Getting Started with Database Accelerator (DBXL) v2.5

This Getting Started guide is an introductory tutorial that will show you how to quickly familiarize yourself with some of the features of DBXL v2.5 and to setup the Expense Report form sample form in DBXL on your own. After completing this document you should be able to setup your own forms in DBXL and map to SQL.

This document assumes that you have already installed DBXL v2.5 (Server or Client). If you have not installed DBXL 2.5 yet please see the DBXL v2.5 Installation Guide for more details.

This tutorial uses the Expense Report sample form that is included in all InfoPath 2007 installations. For your convenience we have packaged the form with this Getting Started document, available for download on InfoPathDev.com. Although the Expense Report sample form was designed for InfoPath 2007 it will work with either InfoPath 2007 or InfoPath 2010. Instructions for both InfoPath 2010 and InfoPath 2007 are included below.

After you complete this tutorial you might have follow up questions; make sure to visit the DBXL v2.5 Product page on Qdabra.com to find a list of all available documents.

Note: DBXL v2.5 includes an Expense Report form that is installed and configured. The form is mapped to SQL and upon install the necessary tables were setup in your SQL server DB. You can use this form as-is and integrate with SharePoint right away and generate Excel or SSRS reports from the installed form. The sample document type is called **QdExpenseReport-Sample** (see Appendix for more information). Once you are finished with the tutorial below, you should end up with a solution similar to the QdExpenseReport-Sample included in DBXL.

CONTENTS

Task 1: Design the form 2

InfoPath 2010 2

Active Directory 2

Submit to DBXL 6

InfoPath 2007 8

Active Directory 9

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

Submit to DBXL 12

Task 2: Create a DBXL Document Type 15

Task 3: Create a SQL table to store data 17

Task 4: Configure SQL Mapping 17

Task 5: Modify permissions 21

Add another DBXL Administrator 21

Add a DBXL user 21

Support 23

TASK 1: DESIGN THE FORM

In the first task we will design the InfoPath form by adding connections to the DBXL web service.

INFOPATH 2010

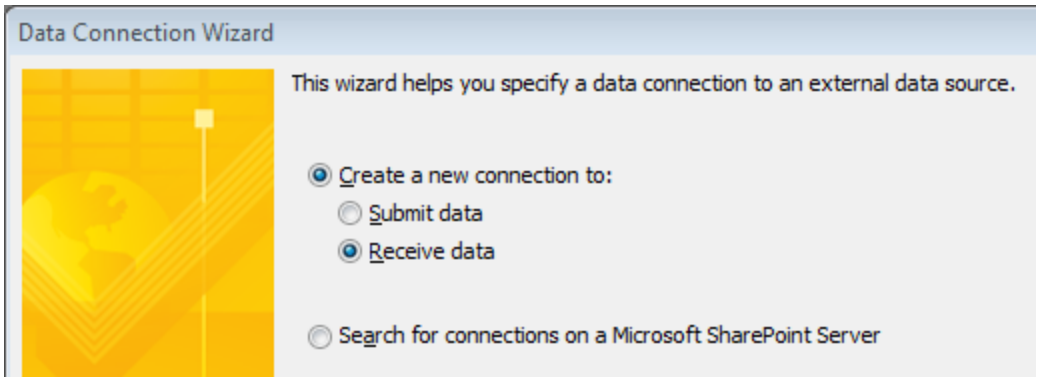
The following steps apply to users designing the form in InfoPath 2010.

ACTIVE DIRECTORY

Active Directory includes a wealth of information about your company's organization. This task will show you how to auto populate fields in your InfoPath form with the email address and full name of the currently logged on user.

1. Open the Expense Report form template. To do this, right click on the Expense Report sample XSN on your local drive, and select **Design**.
2. Add the Active Directory Data Connection to your form.
    a. In the InfoPath 2010 ribbon, select **Data** and then click on **Data Connections**.
    b. Click **Add**.
    c. Select the radio button to **Create a New Connection to** and **Receive Data**. Click **Next**.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



d. Select the radio button for **SOAP Web Service** and then click **Next**.

e. Enter the URL of the service: **http://<yourservername>/QdabraWebService/ADUserInfo.asmx**. Click **Next**.

f. Select the operation (Web Method) named **GetMyInfo**. Click **Next**.

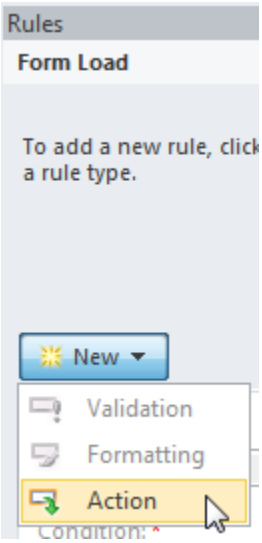
g. Click **Next**. There is no need to check the checkbox of **Store a copy of the data in the form template**.

h. Uncheck the **Automatically retrieve data when form is opened** checkbox, click **Finish**, and then click **Close**.

3. Change the **Employee Email** and **Name** fields to auto-populate.

a. In the Data tab of InfoPath 2010, click **Form Load**.

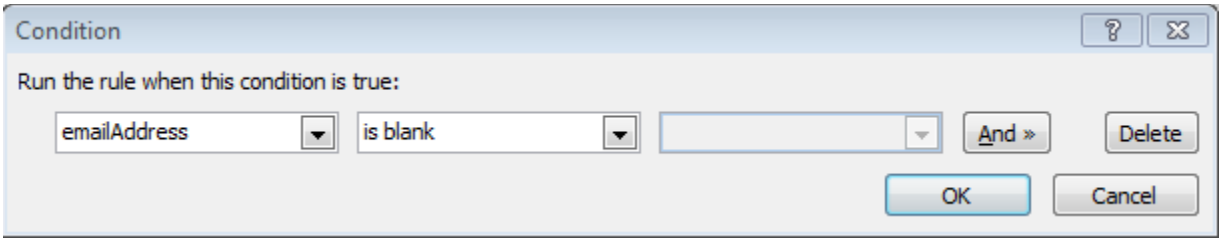
b. Click **New** and select **Action**.



c. Click the blue text under **Condition**.

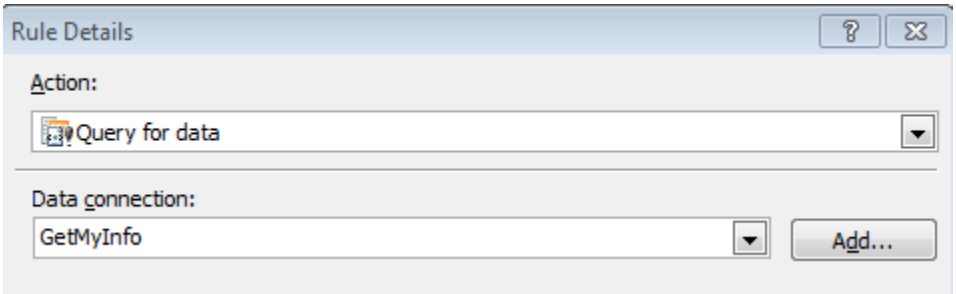
Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

d. We will enter a condition because we only want to load this data once, when the form is first created. Select "**Select a field or group**" from the first dropdown and select the **emailAddress** under the **Employee** node. In the middle dropdown, select "**Is blank**". The condition should look at shown below. Click **OK**.



e. Next to "**Run these actions**", click **Add** and select "**Query for data**".

f. Select the **GetMyInfo** data connection and click **OK**.



g. Next to "**Run these actions**", click **Add** and select "**Set a field's value**".

h. Click the icon to the right of "**Field**" and select the **emailAddress** node under the **Employee** node.

i. Click the icon to the right of "**Value**" and click **Insert field or group**.

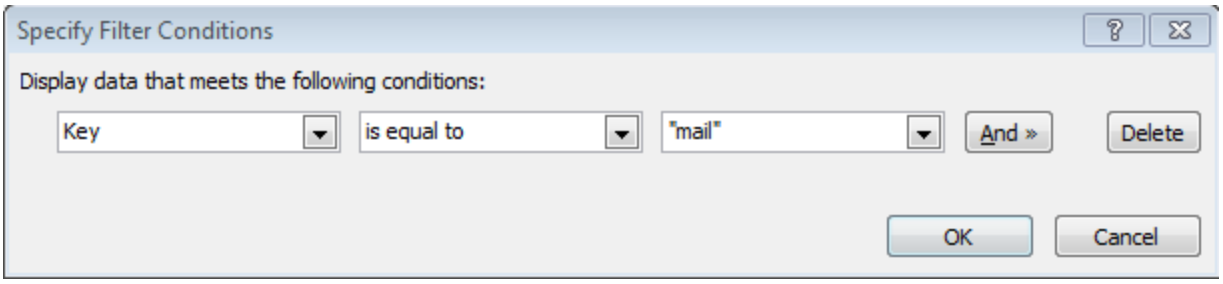
j. Use the dropdown to switch to the **GetMyInfo** secondary data connection.

k. Expand the **datafields** node, select the **Value** node, and click **Filter Data**.

l. Click **Add**. We need to add a filter because the Active Directory web service returns multiple keys and their values. At this point, we want only the email.

m. Select **Key** in the first dropdown and "**is equal to**" in the middle dropdown. In the third dropdown, select **Type text** and type in "**mail**" (no quotes).

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



n. Click OK until all dialogs are closed.

o. Now we'll set the employeee name field. Next to "**Run these actions**", click **Add** and select "**Set a field's value**".

p. Click the icon to the right of "**Field**" and select the **name** field under the **Employee** node.

q. Click the icon to the right of "**Value**" and click **Insert field or group**.

r. Use the dropdown to switch to the **GetMyInfo** secondary data connection.

s. Expand the **datafields** node, select the **Value** node, and click **Filter Data**.

t. Click **Add**. We need to add a filter because the Active Directory web service returns multiple keys and their values. At this point, we want only the email.

u. Select **Key** in the first dropdown and "**is equal to**" in the middle dropdown. In the third dropdown, select **Type text** and type in "**name**" (no quotes).

v. Switch to the **Home** tab in the InfoPath 2010 ribbon and click **Preview**. You will see the current user's data:

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

w. Click on **Close Preview**.

SUBMIT TO DBXL

1. Configure a Submit button to use the DBXL web service.

    a. Switch to the **Data** tab in the InfoPath 2010 ribbon and click on **Submit Options**.

    b. In the dropdown under "Send form data to a single destination", select **Web Service**.

    c. Click **Add**. This will load the Data Connection Wizard.

    d. In the web service field, enter http://<yourservername>/qdabrawebservice/DbxlDocumentService.asmx and click **Next**.

    e. Select **SubmitDocument** and click **Next**.

2. Configure the parameters.

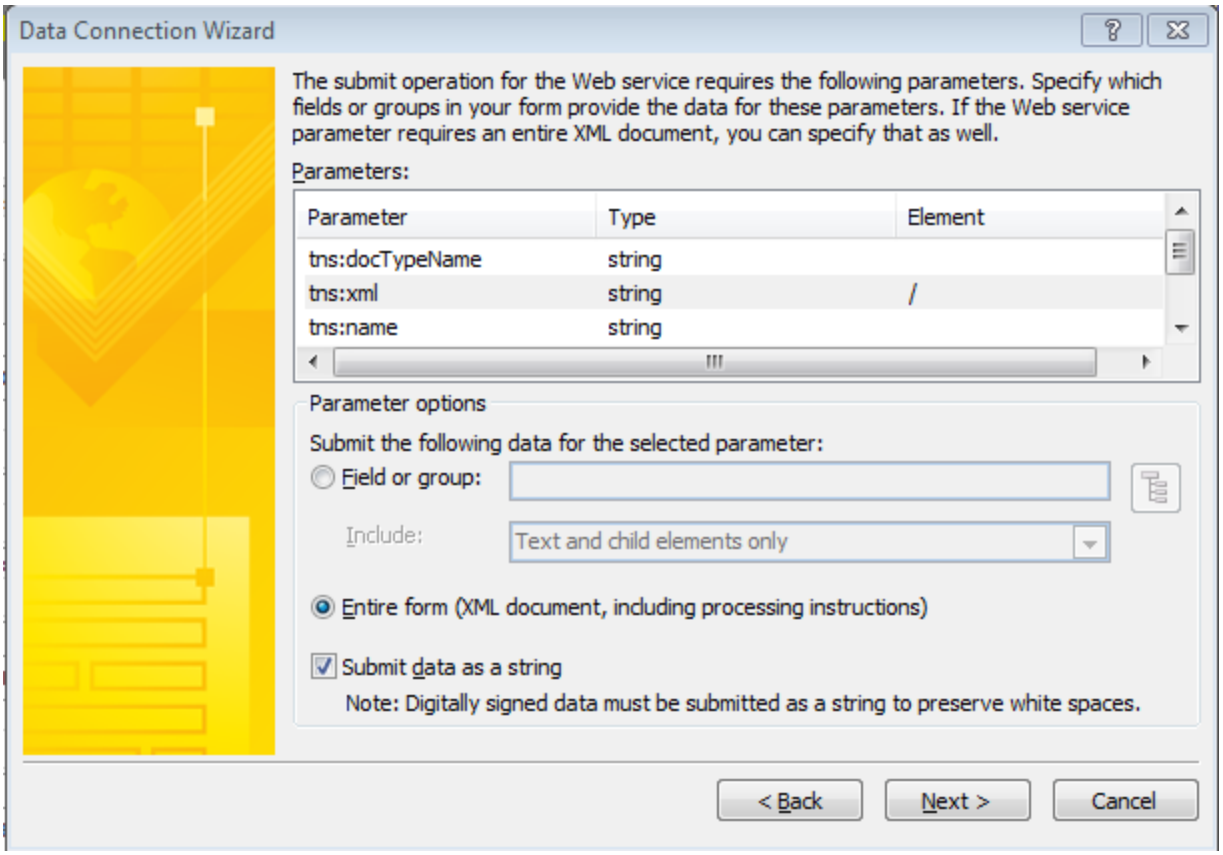
    a. Ignore the first parameter, called **tns:docTypeName**. It will default to the DocType of the form.

    b. Select the second parameter, named **tns:xml**.

        i. Under **Parameter options** select the **Entire form (XML document, including processing instructions)** radio button.

        ii. Check the **Submit data as a string** checkbox.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



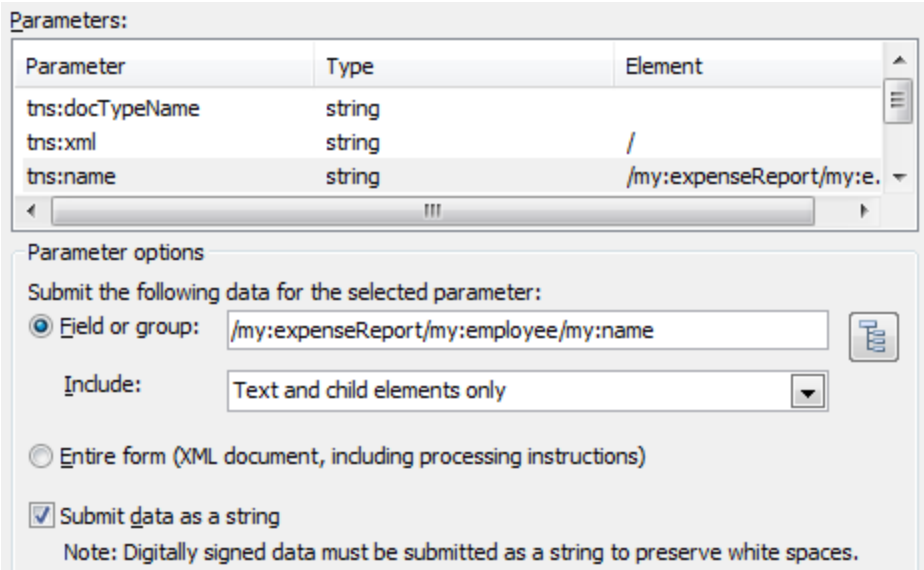
c. OPTIONAL: Select the **tns:name** parameter. You will notice that steps involved in subsections c, d and e has similar steps. This parameter, along with the Author and Description, are called the "DBXL tokens". These are displayed in the document list in the Documents tab in DAT. Setting a value for these is not required.

   i. Select the radio button **Field or Group** in the **Parameter options** section.

   ii. Click the Modify button next to the **Field or Group** text box. The **Select a Field or Group** dialog will appear.

   iii. Select the *expenseReport/employee/name* field, click **OK** and check the **Submit data as a string** checkbox.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



d. Select the **tns:author** parameter

   i. Select the radio button **Field or Group** in the **Parameter options** section

   ii. Click the Modify button next to the **Field or Group** text box. The **Select a Field or Group** dialog will appear.

   iii. Select the *expenseReport/employee/emailAddress*, click **OK** and check the **Submit data as a string** checkbox.

e. Select the **tns:description** parameter.

   i. Select the radio button **Field or Group** in the **Parameter options** section

   ii. Click the Modify button next to the **Field or Group** text box. The **Select a Field or Group** dialog will appear.

   iii. Select the *expenseReport/purpose*, click **OK** and check the **Submit data as a string** checkbox.

f. Click **Next**. Click **Finish**.

g. Back in the **Submit Options** dialog, enter **Submit** into the **Label** field.

h. Click Advanced. In the **After Submit** dropdown, select *Close the form*. Click **OK** twice.

3. Save the updated form by selecting **File** > **Save**. Then close the InfoPath 2010 designer.

INFOPATH 2007

These steps are for users designing the form in InfoPath 2007.

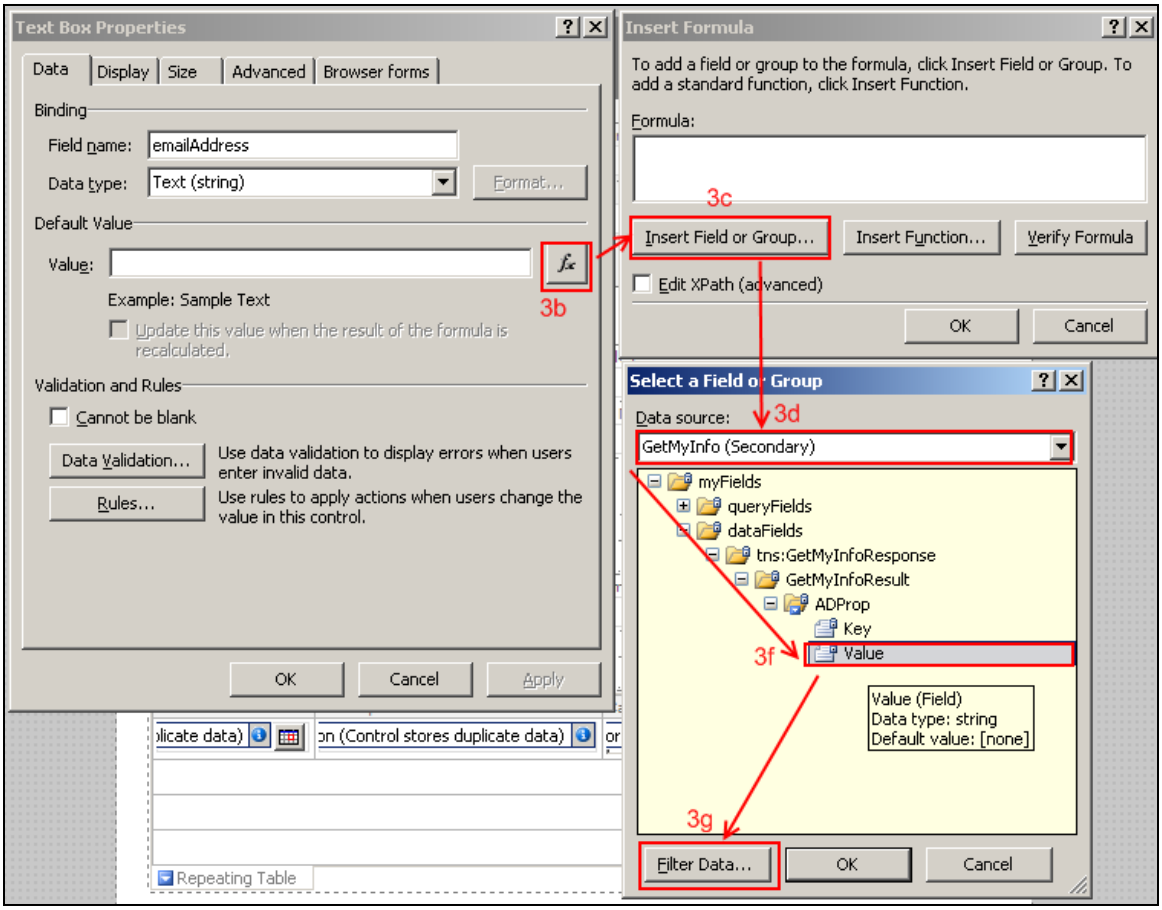
Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

ACTIVE DIRECTORY

Active Directory includes a wealth of information about your company's organization. This task will show you how to auto populate fields in your InfoPath form with the email address and full name of the currently logged on user.

1. Open the Expense Report form template. To do this, right click on the Expense Report sample XSN on your local drive, and select **Design**.
2. Add the Active Directory Data Connection to your form.
    a. Click **Tools** from and select **Data Connections**.
    b. Click **Add**.
    c. Select the radio button to **Create a New Connection to** and **Receive Data**. Click **Next**.
    d. Select the radio button for **Web Service**. Click **Next**.
    e. Enter the URL of the service: http://<yourservername>/QdabraWebService/ADUserInfo.asmx. Click **Next**.
    f. Select the operation (Web Method) named **GetMyInfo**. Click **Next**.
    g. Click **Next**. There is no need to check the checkbox of **Store a copy of the data in the form template**.
    h. Click **Finish**, making sure that the **Automatically retrieve data when form is opened** is checked, and then click **Close**.
3. Change the **Email Address** field to auto-populate.
    a. Double click on the Employee **E-mail Address** field in the view.
    b. In the **Default Value** section, click on the **fx** button to the right of the field.
    c. Click **Insert Field or Group.**
    d. Select the **GetMyInfo (Secondary)** data source from the dropdown
    e. Expand myFields/dataFields/tns:GetMyInfoResponse/GetMyInfoResult/ADProp
    f. Select the **Value** node.
    g. Click **Filter Data.**

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



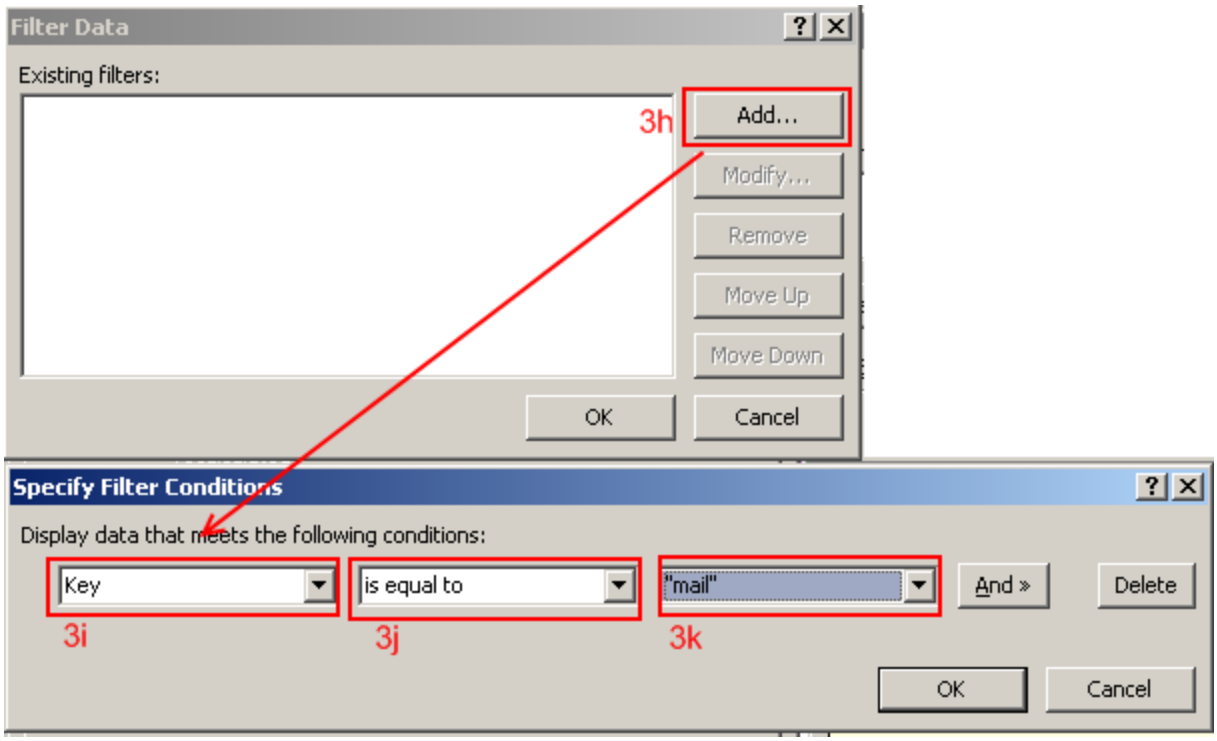
h. On the **Filter Data** dialog, click **Add.**

i. Select **Key** in the first drop down.

j. Leave the second dropdown as **is equal to**.

k. In the third dropdown, select the option **Type text**, then type **mail**. Notice that InfoPath adds double quotes after you type and tab out.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



l. Click **OK** five times to close the dialog boxes.

4. Change the Employee **Name** field to auto-populate.

   a. Double click on the Employee **Name** field in the view.

   b. In the **Default Value** section, click on the **fx** button to the right of the field.

   c. Click **Insert Field or Group**.

   d. Select the **GetMyInfo (Secondary)** data source from the dropdown.

   e. Expand myFields/dataFields/tns:GetMyInfoResponse/GetMyInfoResult/ADProp

   f. Select the **Value** node and click **Filter Data**.

   g. On the **Filter Data** dialog, click **Add.**

   h. Select **Key** in the first drop down.

   i. Leave the second dropdown as **is equal to**.

   j. In the third dropdown select **Type text**, and then type the text **cn**. Notice that InfoPath adds double quotes after you type and tab out.

   k. Click **OK** five times to close the dialogs.

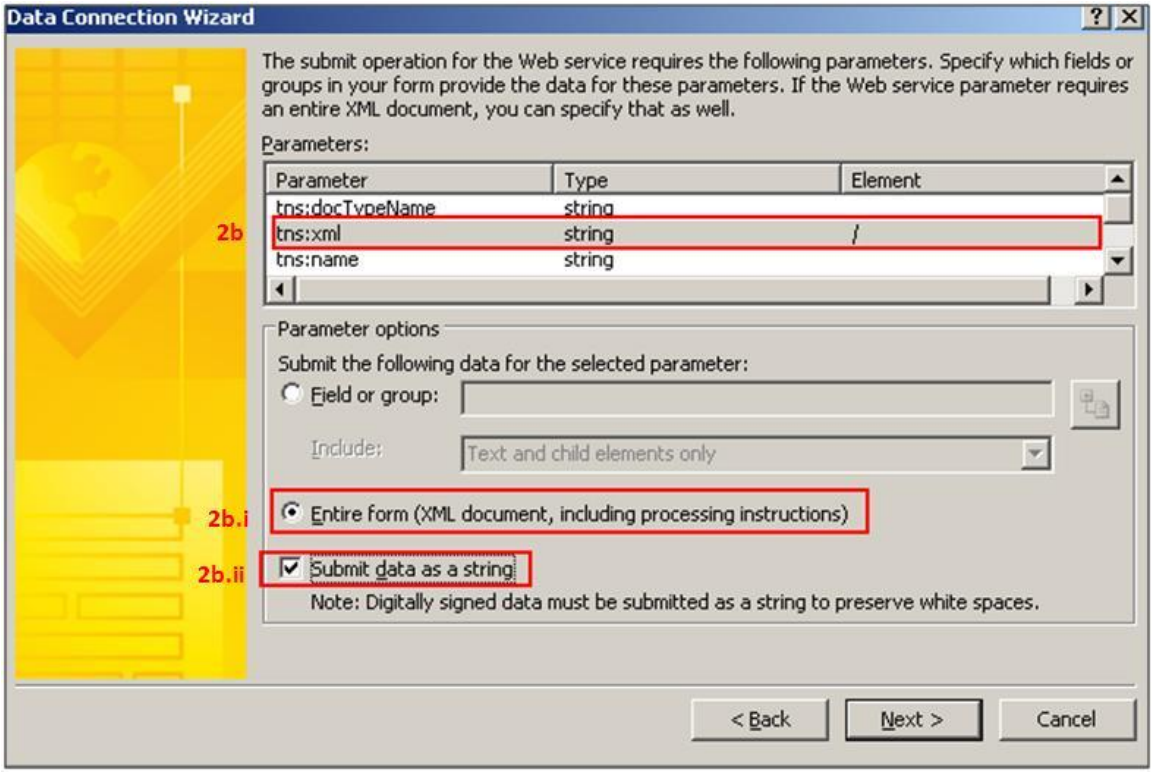
Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

5. Verify the results.
    a. Select **File** > **Save**, and click **Preview**. (Click **Yes** on the Microsoft Office InfoPath Security Notice)
    b. Verify that the fields are auto-populating correctly.
    c. Click on **Close Preview**.

SUBMIT TO DBXL

1. Configure a Submit button to use the DBXL web service.
    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 Web Service in the format http://<yourservername>/QdabraWebService/DbxlDocumentServiceEx.asmx.
    f. Click **Next**. InfoPath will contact the server.
    g. In the list of web method operations, choose the web method to use for submitting the form, which is called **SubmitDocument**.
    h. Click **Next**.
2. Configure the parameters.
    a. Ignore the first parameter, called **tns:docTypeName**. It will default to the DocType of the form.
    b. Select the second parameter, named **tns:xml**.
        i. Under **Parameter options** select the **Entire form (XML document, including processing instructions)** radio button.
        ii. Check the **Submit data as a string** checkbox.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



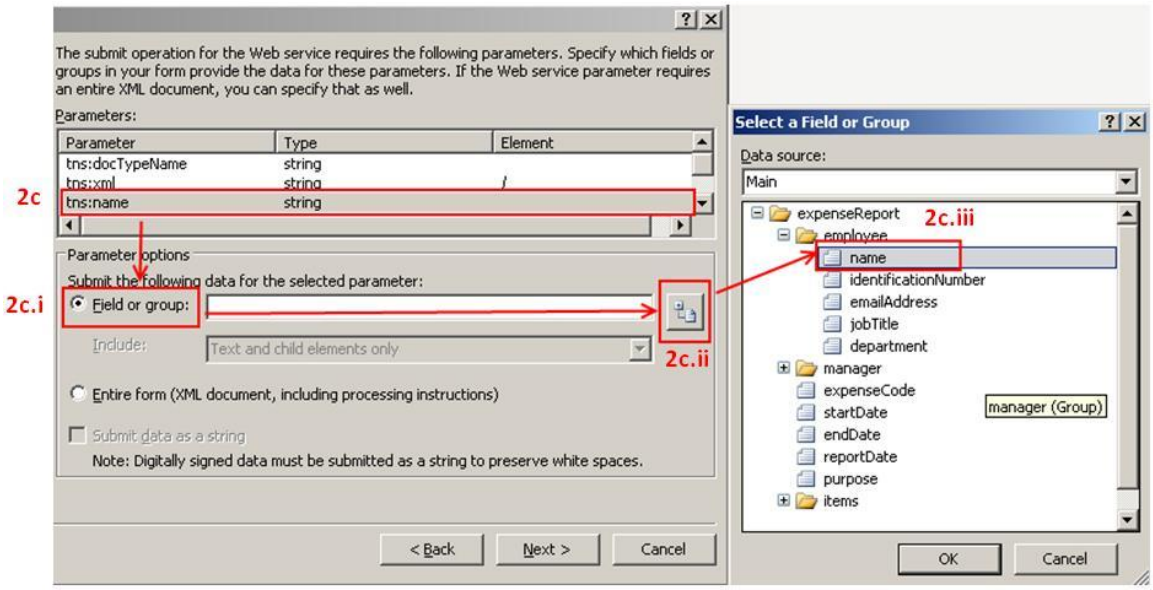
c. Select the **tns:name** parameter. You will notice that steps involved in subsections c, d and e has similar steps. This parameter, along with the Author and Description, are called the "DBXL tokens". These are displayed in the document list in the Documents tab in DAT. Setting a value for these is not required.

   i. Select the radio button **Field or Group** in the **Parameter options** section.

   ii. Click the Modify button next to the **Field or Group** text box. The **Select a Field or Group** dialog will appear.

   iii. Select the *expenseReport/employee/name* field, click **OK** and check the **Submit data as a string** checkbox.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



d. Select the **tns:author** parameter

   i. Select the radio button **Field or Group** in the **Parameter options** section

   ii. Click the Modify button next to the **Field or Group** text box. The **Select a Field or Group** dialog will appear.

   iii. Select the *expenseReport/employee/emailAddress*, click **OK** and check the **Submit data as a string** checkbox.

e. Select the **tns:description** parameter.

   i. Select the radio button **Field or Group** in the **Parameter options** section

   ii. Click the Modify button next to the **Field or Group** text box. The **Select a Field or Group** dialog will appear.

   iii. Select the *expenseReport/purpose*, click **OK** and check the **Submit data as a string** checkbox.

f. Click **Next**. Click **Finish**.

3. Configure the form to close upon submit

   a. Click **Advanced** in the **Submit Options** window.

   b. In the **After Submit** dropdown, select *Close the form*.

   c. Click **OK** twice.

4. Save the updated form.

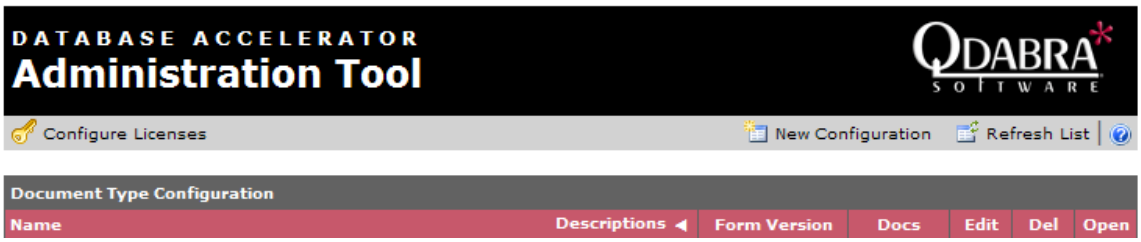
Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

a. Click **Save** on the **File** menu and save the form to a location on your computer. You will use this template in Task 2.

b. **Close** the form.

TASK 2: CREATE A DBXL DOCUMENT TYPE

In this section, we will use the DBXL Administration Tool (DAT). Access it through Internet Explorer by entering the URL ***http://<<yourservername>>/QdabraWebService/default.htm***. Once the page opens, click the DBXL Administration Tool hyperlink. InfoPath should open a form similar to the one below:



1. Create a new configuration in DAT.

   a. In DAT, click on **New Configuration**.

   b. Type **ExpenseReport** into the **Name** field.

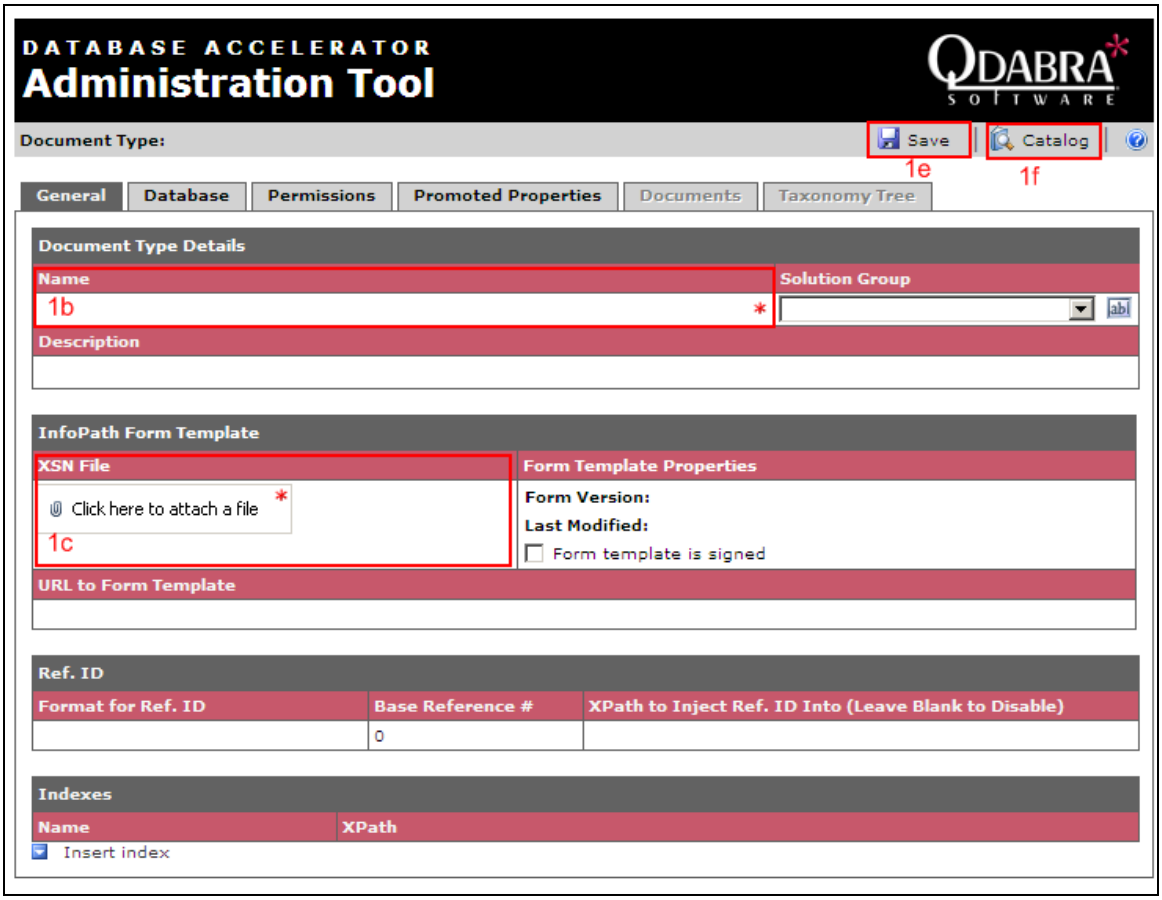
   c. Under **InfoPath Form Template**, click on **Click here to attach a file**.

   d. Select the Expense Report XSN and click **Insert**.

   e. Click **Save**. DBXL will confirm the operation through a dialog box. Click **OK** to close it.

   f. Click **Catalog**.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



2. Verify that you can open the form.

   a. In the **ExpenseReport** row click the InfoPath icon under **Open**. The form will open and you will be able to fill it out.

3. Verify Submit works.

   a. Fill out the form including the **Business Purpose**. **Employee Name** and **Employee Email Address** fields should be auto-populated. When done, click the **Submit** button. The form will automatically close.

   b. Back in the DAT Catalog, click **Refresh List**. You'll see that the number in the Docs column has increased

   c. You can also click Edit for the Document Type, and then click the **Documents** tab. The new document is added to the Documents table.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

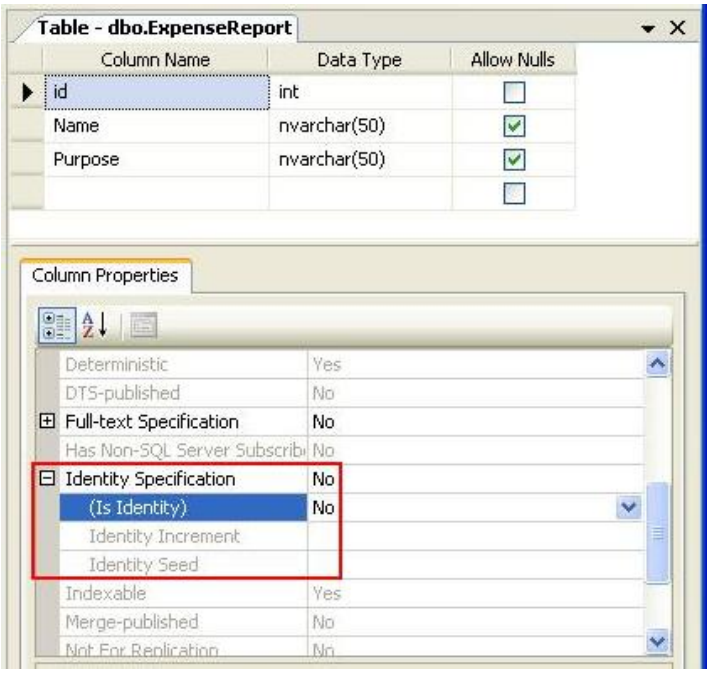
TASK 3: CREATE A SQL TABLE TO STORE DATA

The following steps assume that you have SQL Server Management Studio (Express or full version)

1. Design the database
    a. Open SQL Server Management Studio and connect to the SQL instance you wish to shred into.
    b. Create a database and a table to store your data. In this tutorial, we created a database named **DBXLTutorial** that contains the **ExpenseReport** table.
    c. To prevent shredding errors when trying to store information greater than the defined column length, set the appropriate columns to allow the maximum textbox length in InfoPath. For this exercise, use the following data type information for the columns needed:

    ID: int
    Name: nvarchar(50)
    Purpose: nvarchar(50)

    d. Open the **Column Properties** dialog box for the **ID** column, and scroll down until you find the **Identity Specification** section. Expand this section, and make sure **(Is Identity)** is marked as **No**.



    e. Make sure that the **Allow Nulls** option is checked for each of the columns, except ID, and then save the table.

TASK 4: CONFIGURE SQL MAPPING

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

Now that our table and form are ready, we can proceed to create the database mapping in the DBXL Administration Tool (DAT) to enable shredding.

1. Configure a database connection

    a. In DAT, click on the **Database** tab.

    b. Enter the Data Connection String appropriate for your SQL server. For example, *Integrated Security=SSPI;Data Source=(LOCAL)\sqlexpress;Initial Catalog=Your_Database*

    c. Tab out of the Connection String field to make the Test Connection button to appear. Click **Test Connection**, and click **OK** when success is confirmed.

    d. Click **Save**, and then click **OK** in the confirmation dialog.

2. Configure the database mapping

    a. Click **Insert Table or Column**.

    b. Click on the **Select Database Table** icon found next to the Table Name field. This will load the SQL schema in the Task Pane.

    c. In the Database Tree, double-click on the table into which you wish to save data, which we have assumed is called **ExpenseReport**.

    d. Click on the **Select Schema Node** icon found next to the Node Path field. This will load the form's XML schema in the Task Pane.

    e. Double-click on the *my:expenseReport* node.

    f. Click on the topmost **Insert Table or Column** link.

    g. From the **Type** dropdown, select **Column**.

    h. Click on the **Select Database Column** icon.

    i. In the Database Tree, double-click on the **Name** column.

    j. Click on the **Select Schema Node** icon and double click on the *my:employee/my:name* node.

    k. Click on the topmost **Insert Table or Column** link.

    l. From the **Type** dropdown, select **Column**.

    m. Click on the **Select Database Column** icon.

    n. In the Database Tree, double-click on the **Purpose** column.

    o. Click on the **Select Schema Node** icon and double click on the *my:purpose* node.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

p. Click on the topmost **Insert Table or Column** link.

q. From the **Type** dropdown, select **Column**.

r. Click on the **Select Database Column** icon.

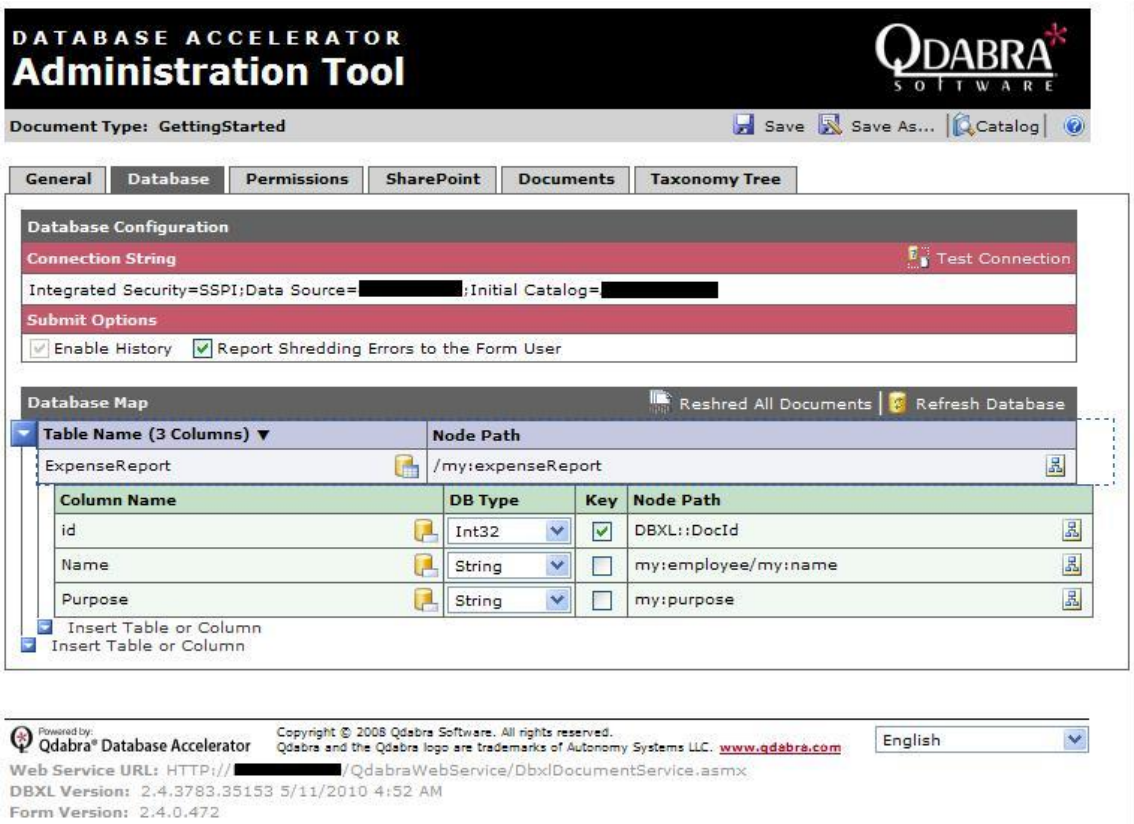
s. In the Database Tree, double-click on the **id** column.

t. Click on the **Select Schema Node** icon and double click on the *DBXL::DocID* node.

u. Check the **Key** checkbox.

v. Click **Save** in the DAT header, and then click **OK** in the confirmation dialog.

Your completed database mapping should look like this:



3. Test reshredding

a. Click **Reshred All Documents** and click **YES** on the warning dialog.

b. Click **OK** in the confirmation dialog.

c. Use SQL Server Management Studio to verify that a document (added in Task 4) has now been mapped to your table.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

4. Submit a document and verify the SQL table

    a. Click on the **General** tab.

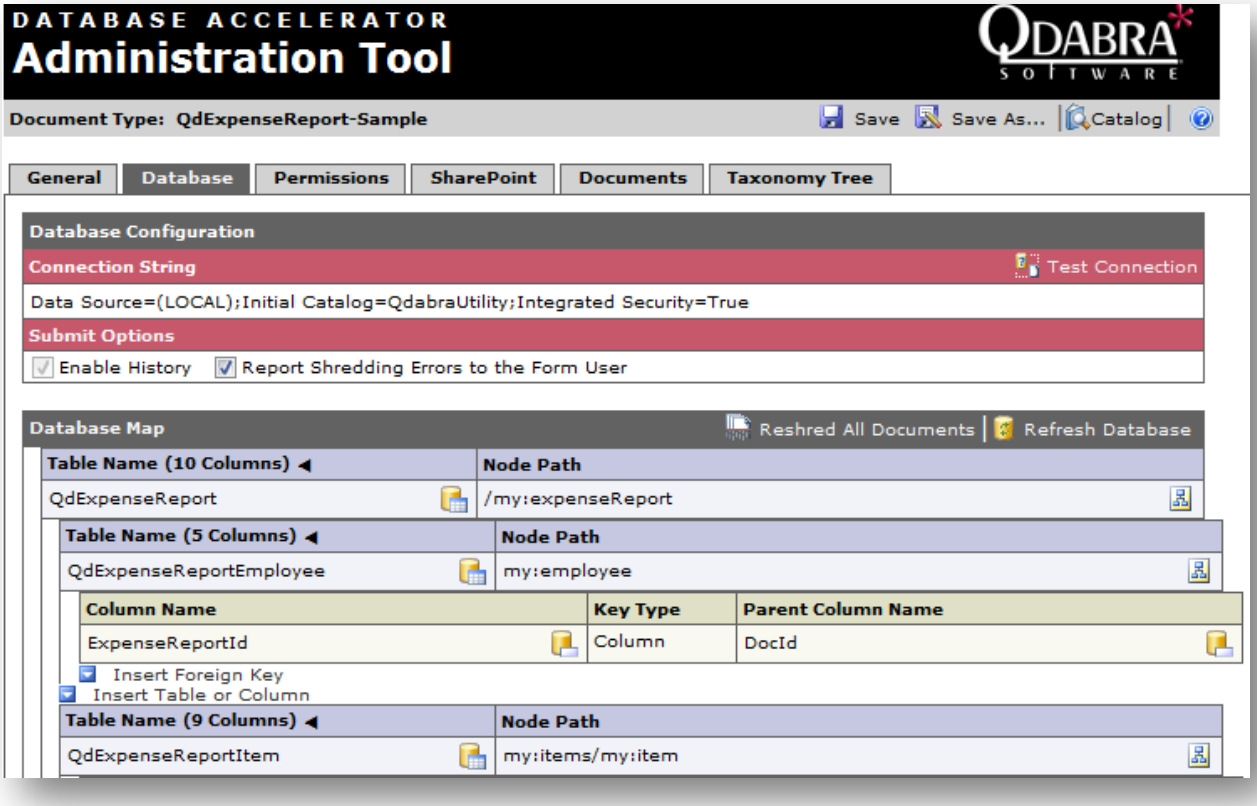
    b. Click **Open**.

    c. Fill out the form including the Business Purpose. Employee Name and Employee Email Address fields should be auto-populated. When done, click the **Submit** button.

    d. Click the **Documents** tab in DAT. The new document is added to the Documents column.

    e. Use SQL Server Management Studio to verify that the new document has been mapped to your table.

You will notice that DBXL comes with a Document Type called QdExpenseReport-Sample. In the DAT catalog view, click **Edit** for this Document Type, then switch to the Database tab. You'll see that the QdExpenseReport-Sample comes with a sample database mapping (screenshot below). Compare this to the database mapping you have implemented above to further your understanding.



Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

TASK 5: MODIFY PERMISSIONS

As the DBXL Administrator you can add permissions for other users. Here is how to edit these permissions:

1. In the Qdabrawebservice folder where DBXL was installed, find the web.config file.
2. Open the file (in a text editor, like Notepad) and look for the following: **ReaderGroupAlias**, **WriterGroupAlias**, **AdminGroupAlias** and **DbxlAdminAlias**.
3. Each of these Keys will be assigned a value (that is, a group).

ADD ANOTHER DBXL ADMINISTRATOR

During installation, DBXL creates a group called DBXLAdmins on the machine where DBXL was installed. That group, by default, is assigned to **AdminGroupAlias** and **DbxlAdminAlias**. To add a DBXL Administrator, simply add users to that group.

ADD A DBXL USER

ReaderGroupAlias and **WriterGroupAlias** are blank by default on new installations of DBXL.

1. On the machine where DBXL was installed, create a group called **DBXLUsers** and add users to this group.
2. Back in web.config, change **ReaderGroupAlias**, **WriterGroupAlias** to use ***<machinename>\DBXLUsers***.
3. Save web.config.

For additional information on permissions, refer to this document.

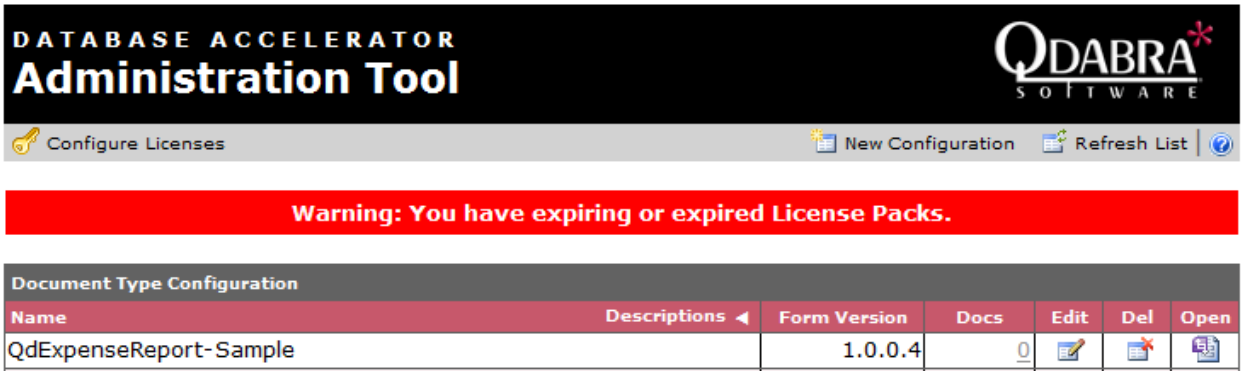
APPENDIX

In this Appendix we provide additional information about the **QdExpenseReport-Sample** document type that is included in DBXL v2.5.

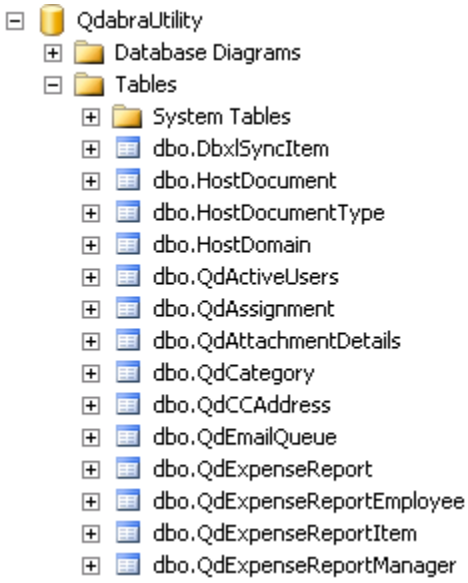
This document type includes:

- A data connection that submits the form to the DBXL web service. This is described within Task 1 in this document.
- The document type itself, of course, which is described in Task 2.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.

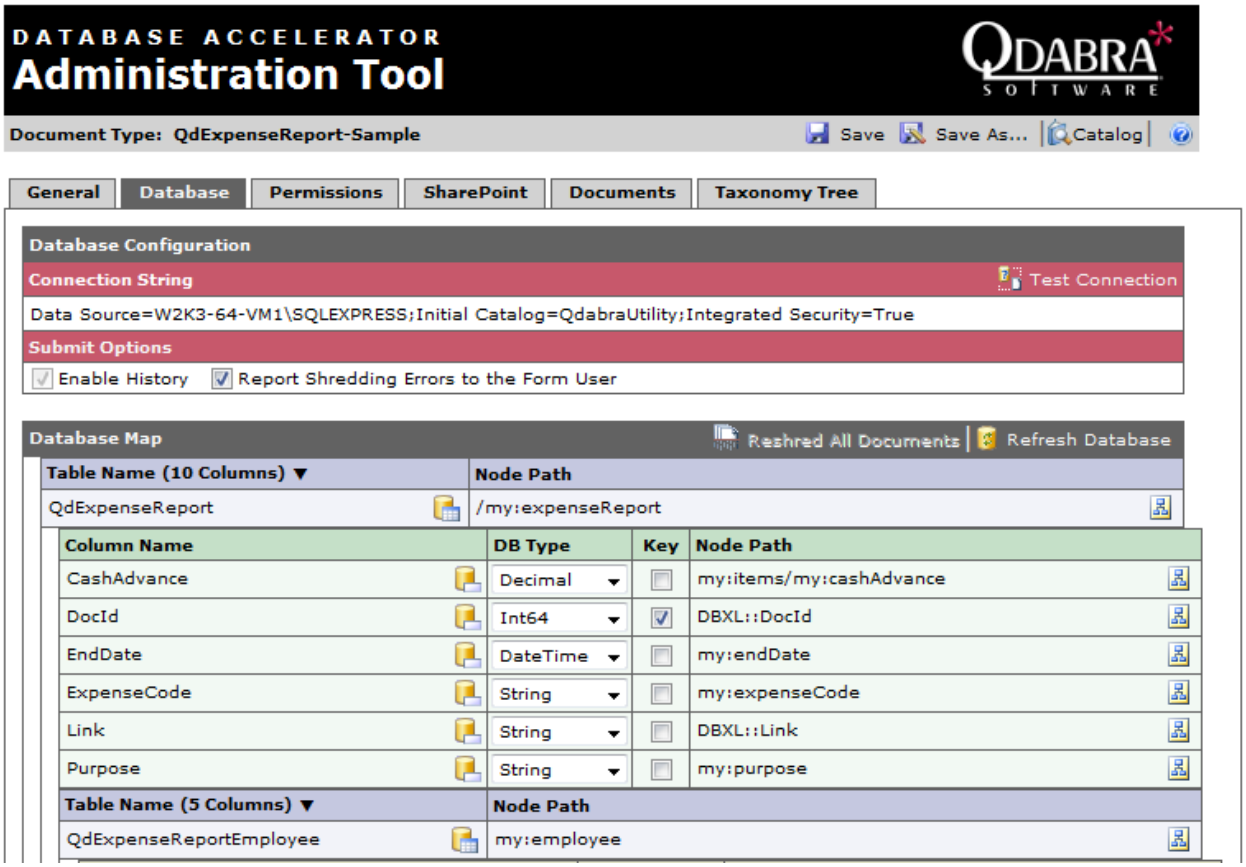


- SQL tables where the data from the Expense Report sample gets mapped. This is described in Task 3 above. The **QdExpenseReport** tables are located in the **QdabraUtility** database.



- Database (SQL) Mapping, which is described in Task 4.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.



You can use this database mapping as a reference when setting up your own. This Expense Report sample can also be extended to explore other DBXL features, such as SharePoint mapping or Active Directory.

SUPPORT

If you have questions about the information in this document, please contact us for assistance.

Licensed customers can contact us via **Support@Qdabra.com**. Alternatively, please use the InfoPathDev.com Qdabra Product support forums to request help.

Copyright © 2006-2011 Autonomy Systems, LLC. All rights reserved.