



How to use RefIDs and the RefID Plug In

Every document created in DBXL will have a unique DocID. Typically we refer to documents stored in DBXL by using their DocID.

However, every document also has a RefID. The RefID is unique within the document's DocType. The RefID's format can be defined from the DBXL Administration Tool (DAT). DBXL also includes a mechanism for defining a custom RefID.

In this tutorial we will discuss how to use the RefID, how to create a RefID format in DAT, and how to create a custom RefID via managed code using the RefID plugin.

PRE-REQUISITES

For this tutorial you will need to have a server with **DBXL** installed. To create a custom RefID, you will need to install **Visual Studio 2005** or a later version on your machine.

- 1) Create an XSN with **Submit** button to use the DBXL web service. This tutorial uses the below layout.

Name	
Author	
Description	
Submit	

- 2) Create a new configuration in DAT using the xsn created in the step above.



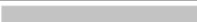
DATABASE ACCELERATOR
Administration Tool



Document Type: template  Save  Catalog

General Database Permissions SharePoint Documents Taxonomy Tree

Document Type Details

Name	Solution Group
template	 
Description	
Last Modified By	
 arunaa	

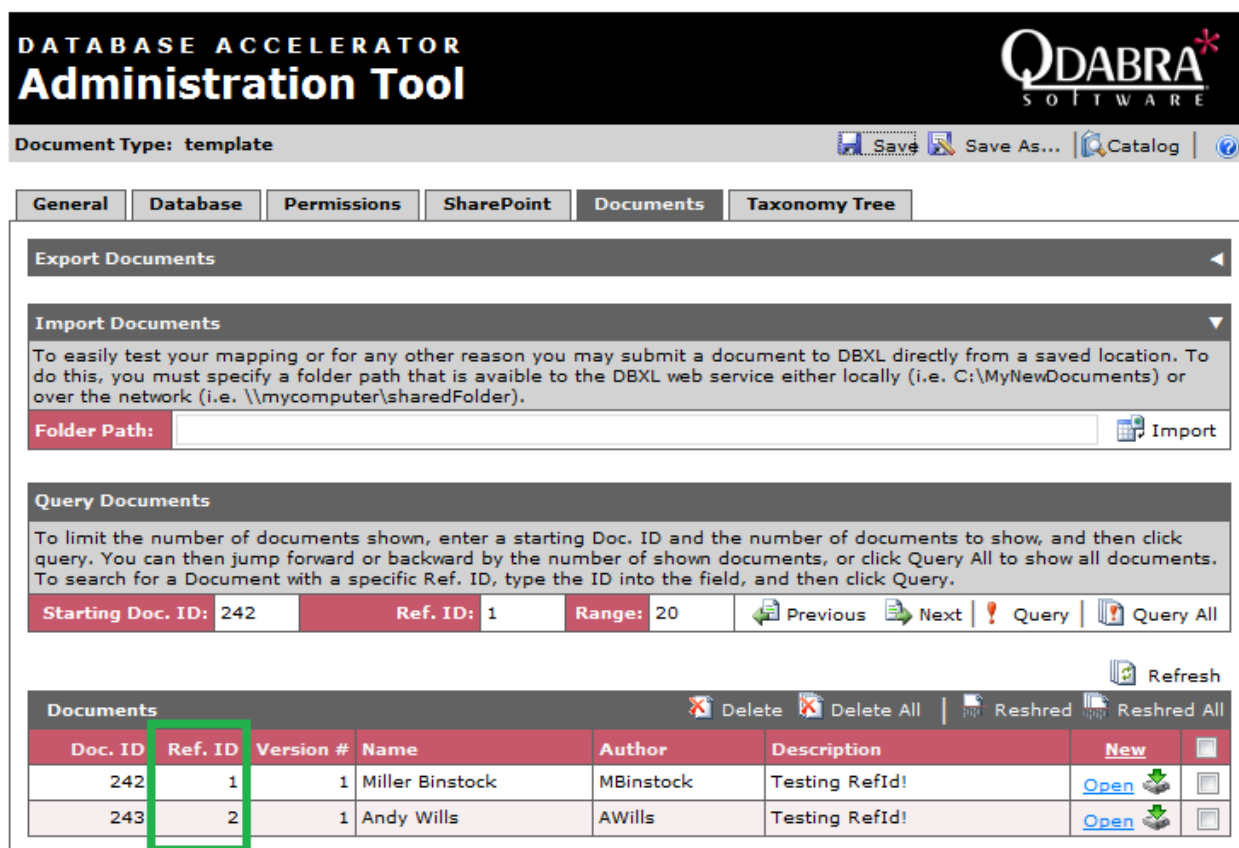
InfoPath Form Template

XSN File	Form Template Properties
 SampleForm.xsn Microsoft InfoPath Form Template 5.11 KB	Form Version: 1.0.0.4 Last Modified: 4/29/2010 11:10:39 AM ☐ Form template is signed ☐ URN-based solution



WHAT ARE REFIDS?

Every document created in DBXL has a RefID, and the RefID is unique within the document's DocType.



DATABASE ACCELERATOR Administration Tool

Document Type: template

Save Save As... Catalog

General Database Permissions SharePoint Documents Taxonomy Tree

Export Documents

Import Documents

To easily test your mapping or for any other reason you may submit a document to DBXL directly from a saved location. To do this, you must specify a folder path that is available to the DBXL web service either locally (i.e. C:\MyNewDocuments) or over the network (i.e. \\mycomputer\sharedFolder).

Folder Path: Import

Query Documents

To limit the number of documents shown, enter a starting Doc. ID and the number of documents to show, and then click query. You can then jump forward or backward by the number of shown documents, or click Query All to show all documents. To search for a Document with a specific Ref. ID, type the ID into the field, and then click Query.

Starting Doc. ID: 242 Ref. ID: 1 Range: 20 Previous Next Query Query All

Refresh

Doc. ID	Ref. ID	Version #	Name	Author	Description	New
242	1	1	Miller Binstock	MBinstock	Testing RefId!	Open
243	2	1	Andy Wills	AWills	Testing RefId!	Open

In order to use the RefID, use URLs with the following format:

<http://<servername>/QdabraWebService/Documents/<DocTypeName>/<RefID>.xml>



<http://www.qdabra.com>

Last updated on 9/12/2012 8:49 PM

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

CREATING A REFID IN DBXL DAT

1. Open DAT and click on the **Edit Configuration** () button for the document type you wish to edit.

template	1.0.0.7	0			
----------	---------	---	---	---	---

2. In the **General** tab, enter a friendly RefID format in **Format for Ref. ID** field. This tutorial uses '**RefId {0}**'.

Ref. ID		
Format for Ref. ID	Base Reference #	XPath to Inject Ref. ID Into (Leave Blank to Disable)
RefId {0}	2	

3. Click on the **Save** button to save the document type.
4. In the **General** tab, click on this  button to open the xsn.
5. Fill the form and click on the **Submit** button. Submit at least two forms so that we can see the RefId format for both the documents.
6. Switch to the **Documents** tab in DAT, you will see two documents with the RefId format as specified in the **General** tab.

Documents							 Delete	 Delete All	 Reshred	 Reshred All
Doc. ID	Ref. ID	Version #	Name	Author	Description	New				
246	RefId 1	1	Arnold	AS	Testing RefId!	Open 				
247	RefId 2	1	CliffB	CB	Testing RefId!	Open 				

7. We can use the Ref Id to open a specific document, ex: we can use the Ref Id (**RefId 1**) to open a document using the following format:

<http://<servername>/QdabraWebService/Documents/template/RefId 1.xml>



STORE THE REFID VALUE IN A FORM FIELD

1. Open the template that was created earlier in design mode to add a new field that holds the **Refid** value (for the sake of this tutorial, let's name the new field as **refid**)

Name	
Author	
Description	
Ref Id	

2. Save all the changes and re-upload to DAT.
3. In the **XPath to Inject Ref. ID Into** field, enter the XPath of the location in the XML where the reference ID is to be stored.

Ref. ID		
Format for Ref. ID	Base Reference #	XPath to Inject Ref. ID Into (Leave Blank to Disable)
<u>RefId {0}</u>	4	/my:myFields/my:refId

4. Click on the **Save** button to save the document type.
5. In the **General** tab, click on this  button to open the xsn.
6. Fill the form and click on the **Submit** button.

Sanity Check - Form 1

Name	John Smith
Author	JSmith
Description	Testing RefId!
Ref Id	

7. Switch to the **Documents** tab in DAT.

Doc. ID	Ref. ID	Version #	Name	Author	Description	New	
302	RefId 5	1	John Smith	JSmith	Testing RefId!	Open 	

8. Click on this  button to open the newly submitted document. You will see the Ref Id field is populated with the **Refid 5** value.



Sanity Check - Form 1

Name	John Smith
Author	JSmith
Description	Testing RefId!
Ref Id	RefId 5



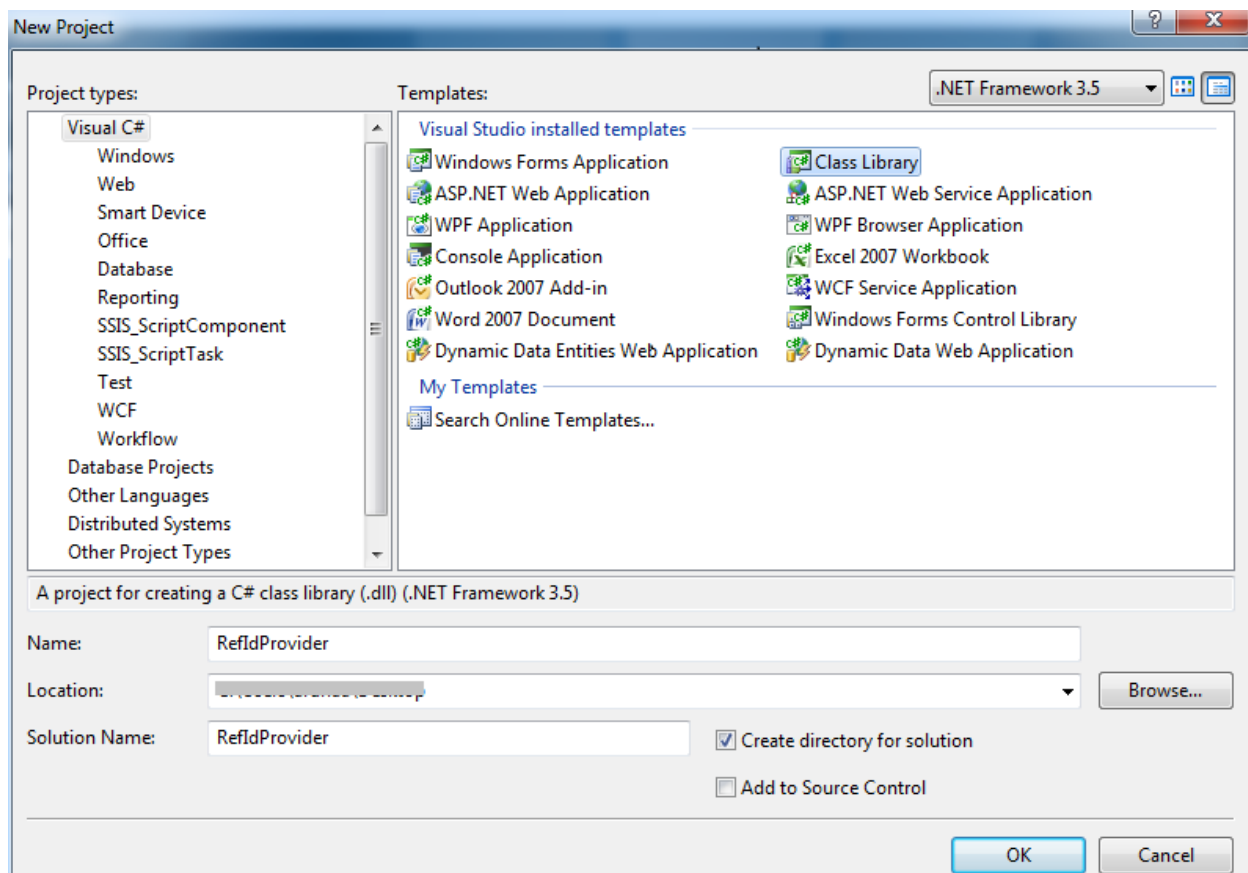
CREATING A CUSTOM REFID USING THE REF ID PLUGIN

Here is a simple walkthrough to illustrate how to create a custom RefID plugin using a simple C# class. This example uses steps and screenshots from Visual Studio 2008.

1. Open Visual Studio and create a new project by selecting **New -> Project...** from the **File** menu.

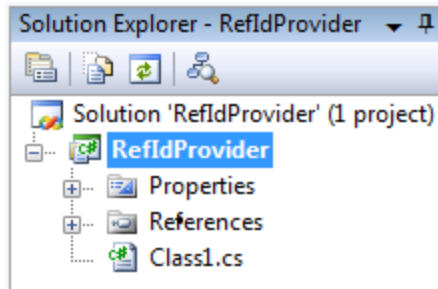


2. Select **Visual C#** under **Project types** and then select **Class Library** under **Templates**. This tutorial uses **RefIdProvider** as the project name. Click the **OK** button.

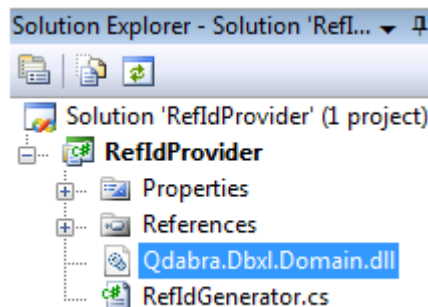


3. A new project is created with a **Class1.cs** file.





4. Give **Class.cs** a meaningful name (this tutorial uses **RefIdGenerator.cs**)
5. Add a pre-built dll:
 - a. In the Solution Explorer window of Visual Studio, right click on the project name and select **Add > Existing Item...** from the context menu.
 - b. Navigate to the **RefIdPlugin** (TODO: I may change this name, depending on what the folder name will be) folder of this tutorial and select **Executable Files** from the **Files of Type** dropdown box, and select the *Qdabra.Dbxl.Domain.dll*. Now your project should contain the *Qdabra.Dbxl.Domain.dll*.
 - c. Now right click on the **References** and select **Add Reference...** menu option.
 - d. Select the **Browse** tab, and this time browse to the *Qdabra.Dbxl.Domain.dll* file with in your code project. Click the **OK** button.



6. Open **RefIdGenerator.cs** file. After all of the using statements at the top, add a reference, **Qdabra.Dbxl.Domain.dll** and indicate that this class implements the **IDbxlRefIdProvider** interface.



```

RefIdProvider.RefIdGenerator
3  using System.Data.SqlClient;
4  using System.Linq;
5  using System.Text;
6  using System.Xml;
7  using System.Xml.XPath;
8
9  using Qdabra.Dbxl.Domain;
10
11 namespace RefIdProvider
12 {
13     public class RefIdGenerator : IDbxlRefIdProvider
14     {

```

7. Now paste the following code in between the opening and closing brackets of the class. (Note: You can use your own logic to create a unique refid string; this tutorial uses unique GUID as a refid string).

```

        private string _docTypeName = null;
        private string _connectionString = null;
        private string _initInfo = null;

        public void Init(string docTypeName, string connectionString,
string initInfo)
        {
            _docTypeName = docTypeName;
            _connectionString = connectionString;
            _initInfo = initInfo;
        }

        public string GetRefId(string docType, string refIdFormat,
XmlDocument doc)
        {
            // create a unique GUID
            string refIdString = Guid.NewGuid().ToString();
            return refIdString;
        }

```

8. Now compile this plugin project into a DLL, and copy this DLL into DBXL's bin folder.
9. Now the last step is to modify DBXL's web.config file to indicate that it should use this plugin.
- Open DBXL's web.config file and locate the **<documentTypes>** section .
 - Now add an entry to the **<documentType>** element inside the **<documentTypes>** section.
 - Now add an entry to the **<refidProvider>** element inside the **<documentType>** section.

```

<documentTypes>
  <documentType name="">
    <refidProvider type="" connectionString="" initInfo="" />
  </documentType>
</documentTypes>

```



- d. The **<documentType>** element should have one attribute, **name**, which is the name of the documentType in DBXL DAT (DBXL Database Accelerator tool). This tutorial uses **Template** as a documentType.
- e. The **<refIdProvider>** element should have three attributes, **type**, which should indicate the class and assembly name of the plugin provider. Since the assembly in this case is **RefIdProvider**, and the class is **RefIdGenerator** in **RefIdProvider** namespace, **connectionString**, the connection string, if needed (and **initInfo**, , so you should have a result like this:

```
<documentTypes>
  <documentType name="template">
    <refIdProvider type="RefIdProvider.RefIdGenerator,
RefIdProvider" connectionString="" initInfo="" />
  </documentType>
</documentTypes>
```

- f. If you create your own plugin with a different assembly name, namespace or class, a guide for the format to use in the type=attribute is

NamespaceName.ClassName,AssemblyName

10. If you carried out all of these steps successfully, your RefId plugin should be in place and ready to test. In order to test it, you will need to submit a document in DBXL DAT for a **template** document type.

- a. Open DBXL DAT and click on the **Edit Configuration** () button for a **template** document type.

template	1.0.0.7	0			
----------	---------	---	---	---	---

- b. In the **General** tab, click on this  button to submit a new document.
- c. Fill the form and click on the **Submit** button.

Name	John Smith
Author	JSmith
Description	Testing RefId Plugin!
Submit	

- d. Switch to the **Documents** tab in DAT, you will see a new document with unique GUID as a **Ref. ID**.

Documents							Refresh
							
Doc. ID	Ref. ID	Version #	Name	Author	Description	New	
225	16cf1fca-31a0-4bc5-81e3-51eee0db	1	John Smith	JSmith	Testing RefId Plugin!	Open 	

