



QDABRA DATABASE ACCELERATOR V2.3

Create a SharePoint Data View that uses GetListItems to access DBXL forms

SharePoint Designer allows you to receive data from DBXL's web methods, and one method is particularly interesting when creating a SharePoint Data View. The DBXLDocumentService web service contains GetListItems, which retrieves promoted properties in the form and displays them as columns in SharePoint. GetListItems has an added bonus: it will respect DBXL permissions.

PRE-REQUISITES

To successfully complete this tutorial, you **MUST** have the following:

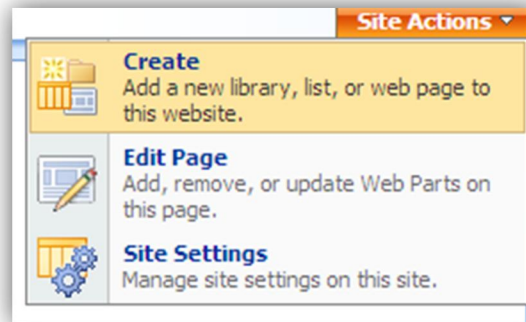
- A form template with promoted properties.
- This form template is published as a Document Type in DBXL, and documents have already been added.
- Microsoft Office SharePoint Designer 2007.

RELATED DOCUMENTATION

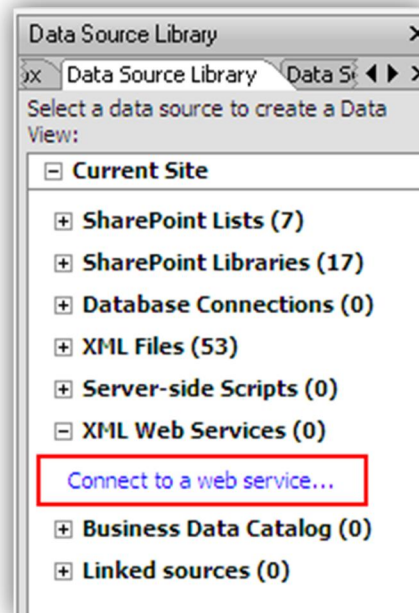
- Another option involves first creating a Database (SQL) Mapping for your DBXL Document Type, then configuring your data view to retrieve data from SQL. Please read the document called [Create a SQL-based SharePoint Data View to access your DBXL Forms](#) for information on this alternative method.
- If you require DBXL Document-Level permissions, please read [Use Document-Level Permissions for browser-based forms in a data view](#).

TUTORIAL

- A. Create a new SharePoint List.
 1. Navigate to the site where you wish to add this data. Click on **Site Actions** and then click on **Create**.



2. Under **Custom Lists**, click on **Custom List**.
 3. Enter a **Name**, a **Description**, and select whether the list will show in the Quick Launch. When done, click **Create**.
- B. Modify the list using SharePoint Designer.
4. Open this newly created SharePoint page in SharePoint Designer by selecting **File > Edit with SharePoint Designer** in Internet Explorer. You can also directly open it by clicking **Open** icon in SharePoint Designer and entering your SharePoint library URL.
 5. Click on **Data View > Manage Data Sources**.
 6. Under **XML Web Services**, click **Connect to a web service**.



7. Enter the **Service Description Location** (URL) to the DBXLDocumentService, usually located at <http://<servername>/QdabraWebService/DbxlDocumentService.aspx>, and click **Connect Now**.



Data Source Properties

General Source Login

Service description location:
http://<servername>/QdabraWebService/DbxIDocumentService.asmx Browse...

Example: http://example.com?WSDL

Connect Now Establish a connection with the specified XML Web Service.

Select which data command to configure: Select

8. In the Operation dropdown, select **GetMyListItems**. (You can use **GetListItems** instead to list all documents, if permissions are not a part of your solution.)
9. Double-click on the **listName** and **viewName** parameters, and enter the name of the Document Type you wish to query. Click **OK**.

Data Source Properties

General Source Login

Service description location:
http://<servername>/QdabraWebService/DbxIDocumentService.asmx?WSDL Browse...

Example: http://example.com?WSDL

Disconnect

Select which data command to configure: Select

Select Connection Info

Service Name: DbxIDocumentService

Port: DbxIDocumentServiceSoap

Address: http://<servername>/QdabraWebService/DbxIDocumentService.asmx

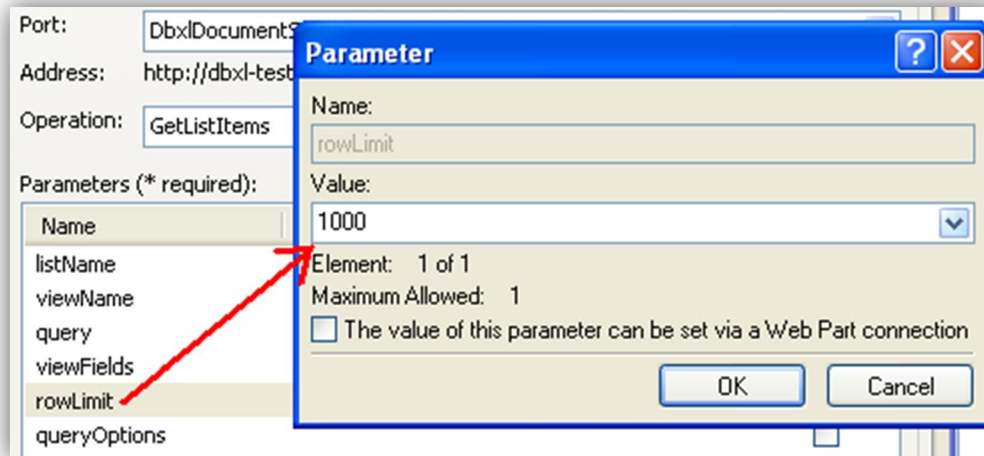
Operation: GetListItems

Parameters (* required):

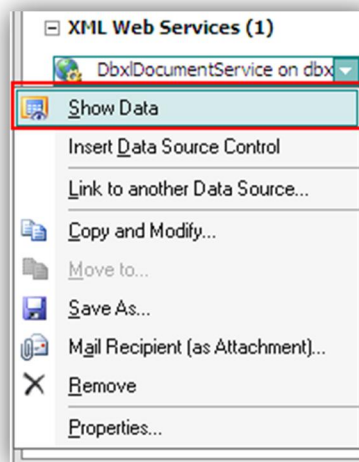
Name	Value	Element	Type	Runtim...
listName	<u>MyDocType</u>		s:string	☐
<u>viewName</u>	<u>MyDocType</u>		s:string	☐
query				☐
viewFields				☐
rowLimit			s:string	☐
queryOptions				☐



GetListItems limits the results returned in your SharePoint list to the first 100 documents. In order to increase this, set the rowLimit to more than 100 rows.



10. The new web service connection will show up under XML Web Services in the Data Source Library. Click the down arrow for the connection you just created and click on **Show Data**.



The Data Source Details tab will show the columns retrieved from the Document Type. You can design the page as desired. For this example, we will do the following:

11. Hold down the Control key and use your mouse to click on the DocID, Name, Author and Description (along with any other columns you wish to display). Then drag and drop these columns into the location on the page where you wish to display this data.



C. IPFS-enable the Data View (*Optional*).

The steps in this section (C) are only necessary if your form template has code, needs to be browser-enabled, and has already been admin-deployed. If this is not necessary, do the following to add a hyperlink:

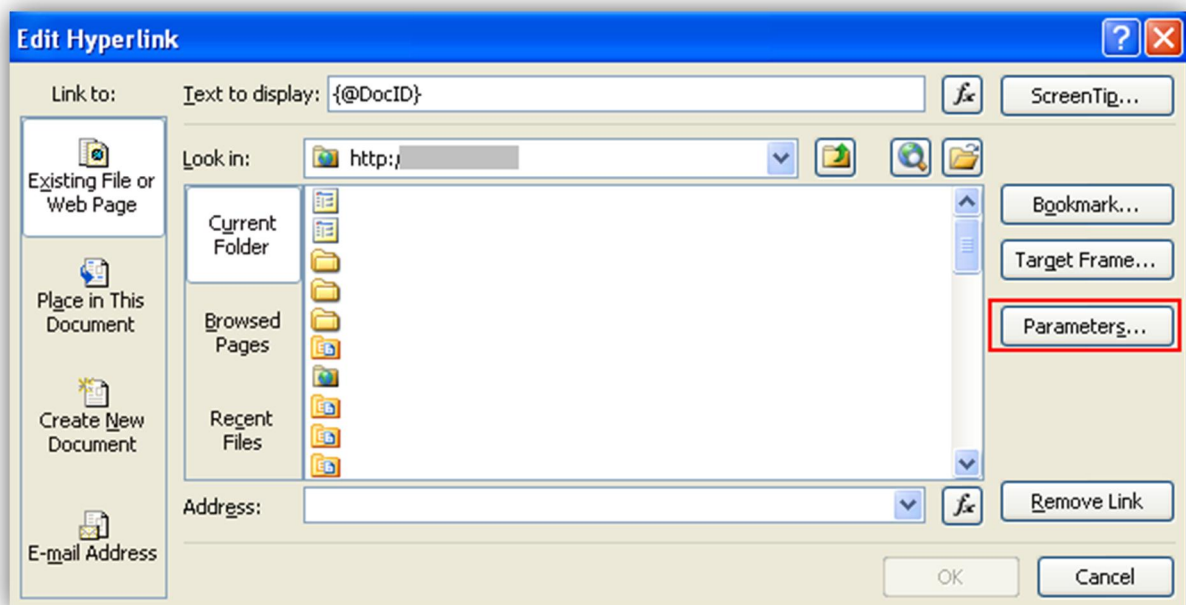
- Right-click on any entry in the DocID column and click **Hyperlink**.
- Enter **{@Link}** in the Address field and click **OK**.

12. Right click on any entry in the DocID column and click **Hyperlink**. We will now create a hyperlink to IPFS-enable the forms.

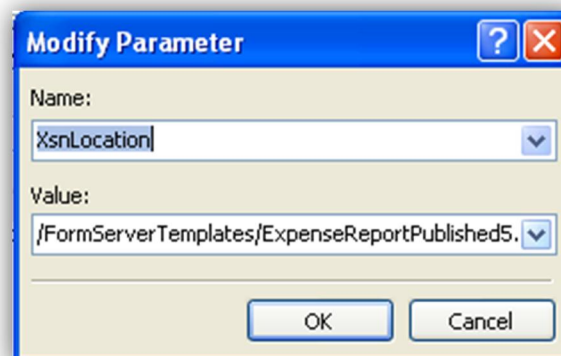


DocID	Author	Name	Description
6	jo-anne	Expense Report	test1

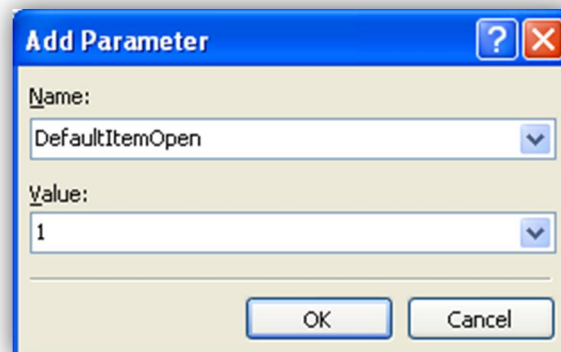
13. Click on the **Parameters** button.



14. In the Path text field, enter `http://<your server name>/_layouts/FormServer.aspx`
15. Click **Add** button. Enter XsnLocation in the Name field.
16. Enter value `/FormServerTemplates/<your published xsn name>.xsn`, and then click **OK**.

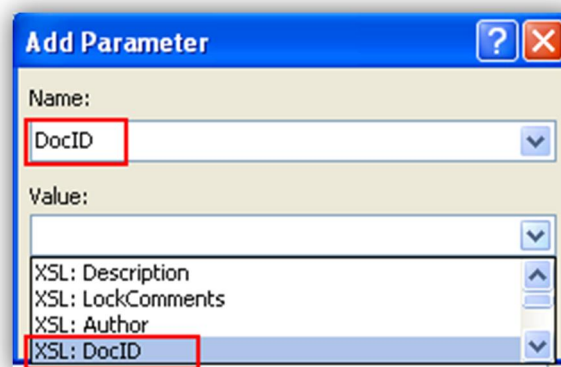


17. Click **Add** button.
18. Enter DefaultItemOpen in Name, and Value equal to 1. Click **OK**.



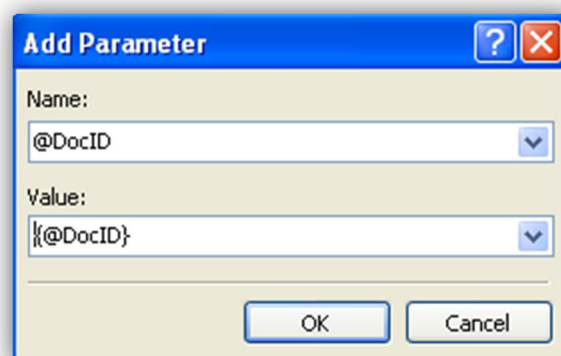
The 'Add Parameter' dialog box has a blue title bar with a question mark and a close button. It contains two sections: 'Name:' with a dropdown menu showing 'DefaultItemOpen', and 'Value:' with a dropdown menu showing '1'. At the bottom are 'OK' and 'Cancel' buttons.

19. Click **Add** button again.
20. Enter DocID and select **XSL:DocID** in the Value dropdown.



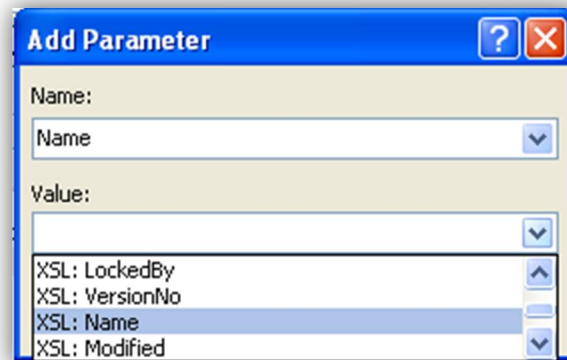
The 'Add Parameter' dialog box shows 'DocID' in the Name field. The Value dropdown is open, showing a list of XSL paths: 'XSL: Description', 'XSL: LockComments', 'XSL: Author', and 'XSL: DocID'. The 'XSL: DocID' option is selected and highlighted with a red box.

21. You will get something like this:



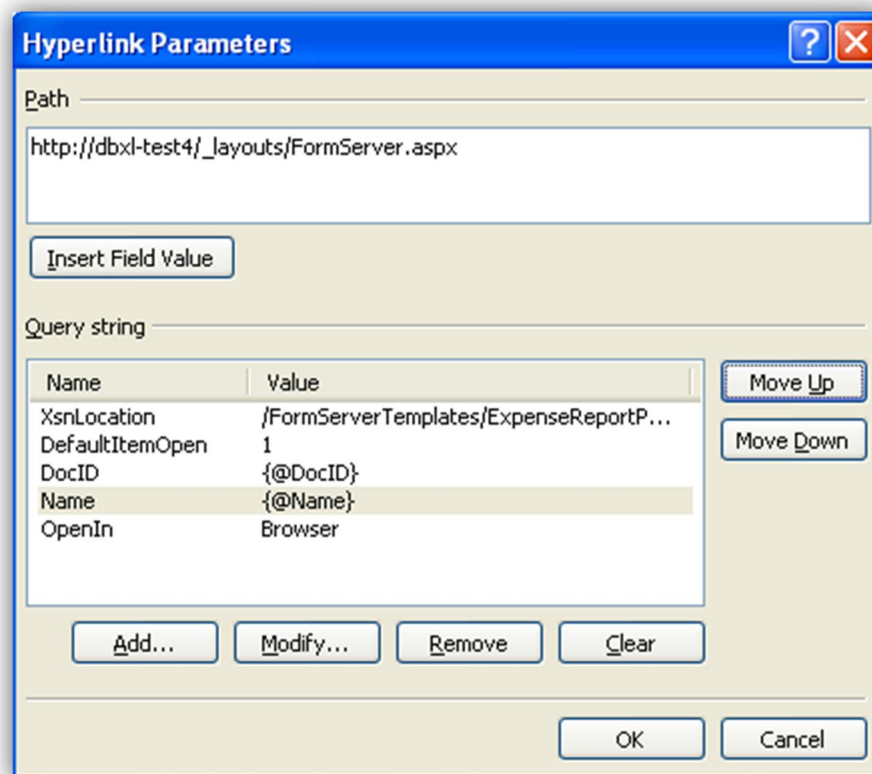
The 'Add Parameter' dialog box shows '@DocID' in the Name field and '{{@DocID}}' in the Value field. The 'OK' and 'Cancel' buttons are at the bottom.

22. Remove the '@' of DocID in the Name field and click **OK**.
23. Repeat steps 19 –22, this time using **Name**:



The 'Add Parameter' dialog box has a title bar with a question mark and a close button. It contains two main sections: 'Name:' and 'Value:'. The 'Name:' section has a dropdown menu currently showing 'Name'. The 'Value:' section has a dropdown menu currently showing 'XSL: Name', which is highlighted. Below these are four other options in the 'Value:' dropdown: 'XSL: LockedBy', 'XSL: VersionNo', and 'XSL: Modified'.

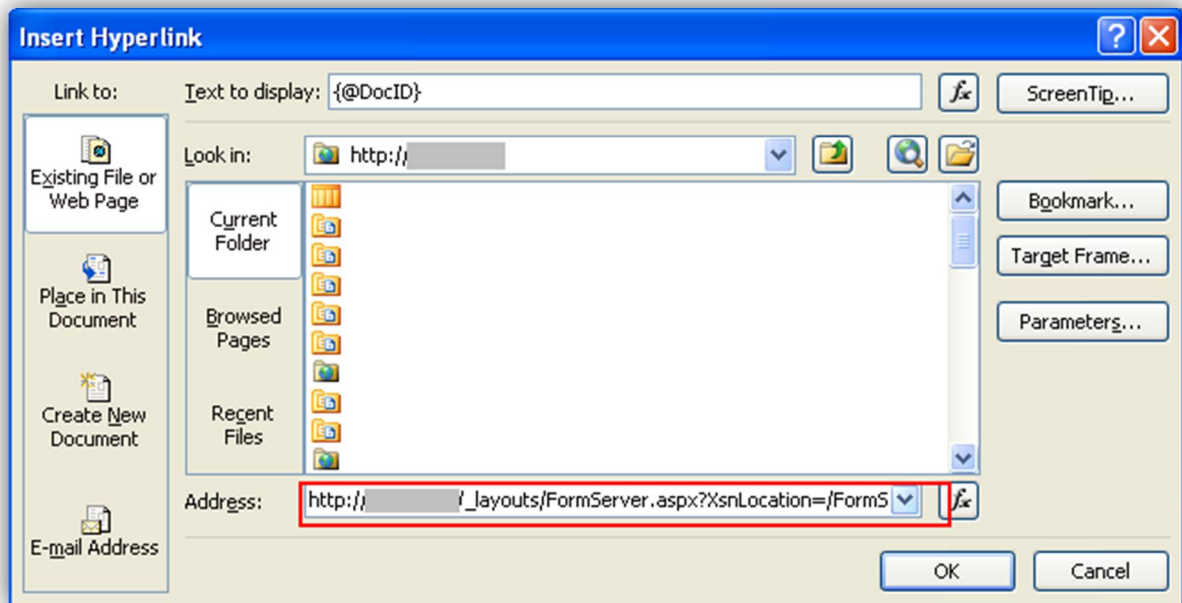
24. Click the **Add** button. Enter **OpenIn** for **Name** and **Browser** as **Value**. Then click **OK**.
25. You will now have your parameters listed as follows:



The 'Hyperlink Parameters' dialog box has a title bar with a question mark and a close button. It contains a 'Path' field with the text 'http://dbxl-test4/_layouts/FormServer.aspx'. Below the path field is an 'Insert Field Value' button. Below that is a 'Query string' section containing a table with two columns: 'Name' and 'Value'. The table has five rows: 'XsnLocation' with value '/FormServerTemplates/ExpenseReportP...', 'DefaultItemOpen' with value '1', 'DocID' with value '{@DocID}', 'Name' with value '{@Name}', and 'OpenIn' with value 'Browser'. To the right of the table are 'Move Up' and 'Move Down' buttons. At the bottom of the dialog are four buttons: 'Add...', 'Modify...', 'Remove', and 'Clear'. At the very bottom are 'OK' and 'Cancel' buttons.

Name	Value
XsnLocation	/FormServerTemplates/ExpenseReportP...
DefaultItemOpen	1
DocID	{@DocID}
Name	{@Name}
OpenIn	Browser

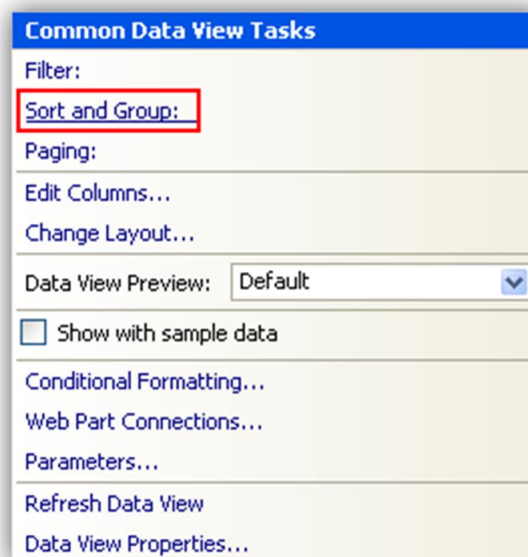
26. Click **OK**.
27. Verify that the link generated in the Address bar is correct. The address should be:
http://<yourservername>/_layouts/FormServer.aspx?XsnLocation=/FormServerTemplates/<templatepublished>.xsn&DefaultItemOpen=1&DocID={@DocID}&Name={@Name}&OpenIn=Browser



28. Click **OK**.

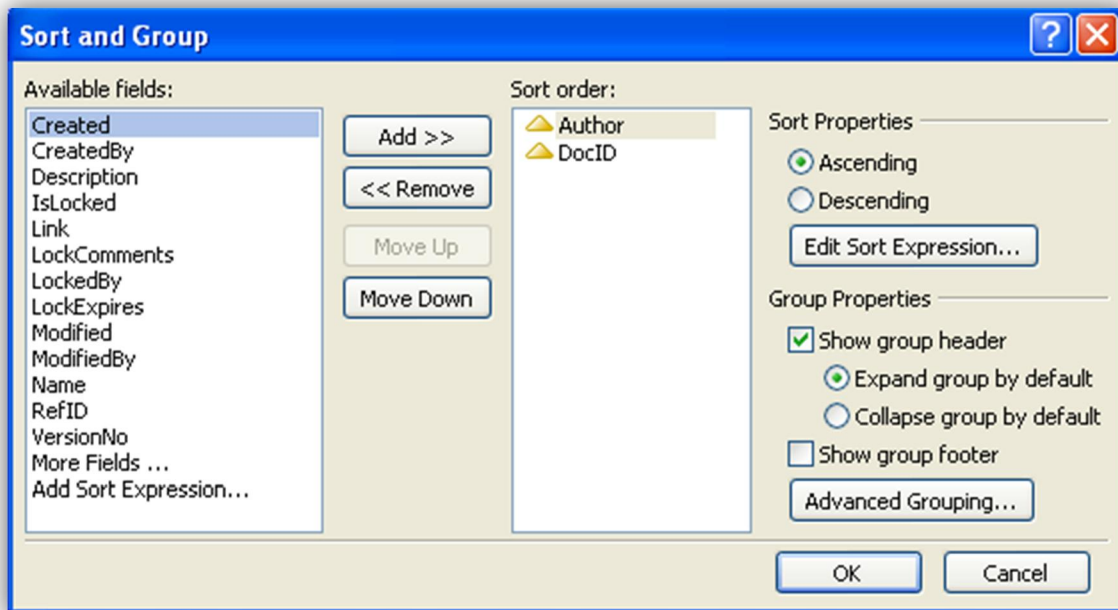
D. Data View sorting and grouping (*Optional*)

29. Click on the arrow to the right of the table, called **Common Data View Tasks**, and click on **Sort and Group**.



30. Select **DocID** and click **Add >>**. Repeat for **Author**.

31. Click **Author** in **Sort order**, click **Move Up**, and then click **Show group header**. Click **OK**.

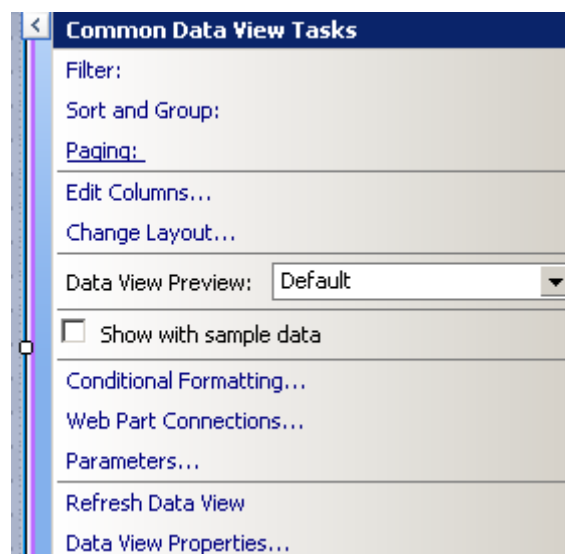


32. Click **File > Save** to save the changes to your SharePoint List.
33. In Internet Explorer, open the URL to your SharePoint List

E. Paging (*Optional*)

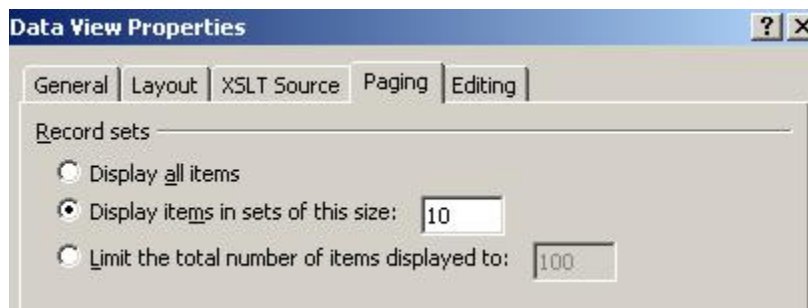
Should your document type contain a large amount of documents, you can use paging to improve performance. For more info, [see this page](#).

34. Click on the arrow to the right of the table and select Paging.





35. Select the **Paging** tab, then select the option to **Display items in sets of this size**. Enter the number of items you wish to display per page.



36. Click **File > Save** to save the changes to your SharePoint List.
37. In Internet Explorer, open the URL to your SharePoint List to view the change.

UPGRADING FROM PREVIOUS VERSION OF DBXL

In DBXL v2.3, the encoding of promoted properties has changed. This affects users of SharePoint web parts or data views that make use of promoted properties that contain spaces. Therefore, all data views using GetMyListItems or GetListItems that have promoted properties containing spaces will no longer have the _x005F in the name. For example, 'Status Code', which was 'Status_x005F_x0020_Code' is now 'Status_x0020_Code'. If you upgrade from a previous version of DBXL to DBXL v2.3, you will need to update your data view.