



QDABRA DATABASE ACCELERATOR

DBXL Permissions when using SharePoint form libraries

Conditional SharePoint mappings in DBXL allow you to use hardcoded form libraries as a workaround for Document Type Active Directory group permissions. These steps will illustrate an example.

DOCUMENT-LEVEL PERMISSIONS USING A SHAREPOINT FORM LIBRARY

1. Modify the Expense Report such that it loads Active Directory usernames.
 - a. Open InfoPath, select **Design a Form Template** and click on **Customize a sample**.
 - b. Select **Sample – Expense Report** and click on **Design this form**.
 - c. Click **Tools** from menu, select **Data Connections** and click **Add**.
 - d. Select the radio button to **Create a New Connection** to and **Receive Data**. Click **Next**.
 - e. Select the radio button for **Web Service**. Click **Next**.
 - f. Enter the URL of the service: <http://<servername>/QdabraWebService/ADUserInfo.asmx>. Click **Next**.
 - g. Select the operation named *GetMyInfo*. Click **Next**.
 - h. There is no need to check the checkbox for **Store a copy of the data in the form template**. Click **Next**.
 - i. Click **Finish**, making sure that the **Automatically retrieve data when form is opened** is checked, and then click **Close**.
 - j. Double click on the **Name** field in the view.
 - k. In the **Default Value** section, click on the **fx** button to the right of the field.
 - l. Click **Insert Field or Group** and select the *GetMyInfo (Secondary)* data source from the dropdown.
 - m. Expand *myFields/dataFields/tns:GetMyInfoResponse/GetMyInfoResult/ADProp*.
 - n. Select the **Value** node, click **Filter Data**, and then click **Add**.
 - o. Select **Key** in the first drop down. Leave the second dropdown as **is equal to**. In the third dropdown select **Type text**, and then type the text *sAMAccountName*. Notice that InfoPath adds double quotes after you type and tab out.
 - p. Click **OK** five times to close the dialogs.
2. Verify that your form loads the Active Directory data.
 - a. Select **File > Save**. For the purposes of this illustration, save the form as *ExpenseReport2.xsn*.
 - b. Click **Preview**.
 - c. Verify that the fields are auto-populating correctly.
 - d. Click on **Close Preview**.
3. Modify the Expense Report form such that it submits to DBXL
 - a. Double click on the **Submit** button to display properties.
 - b. Click on **Submit Options**.
 - c. Select **Web Service** from the first dropdown.
 - d. Click the **Add** button.
 - e. Type in the location of the Qdabra DBXL Document Web Service in the format <http://<servername>/QdabraWebService/DbxlDocumentService.asmx> and click **Next**.
 - f. In the list of web method operations, choose **SubmitDocument**. Click **Next**.
 - g. Ignore the first parameter, called *tns:docTypeName*, because it will default to the DocType of the form. Select the second parameter, named *tns:xml*. Under **Parameter options** select



- the **Entire form (XML document, including processing instructions)** radio button. Check the **Submit data as a string** checkbox. Click **Next** and then click **Finish**.
- h. Click OK twice to close the dialogs.
 - i. Save the updated form by clicking **Save** on the **File** menu. Close the form.
4. Create a new Document Type
 - a. Open DAT by clicking on the **DBXL Administration Tool** link in the <http://<servername>/QdabraWebService/default.htm> page.
 - b. Attach the form template and name the configuration.
 - c. Click **Save**, and then click **OK** in the confirmation dialog.
 5. Verify Submit works.
 - a. In the General tab, click **Open**.
 - b. Fill out the form. When done, click the **Submit** button.
 - c. Click the **Documents** tab in DAT to verify that the new document has been added.
 6. Add a SharePoint mapping
 - a. Switch to the SharePoint tab in DAT and click on **Insert Mapping**.
 - b. Enter the URL for the new SharePoint library, such as <http://<servername>/ExpenseReport>, check the **Publish Library and XSN** checkbox. Click **Save** and then **OK** to close the confirmation dialog.
 7. Verify the SharePoint mapping
 - a. In the General tab, click **Open**.
 - b. Fill out the form. When done, click the **Submit** button.
 - c. Click the **Documents** tab in DAT to verify that the new document has been added.
 - d. In Internet Explorer, open the SharePoint form library and verify that the new document was added.
 8. Enforce SharePoint “permissions” using conditional mapping
 - a. In the **SharePoint** tab in DAT, click on **Insert Expression**.
 - b. Enter `/my:expenseReport/my:employee/my:name='<youralias>'`
 - c. Click **Save**, and then click **OK** in the confirmation dialog.
 9. Enforce DBXL permissions
 - a. In the **Permissions** tab in DAT, click on Insert document level permissions (the second table).
 - b. Enter `/my:expenseReport/my:employee/my:name` for the Membership Xpath and give it a name.
 - c. Check the checkboxes for **Write**.
 - d. Check the **Enforce Permissions** checkbox.
 - e. Click **Save**, and then click **OK** in the confirmation dialog.
 10. Verify the enforcement of permissions
 - a. Submit two documents, one with your alias in the Name field and one with someone else’s alias.
 - b. Verify documents show up in Documents tab in DAT.
 - c. Refresh the SharePoint form library to confirm one document didn’t show.

UNDERSTANDING THE RESULTS

You will be able to open the document that contains your useralias, whether you access it from DAT (using the DBXL document handler URL, <http://servername/QdabraWebService/Documents/<mydoc>/<mydoc>.xml>) or access it via the SharePoint



form library (*<http://servername/ExpenseReport/<notmydoc>.xml>*). The document that was not assigned to you, however, will not open.

Note that, if you are the DBXLAdmin, you will be able to open any document. Remove those permissions in order to test this solution.

CREATE A SHAREPOINT DATA VIEW TO ACCESS DBXL DOCUMENTS

SharePoint Designer allows you to receive data from DBXL's web methods, and one method, GetListItems, is particularly interesting when creating a SharePoint Data View because it respects DBXL permissions. To access that tutorial, please [click here](#).