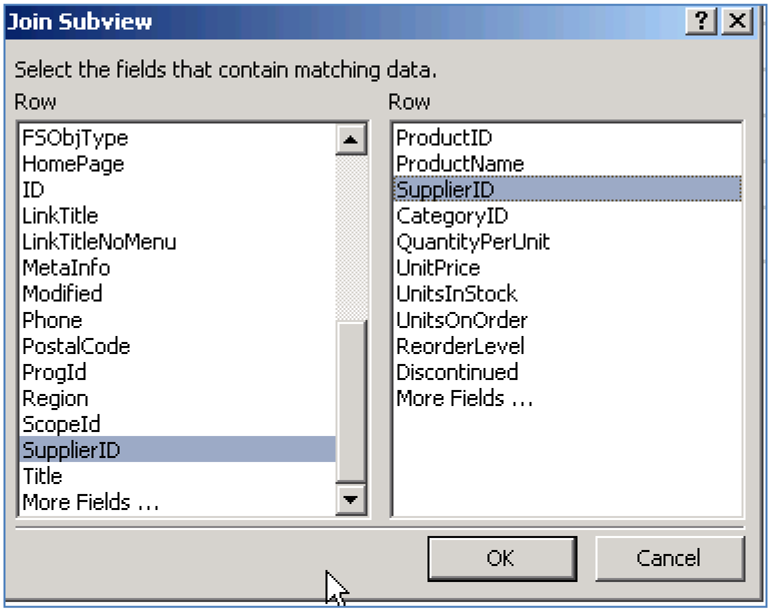


Exotic Liquids
New Orleans Cajun Delights
Grandma Kelly's Homestead
Tokyo Traders
Cooperativa de Quesos 'Las Cabras'
Mayumi's
Pavlova, Ltd.
Specialty Biscuits, Ltd.
PB Knäckebröd AB
Refrescos Americanas LTDA

18. Click next to one of the data values in the first row of your view - for example "**Exotic Liquids**", and now in the **Data Source Details** Pane, Select **ProductName** from **Products_on_Northwind** data source.
19. Click on Insert Selected Fields as... and choose to insert as "Joined SubView" in the menu shown below:

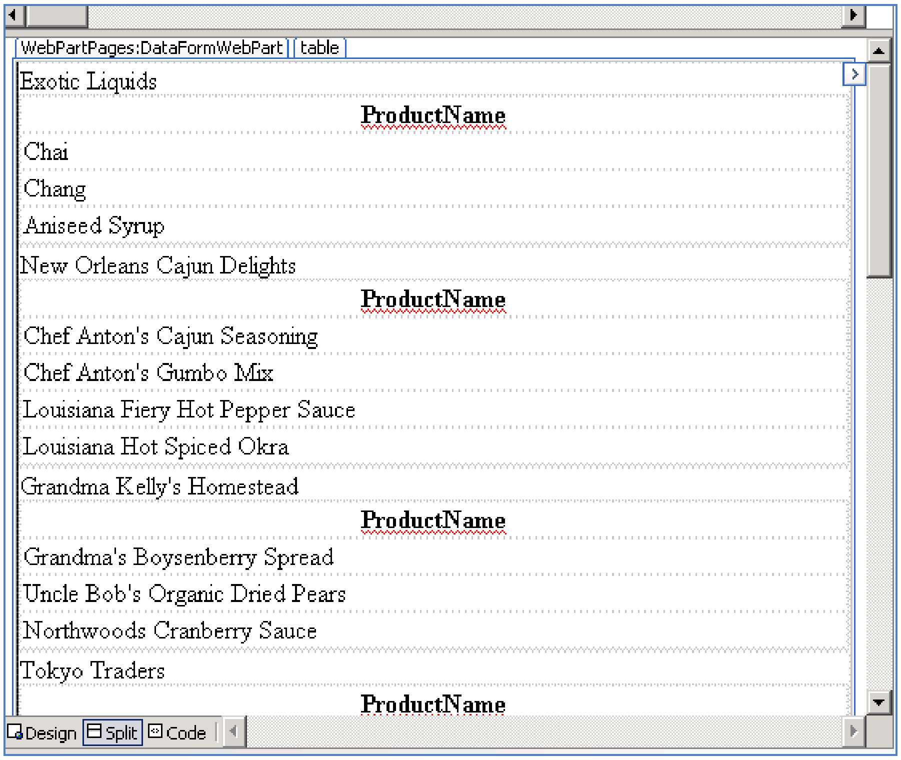


20. Specify SupplierID from left list and SupplierID from right list to be the join columns for both data sources.


Join Subview

Select the fields that contain matching data.

21. Your design surface should look like this:



Exotic Liquids

ProductName

Chai

Chang

Aniseed Syrup

New Orleans Cajun Delights

ProductName

Chef Anton's Cajun Seasoning

Chef Anton's Gumbo Mix

Louisiana Fiery Hot Pepper Sauce

Louisiana Hot Spiced Okra

Grandma Kelly's Homestead

ProductName

Grandma's Boysenberry Spread

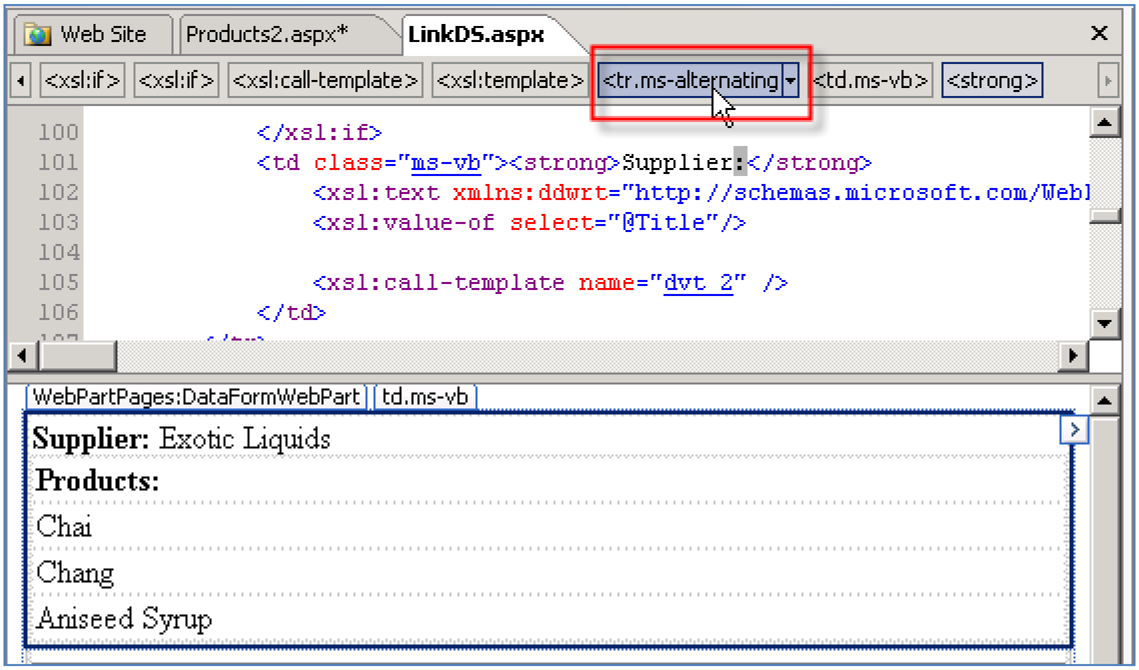
Uncle Bob's Organic Dried Pears

Northwoods Cranberry Sauce

Tokyo Traders

ProductName

22. Change the **ProductName** to **Products.**
23. Click on the left side of one of the data values in the first row of your view - for example "**Exotic Liquids**".
24. Type "Supplier:" and make it bold.
25. Click next to one of the data values in the first row of your view. Using the Quick Tag Selector, select the whole row by clicking the TR button.



```
            </xsl:if>
            <td class="ms-vb"><strong>Supplier:</strong>
                <xsl:text xmlns:ddwrt="http://schemas.microsoft.com/Webl
                <xsl:value-of select="@Title"/>

                <xsl:call-template name="dvt_2" />
            </td>
```

Supplier: Exotic Liquids

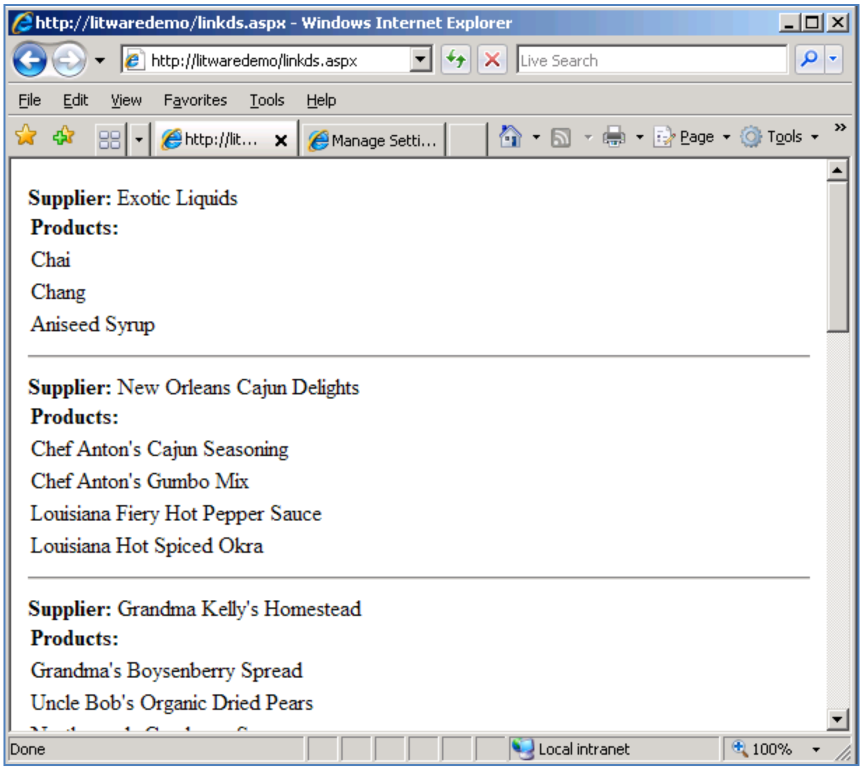
Products:

Chai

Chang

Aniseed Syrup

26. Paste <tr><td><hr/></td></tr> right after this TR ends. This will add a line between each supplier.
27. Save **LinkDS.aspx.**
28. Open Internet explorer and browse to **http://litwaredemo/ LinkDS.aspx.**



Supplier: Exotic Liquids
Products:
Chai
Chang
Aniseed Syrup

Supplier: New Orleans Cajun Delights
Products:
Chef Anton's Cajun Seasoning
Chef Anton's Gumbo Mix
Louisiana Fiery Hot Pepper Sauce
Louisiana Hot Spiced Okra

Supplier: Grandma Kelly's Homestead
Products:
Grandma's Boysenberry Spread
Uncle Bob's Organic Dried Pears

Create your own Data Form Web Part

In nutshell, Data Form Web Part is just an XSLT-rendering Web Part that receives its data in xml format from a Data Source Control. Most of the time , people use Microsoft SharePoint Designer 2007 to create Data Form Web Part or even for deploying it, but the truth is, SharePoint Designer is not the only way you can benefit from all the great things, Data Form Web Part offers to get data on the page with minimal effort and coding!

Imagine this: a WebPart that you can just drag and drop on a page, set couple of properties and it renders the data from your backend system . Picture this: you can completely change the look and feel of your Web Part by editing its associated XSL stylesheets . I am not going to walk you through a full-blown custom Data Form Web Part here. Instead, I'd like to demonstrate how easily you can create a generic one and I will leave the specific implementation up to you.

Important
Unless you have not noticed so far, many of the out of the box Web Parts are derived from Data Form Web Part - for example, RSS Web Part, Business Data List Web Part and Business Data Related Web Part .

In order to create a generic Data Form Web Part, there are only three properties that matter the most:

a. Data Source.
b. XSL Stylesheets.
c. Parameter bindings.

When you inherit your Web Part from **Microsoft.SharePoint.WebPartPages.DataFormWebPart** class, you will automatically get the XSLT and Parameter bindings properties along with bunch of other properties specific to Data Form Web Part. Having said this, all we need to do in our generic Data Form Web Part is to expose the data source as a custom property, hook it up to **DataSourceString** property of Data Form Web Part and then call the **DataBind** method at runtime. Let's get this Web Part coded:

1. Start **Visual Studio.**
2. Create a new **Class Library** project and name it **DFWP.**
3. Add a new class and name it **DFWPx.cs.**
4. Make that class inherit from **Microsoft.SharePoint.WebPartPages.DataFormWebPart.**
5. Add a property to the class and name it **GenericDataSource.** This is the property that appears in the Web Part's properties and gives you the ability to type the data source at runtime.

```
private string genericDataSource;
[WebBrowsable(true)]
[DefaultValue("")]
[WebDescription("Enter the data source for this Data View Web Part")]
[WebDisplayName("DataSource")]
[Personalizable(PersonalizationScope.Shared)]
[Category("Data Source")]
[ReadOnly(false)]
public string GenericDataSource
{
  get
  {
    return genericDataSource;
  }

  set
  {
    genericDataSource = value;
  }
```

6. Override the **OnInit** method and copy and paste the following code snippet into this method.

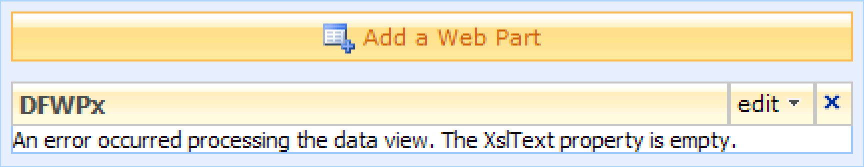
```
protected override void OnInit(EventArgs e)
{
  base.OnInit(e);
  if (! string.IsNullOrEmpty(this.GenericDataSource))
  {
    BuiildGenericDataSource();
    this.DataBind();
  }
}
```

7. Add the following helper method to your class:

```
private void BuiildGenericDataSource()
    {
        string registerTagPrefix = "<%@Register TagPrefix=\"sharepoint\"
        Namespace=\"Microsoft.SharePoint.WebControls\" Assembly=\"Microsoft.SharePoint,
        Version=12.0.0.0, Culture=neutral, PublicKeyToken=71e9bce111e9429c\" %>";
         this.DataSourcesString = string.Concat(registerTagPrefix, this.GenericDataSource);
    }
```

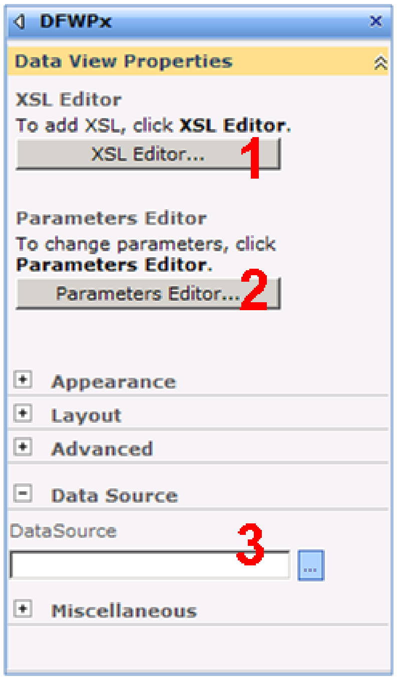
Source code for this class can be download at [http://blogs.devhorizon.com/reza/wp-content/uploads/2008/03/DFWPx. rar](http://blogs.devhorizon.com/reza/wp-content/uploads/2008/03/DFWPx.rar) .

8. Compile the project and deploy the Web Part (Deployment steps are not covered here).
9. For the first time , we you add the Web Part to a zone, you get an error like what is shown below. It is just complaining that there is no XSLT. Move on to the next step.



An error occurred processing the data view. The XslText property is empty.

10. Click on the **edit** menu on the Web Part and select **Modify Shared Web Part** .
11. This will open **DFWPx** Web Part's **properties** pane. Notice our custom **DataSource** property shown underneath **Data Source** section. In the properties pane, you need to change three properties as marked in the picture below:

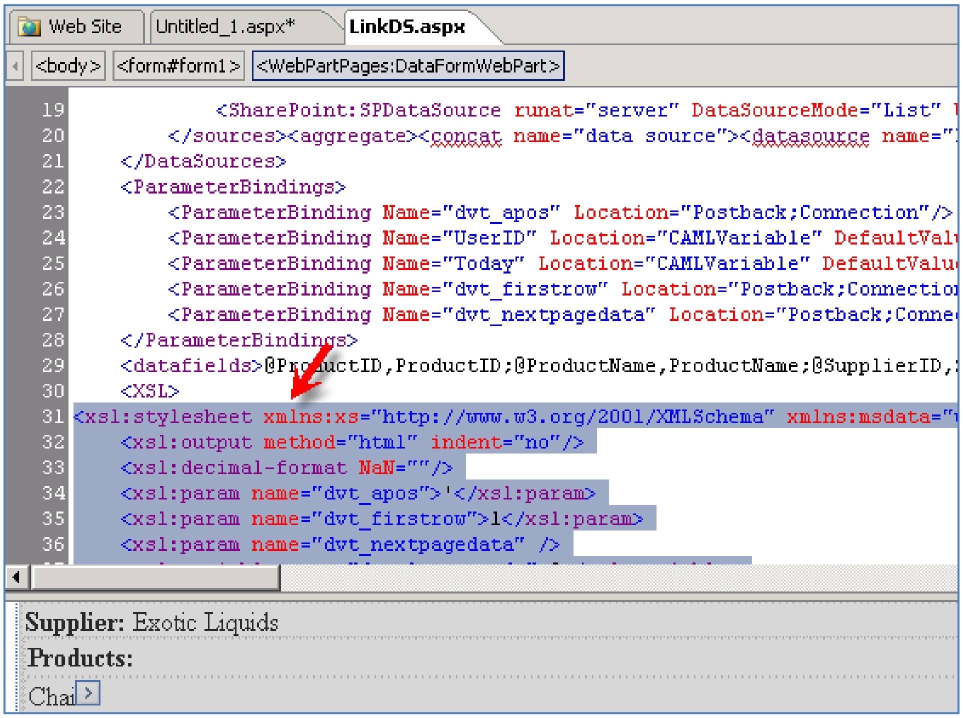


To add XSL, click **XSL Editor**.

To change parameters, click **Parameters Editor**.

12. Open the **LinkDS.aspx** page we created in the previous section in **Microsoft SharePoint Designer 2007**. What we are going to do here , is one of the many example that **Microsoft SharePoint Designer 2007** can make you life much easier both as developer or non-developer.

13. Switch to "Split View" by clicking the **Split** tab at the lower left of the design surface. In the “Design View” half, click next to the first row. In the "Code View", locate <XSL> tag and highlight the whole <xsl:stylesheet> tag. Copy it to the **clipboard** .



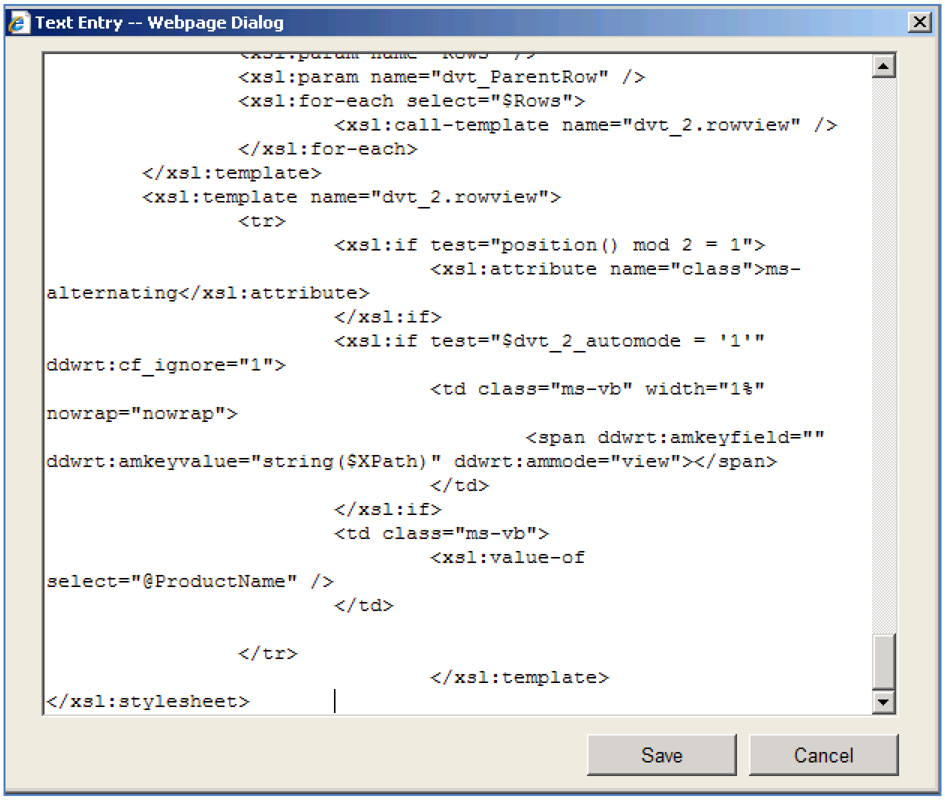
```
            <SharePoint:SPDataSource runat="server" DataSourceMode="List"
        </sources><aggregate><concat name="data source"><datasource name="
    </DataSources>
    <ParameterBindings>
        <ParameterBinding Name="dvt_apos" Location="Postback;Connection"/>
        <ParameterBinding Name="UserID" Location="CAMLVariable" DefaultVal
        <ParameterBinding Name="Today" Location="CAMLVariable" DefaultValu
        <ParameterBinding Name="dvt_firstrow" Location="Postback;Connectio
        <ParameterBinding Name="dvt_nextpagedata" Location="Postback;Conne
    </ParameterBindings>
    <datafields>@ProductID,ProductID;@ProductName,ProductName;@SupplierID,
    <XSL>
<xsl:stylesheet xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:msdata="
    <xsl:output method="html" indent="no"/>
    <xsl:decimal-format NaN=""/>
    <xsl:param name="dvt_apos">'</xsl:param>
    <xsl:param name="dvt_firstrow">1</xsl:param>
    <xsl:param name="dvt_nextpagedata" />
```

Supplier: Exotic Liquids

Products:

Chai

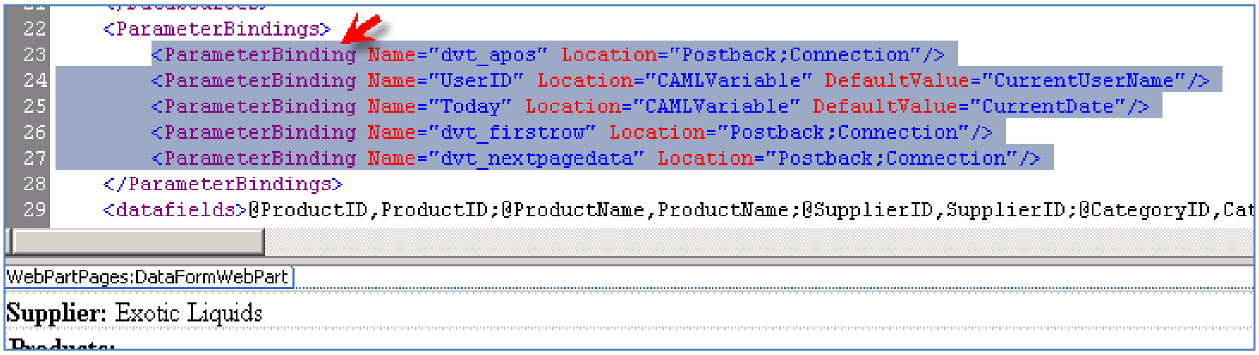
14. In the Web Part's properties pane, click on the **XSLT Editor...** and paste it in the **text entry** dialog box. Click **Save.**



```
                <xsl:param name="Rows" />
                <xsl:param name="dvt_ParentRow" />
                <xsl:for-each select="$Rows">
                        <xsl:call-template name="dvt_2.rowview" />
                </xsl:for-each>
        </xsl:template>
        <xsl:template name="dvt_2.rowview">
                <tr>
                        <xsl:if test="position() mod 2 = 1">
                                <xsl:attribute name="class">ms-
alternating</xsl:attribute>
                        </xsl:if>
                        <xsl:if test="$dvt_2_automode = '1'"
ddwrt:cf_ignore="1">
                                <td class="ms-vb" width="1%"
nowrap="nowrap">
                                        <span ddwrt:amkeyfield=""
ddwrt:amkeyvalue="string($XPath)" ddwrt:ammode="view"></span>
                                </td>
                        </xsl:if>
                        <td class="ms-vb">
                                <xsl:value-of
select="@ProductName" />
                        </td>

                </tr>
                                </xsl:template>
</xsl:stylesheet>
```

15. Back to SharePoint designer. This time , locate <ParamtereBindings> tag and xopy and paste the content into the clipboard.

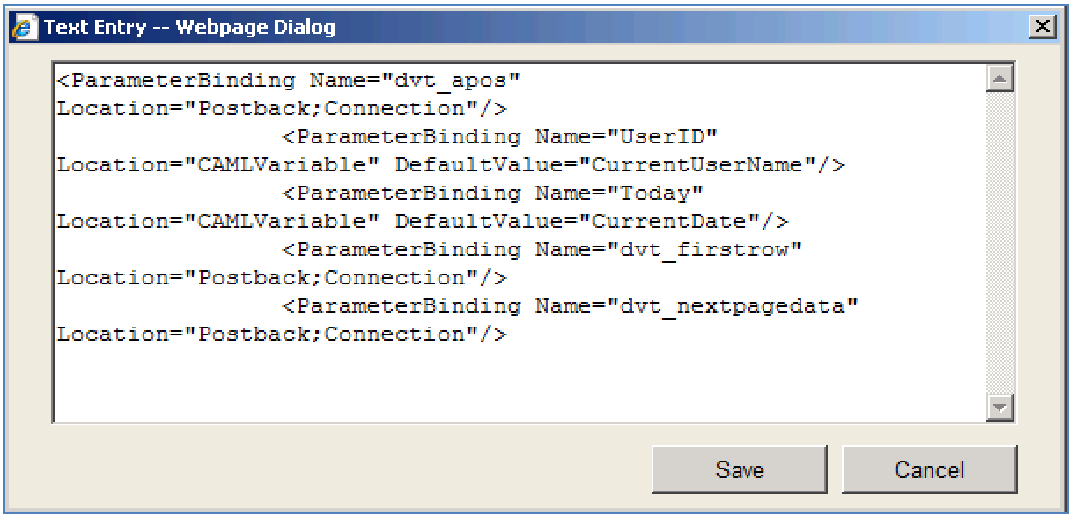


```
    </DataSources>
    <ParameterBindings>
        <ParameterBinding Name="dvt_apos" Location="Postback;Connection"/>
        <ParameterBinding Name="UserID" Location="CAMLVariable" DefaultValue="CurrentUserName"/>
        <ParameterBinding Name="Today" Location="CAMLVariable" DefaultValue="CurrentDate"/>
        <ParameterBinding Name="dvt_firstrow" Location="Postback;Connection"/>
        <ParameterBinding Name="dvt_nextpagedata" Location="Postback;Connection"/>
    </ParameterBindings>
    <datafields>@ProductID,ProductID;@ProductName,ProductName;@SupplierID,SupplierID;@CategoryID,Cat
```

Supplier: Exotic Liquids

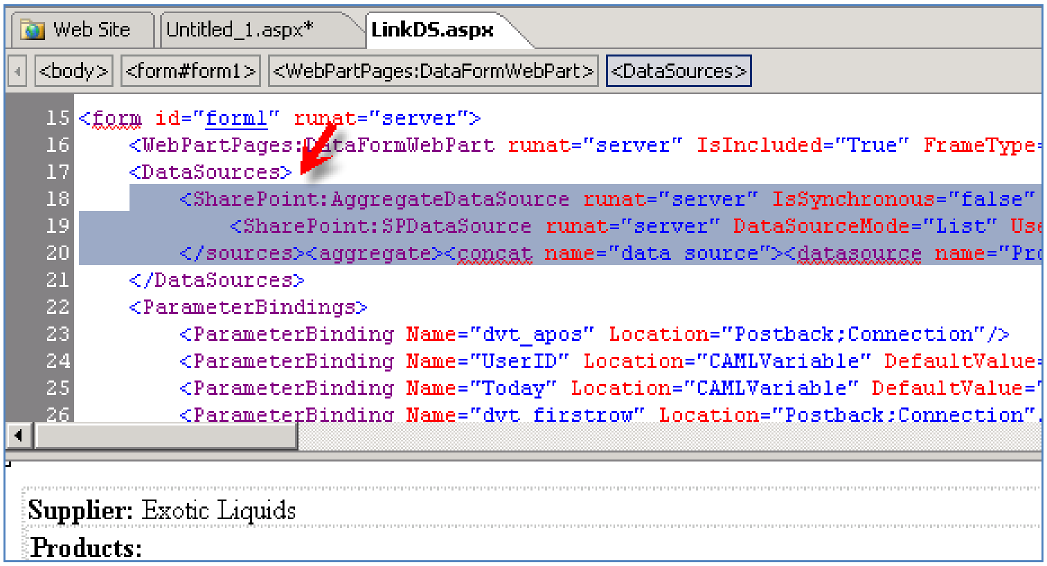
Products:

16. In the Web Part's properties pane, click on the **Parameters Editor...** and paste it in the **text entry** dialog box. Click **Save.**



```
<ParameterBinding Name="dvt_apos"
Location="Postback;Connection"/>
                <ParameterBinding Name="UserID"
Location="CAMLVariable" DefaultValue="CurrentUserName"/>
                <ParameterBinding Name="Today"
Location="CAMLVariable" DefaultValue="CurrentDate"/>
                <ParameterBinding Name="dvt_firstrow"
Location="Postback;Connection"/>
                <ParameterBinding Name="dvt_nextpagedata"
Location="Postback;Connection"/>
```

17. This time in SharePoint designer, locate <DataSources> tag and copy and paste the content.

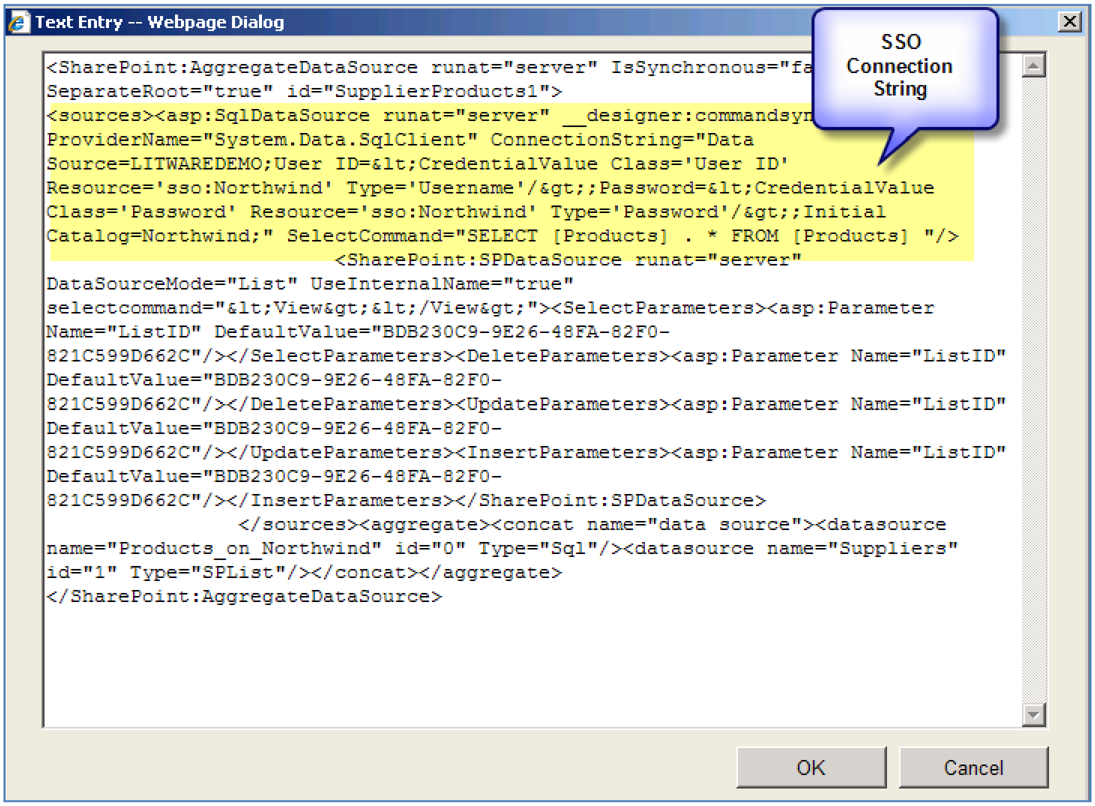


```
<form id="form1" runat="server">
    <WebPartPages:DataFormWebPart runat="server" IsIncluded="True" FrameType=
    <DataSources>
        <SharePoint:AggregateDataSource runat="server" IsSynchronous="false"
            <SharePoint:SPDataSource runat="server" DataSourceMode="List" Use
        </sources><aggregate><concat name="data source"><datasource name="Pro
    </DataSources>
    <ParameterBindings>
        <ParameterBinding Name="dvt_apos" Location="Postback;Connection"/>
        <ParameterBinding Name="UserID" Location="CAMLVariable" DefaultValue=
        <ParameterBinding Name="Today" Location="CAMLVariable" DefaultValue="
        <ParameterBinding Name="dvt_firstrow" Location="Postback;Connection".
```

Supplier: Exotic Liquids

Products:

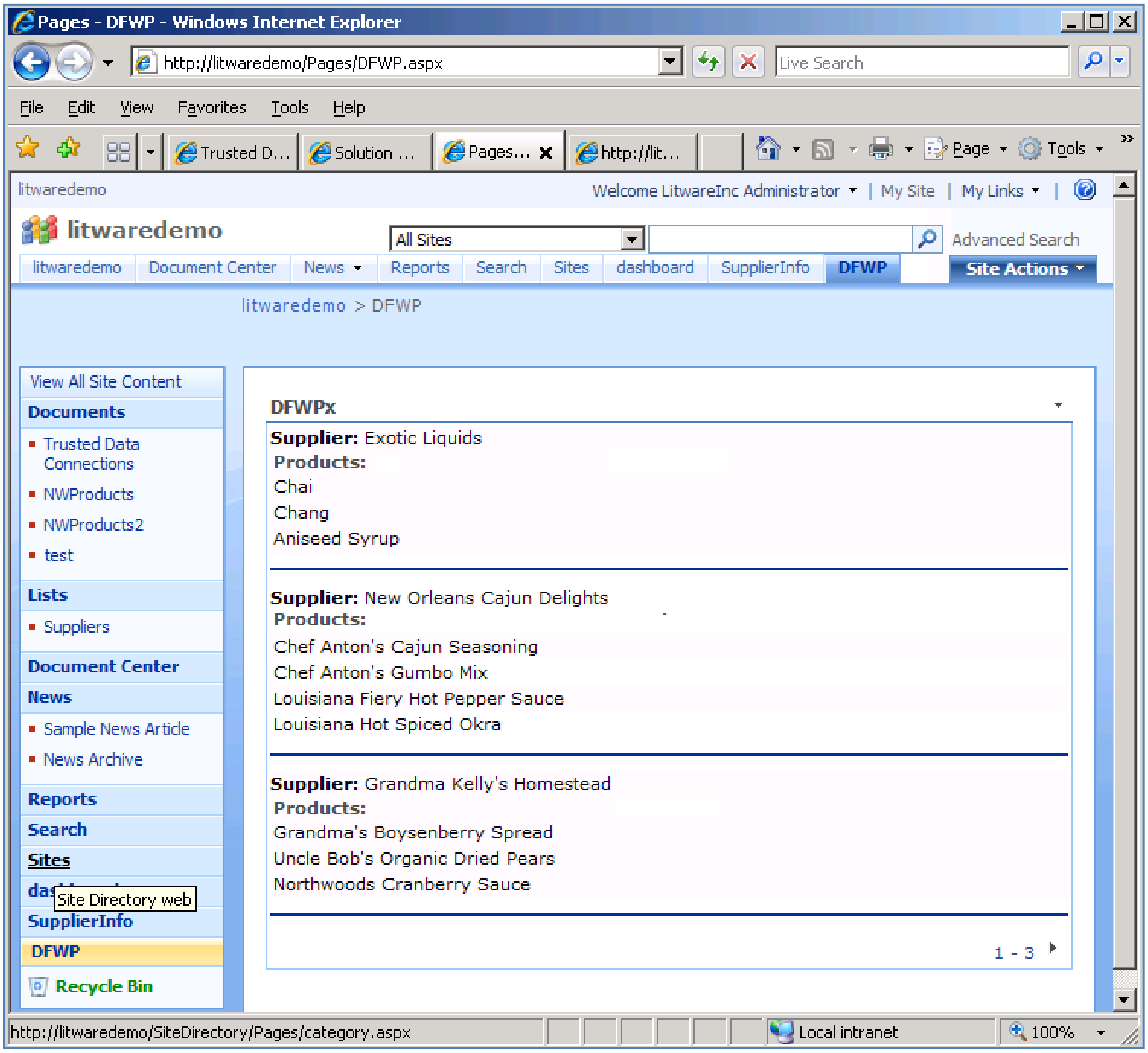
18. In the Web Part's properties pane, underneath **Data Source** section click on the **...** next to **DataSource** text box.
19. and paste it in the **text entry** dialog box. Notice the SSO connection string that SharePoint Designer has generated. Click **OK.**



```
<SharePoint:AggregateDataSource runat="server" IsSynchronous="fa
SeparateRoot="true" id="SupplierProducts1">
<sources><asp:SqlDataSource runat="server" __designer:commandsyn
ProviderName="System.Data.SqlClient" ConnectionString="Data
Source=LITWAREDEMO;User ID=&lt;CredentialValue Class='User ID'
Resource='sso:Northwind' Type='Username'/&gt;;Password=&lt;CredentialValue
Class='Password' Resource='sso:Northwind' Type='Password'/&gt;;Initial
Catalog=Northwind;" SelectCommand="SELECT [Products] . * FROM [Products] "/>
                        <SharePoint:SPDataSource runat="server"
DataSourceMode="List" UseInternalName="true"
selectcommand="&lt;View&gt;&lt;/View&gt;"><SelectParameters><asp:Parameter
Name="ListID" DefaultValue="BDB230C9-9E26-48FA-82F0-
821C599D662C"/></SelectParameters><DeleteParameters><asp:Parameter Name="ListID"
DefaultValue="BDB230C9-9E26-48FA-82F0-
821C599D662C"/></DeleteParameters><UpdateParameters><asp:Parameter Name="ListID"
DefaultValue="BDB230C9-9E26-48FA-82F0-
821C599D662C"/></UpdateParameters><InsertParameters><asp:Parameter Name="ListID"
DefaultValue="BDB230C9-9E26-48FA-82F0-
821C599D662C"/></InsertParameters></SharePoint:SPDataSource>
                </sources><aggregate><concat name="data source"><datasource
name="Products_on_Northwind" id="0" Type="Sql"/><datasource name="Suppliers"
id="1" Type="SPList"/></concat></aggregate>
</SharePoint:AggregateDataSource>
```

20. Click **Ok** in the properties page (You may not see the result yet).

21. Click Publish on the page toolbar. Here is the result .
22. Paginate through the result to make sure it works properly.


Pages - DFWP - Windows Internet Explorer
http://litwaredemo/Pages/DFWP.aspx
Live Search
File Edit View Favorites Tools Help
Trusted D… Solution … Pages… http://lit…
Page Tools
litwaredemo
Welcome LitwareInc Administrator | My Site | My Links |
litwaredemo
All Sites
Advanced Search
litwaredemo Document Center News Reports Search Sites dashboard SupplierInfo DFWP
Site Actions
litwaredemo > DFWP
View All Site Content
Documents
Trusted Data Connections
NWProducts
NWProducts2
test
Lists
Suppliers
Document Center
News
Sample News Article
News Archive
Reports
Search
Sites
Site Directory web
SupplierInfo
DFWP
Recycle Bin
DFWPx
http://litwaredemo/SiteDirectory/Pages/category.aspx
Local intranet
100%

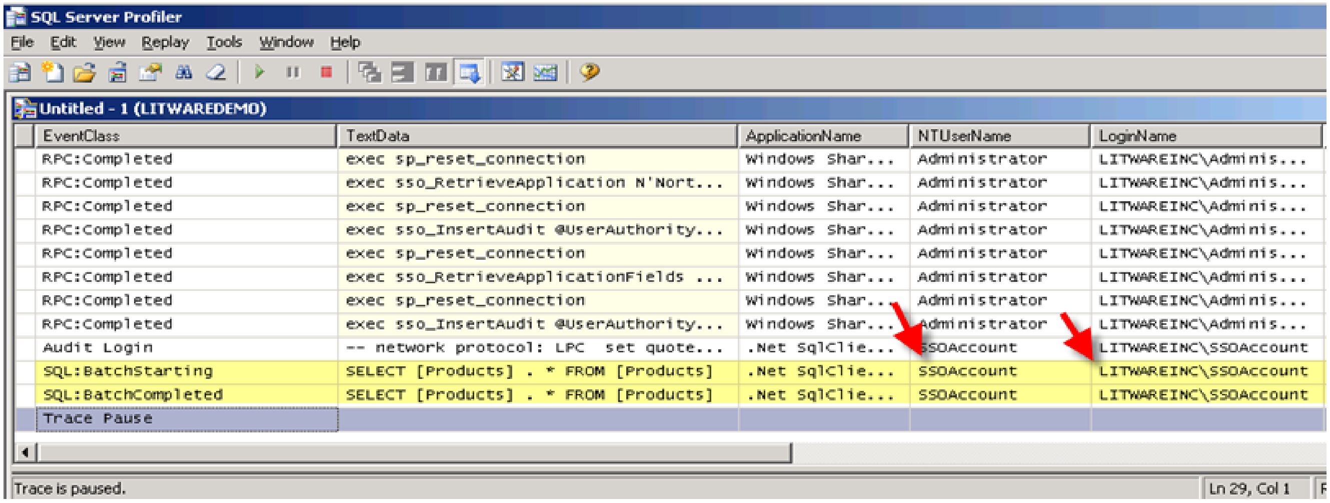
Supplier: Exotic Liquids
Products:
Chai
Chang
Aniseed Syrup

Supplier: New Orleans Cajun Delights
Products:
Chef Anton's Cajun Seasoning
Chef Anton's Gumbo Mix
Louisiana Fiery Hot Pepper Sauce
Louisiana Hot Spiced Okra

Supplier: Grandma Kelly's Homestead
Products:
Grandma's Boysenberry Spread
Uncle Bob's Organic Dried Pears
Northwoods Cranberry Sauce

1 - 3

23. If you start up a new trace in the **SQL Server Profiler** and monitor **SQL:BatchStarting** and **SQL:BacthCompleted** events , you will see how **Litwareinc\administrator** (the account that was browsing to the DFWP page shown in the picture above) is mapped to the **Litwareinc\SSOAccount** which was set up as SSO account in Northwind Application definition .


SQL Server Profiler
File Edit View Replay Tools Window Help
Untitled - 1 (LITWAREDEMO)

EventClass	TextData	ApplicationName	NTUserName	LoginName
RPC:Completed	exec sp_reset_connection	Windows Shar...	Administrator	LITWAREINC\Adminis...
RPC:Completed	exec sso_RetrieveApplication N'Nort...	Windows Shar...	Administrator	LITWAREINC\Adminis...
RPC:Completed	exec sp_reset_connection	Windows Shar...	Administrator	LITWAREINC\Adminis...
RPC:Completed	exec sso_InsertAudit @UserAuthority...	Windows Shar...	Administrator	LITWAREINC\Adminis...
RPC:Completed	exec sp_reset_connection	Windows Shar...	Administrator	LITWAREINC\Adminis...
RPC:Completed	exec sso_RetrieveApplicationFields ...	Windows Shar...	Administrator	LITWAREINC\Adminis...
RPC:Completed	exec sp_reset_connection	Windows Shar...	Administrator	LITWAREINC\Adminis...
RPC:Completed	exec sso_InsertAudit @UserAuthority...	Windows Shar...	Administrator	LITWAREINC\Adminis...
Audit Login	-- network protocol: LPC set quote...	.Net SqlClie...	SSOAccount	LITWAREINC\SSOAccount
SQL:BatchStarting	SELECT [Products] . * FROM [Products]	.Net SqlClie...	SSOAccount	LITWAREINC\SSOAccount
SQL:BatchCompleted	SELECT [Products] . * FROM [Products]	.Net SqlClie...	SSOAccount	LITWAREINC\SSOAccount
Trace Pause				

Trace is paused.

Ln 29, Col 1