



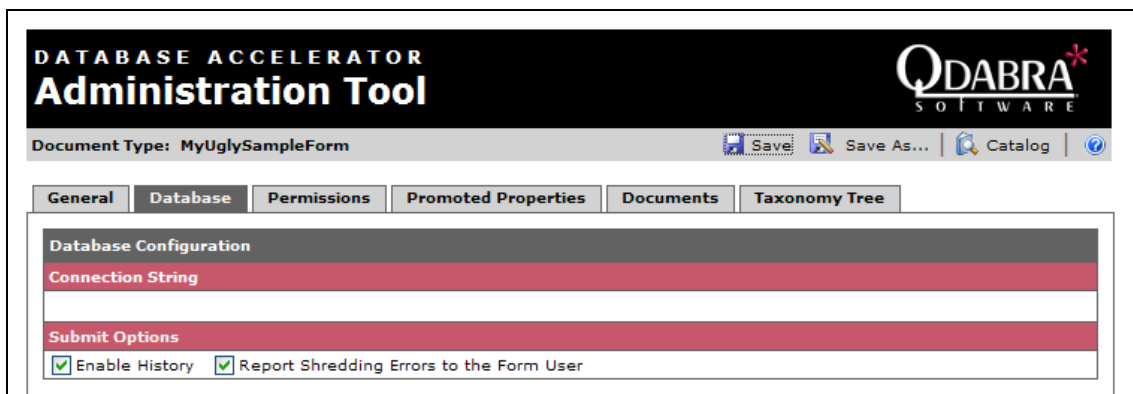
## QDABRA DATABASE ACCELERATOR

### How to link previous form versions from a workflow section

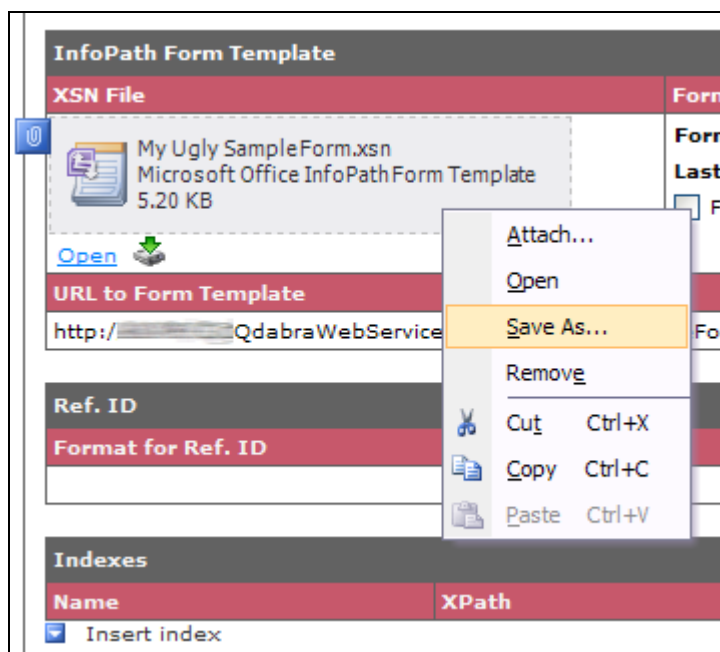
If the **Enable History** checkbox is selected for a particular Document Type, all of the previous versions for a document are saved to the database for historical review and can be retrieved. You can add workflow to your form by following the steps below.

#### DAT CONFIGURATION

1. **Enable History** by selecting the checkbox under the **Database** tab for the Document Type you would like to save versions for:



2. Save the template file from the Document Type locally:





## ADD NODES TO TEMPLATE DATA STRUCTURE

1. Open the template in design mode and add nodes to the Main Data Source for your form.
  - a. Add a group called **WorkflowHistory** under the root node:

The screenshot shows the 'Add Field or Group' dialog box. The 'Name' field contains 'WorkflowHistory'. The 'Type' dropdown is set to 'Group'. The 'Data type', 'Data namespace', and 'Default value' fields are empty. The 'Repeating' checkbox is unchecked, and the 'Cannot be blank (\*)' checkbox is also unchecked. The 'OK' and 'Cancel' buttons are at the bottom.

- b. Add a repeating group called **Workflow** under **WorkflowHistory**:

The screenshot shows the 'Add Field or Group' dialog box. The 'Name' field contains 'Workflow'. The 'Type' dropdown is set to 'Group'. The 'Data type', 'Data namespace', and 'Default value' fields are empty. The 'Repeating' checkbox is checked, and the 'Cannot be blank (\*)' checkbox is unchecked. The 'OK' and 'Cancel' buttons are at the bottom.

- c. Add a DateTime field called **EditDate** under **Workflow**:

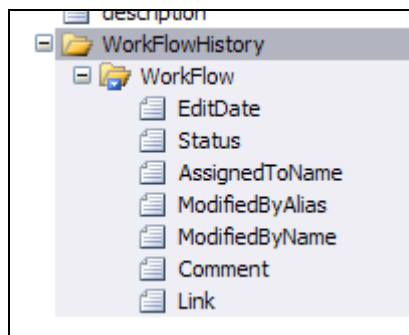


The screenshot shows a dialog box titled "Add Field or Group". It has a "Name" field containing "EditDate", a "Type" dropdown menu set to "Field (element)", and a "Data type" dropdown menu set to "Date and Time (dateTime)". There is an empty "Data namespace" field and an empty "Default value" field. Below these fields is an example of the date and time format: "Example: 12/31/2000 11:59:59 PM". At the bottom, there are two checkboxes: "Repeating" (unchecked) and "Cannot be blank (\*)" (unchecked). The "OK" and "Cancel" buttons are at the bottom right.

- d. Add 6 more String fields under **Workflow**:
- **Status**
  - **AssignedToName**
  - **ModifiedByAlias**
  - **ModifiedByName**
  - **Comment**
  - **Link**

The screenshot shows a dialog box titled "Add Field or Group". It has a "Name" field containing "Status", a "Type" dropdown menu set to "Field (element)", and a "Data type" dropdown menu set to "Text (string)". There is an empty "Data namespace" field and an empty "Default value" field. Below these fields is an example of the text format: "Example: Sample Text". At the bottom, there are two checkboxes: "Repeating" (unchecked) and "Cannot be blank (\*)" (unchecked). The "OK" and "Cancel" buttons are at the bottom right.

The completed schema should look like this in the Data Source Task Pane:



2. The finished xml should look like this:

```
<my:WorkFlowHistory>
  <my:WorkFlow>
    <my:EditDate xsi:nil="true"></my:EditDate>
    <my:Status></my:Status>
    <my:AssignedToName></my:AssignedToName>
    <my:ModifiedByAlias></my:ModifiedByAlias>
    <my:ModifiedByName></my:ModifiedByName>
    <my:Comment></my:Comment>
    <my:Link></my:Link>
  </my:WorkFlow>
</my:WorkFlowHistory>
```

**Note:** This is a suggested structure. You may add whatever fields you like to the repeating group if you wish to track other version metadata.

### ADD NEW FIELDS TO FORM

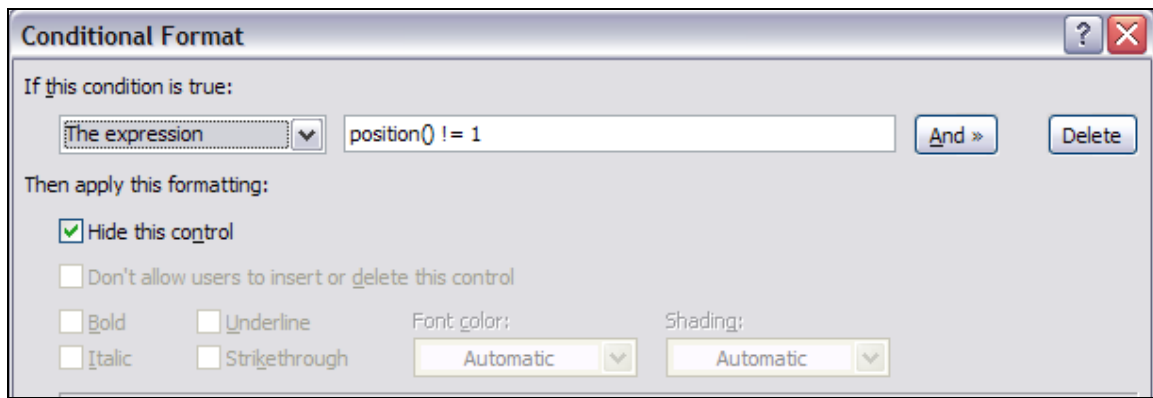
1. Add a repeating section on your form bound to the **WorkFlow** group that contains controls bound to the fields you'd like the user to be able to edit:

A drop down box is used for Status in this sample form, and a text box for Assigned To and Comment.

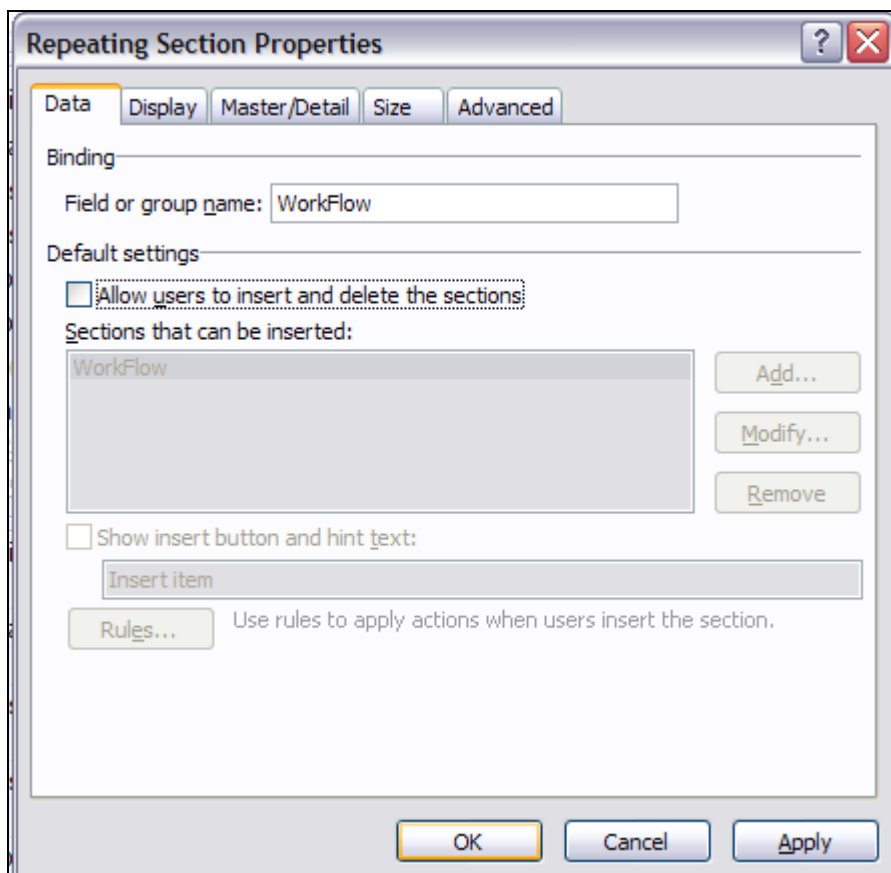
2. For this sample, users will select status and fill out assigned to and add a comment on each version when they modify the form.
3. Right click the repeating section and select Conditional Formatting from the context menu, and then click Add.



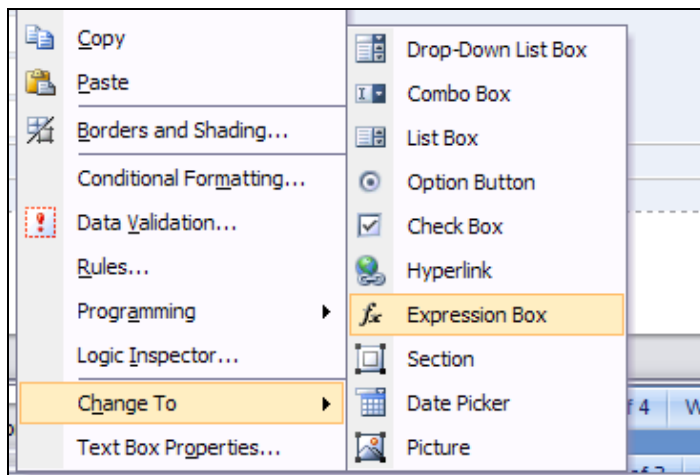
4. Select “The expression” from the first drop down box, and enter **position() != 1** in the text area.
5. Select the **Hide this control** checkbox.
6. The completed conditional formatting should look like this:



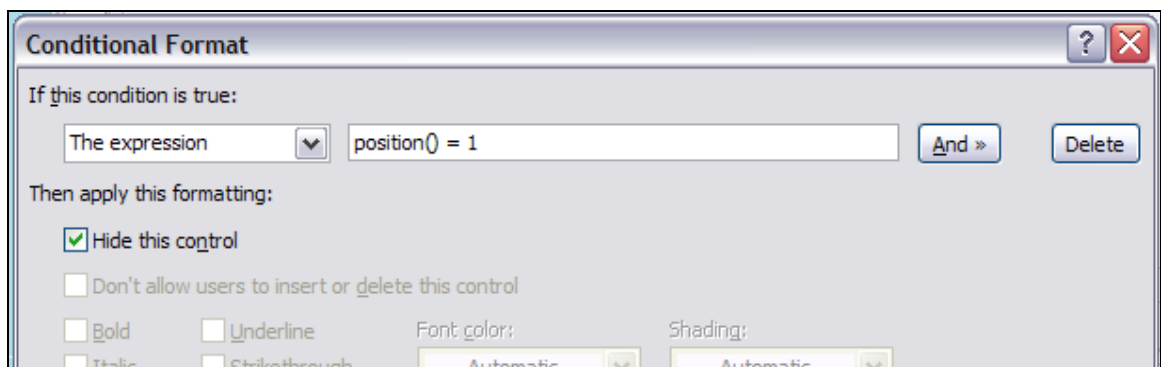
7. In the repeating section properties, deselect **Allow users to insert and delete the sections**:



8. Add another repeating section bound to the **WorkFlow** group that contains the fields you’d like to have displayed as a history. Changing the default text boxes to expression boxes makes this section read only:



9. Set the **Link** field to use a Hyperlink control.
10. Right click the repeating section and select Conditional Formatting from the context menu, and then click Add.
11. Select “The expression” from the first drop down box, and enter **position() = 1** in the text area.
12. Select the **Hide this control** checkbox.
13. The completed conditional formatting should look like this:



14. In the repeating section properties, deselect **Allow users to insert and delete the sections**.

## ADD SECONDARY DATA SOURCE

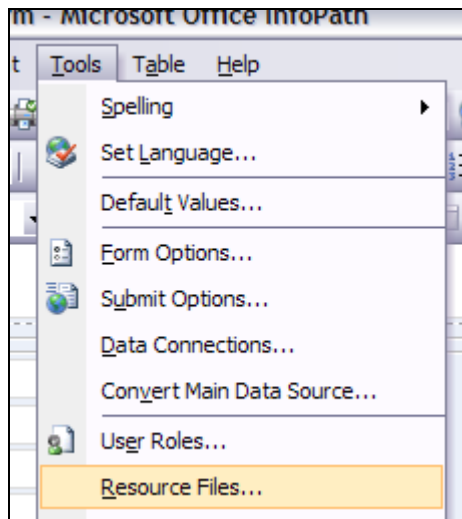
We will use a rule to store the base URL for our forms in a node in a secondary data source. This value will be used as part of the default value for the **Link** node that will give us access to previous versions of a form.

1. Create a new xml file with the following content:

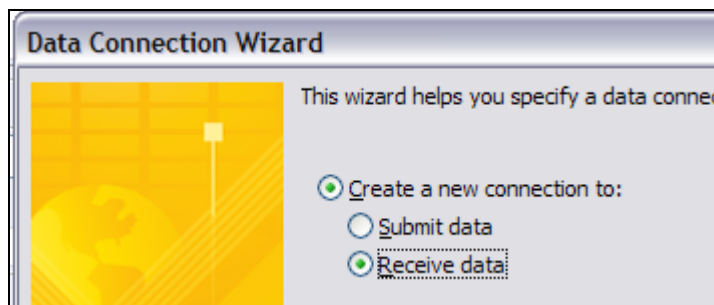
```
<?xml version="1.0" encoding="utf-8" ?>
<ViewOptions>
    <DownLevelLinkUrl></DownLevelLinkUrl>
</ViewOptions>
```



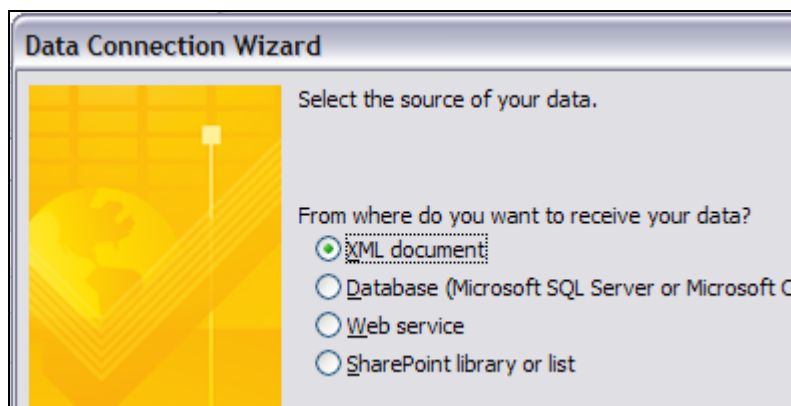
2. Save the file with the name **ViewOptions.xml**
3. Add the new xml file to your form as a resource:
  - a. Go to **Tools / Resource Files**



- b. Select **Add**.
  - c. Navigate to the **ViewOptions.xml** file you created in steps 1 and 2 and select **OK**.
4. Create a new data connection to receive data from the **ViewOptions.xml** resource file:
  - a. Add a new **Receive** data connection:

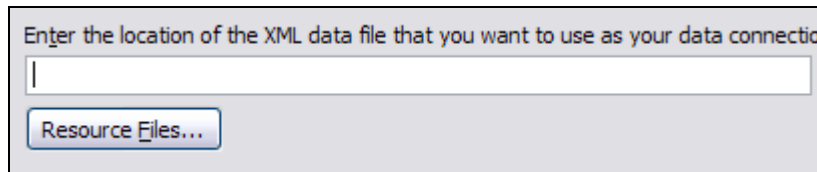


- b. Use an XML document for the source:

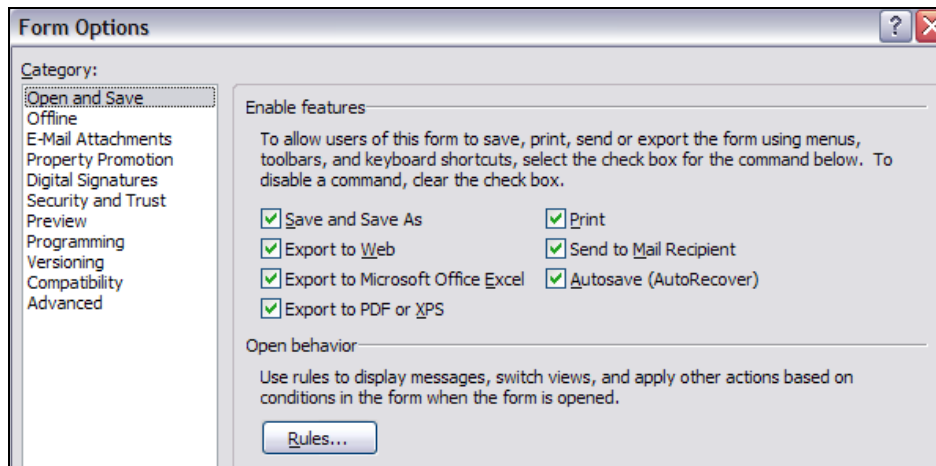




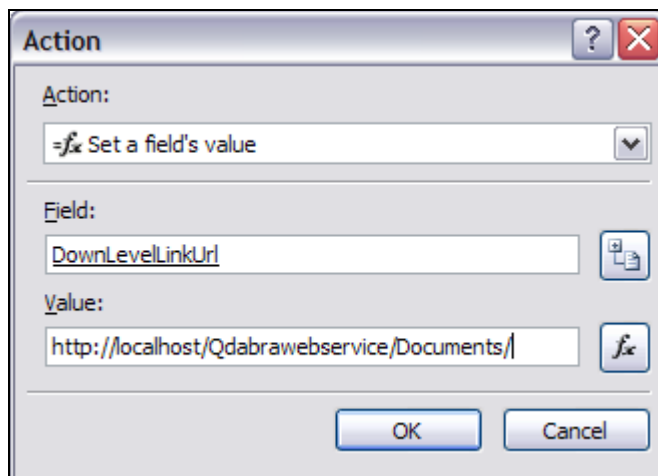
- c. Select the **Resource Files** button, and select the **ViewOptions.xml** resource file:



- 5. Set a form open rule to fill the **DownLevelLinkUrl** field in the new ViewOptions data connection:
  - a. Select **Tools/Form Options**
  - b. Under **Open and Save**, select the **Rules** button:



- c. Create a new rule that sets the value of the **DownLevelLinkUrl** field in the new ViewOptions data connection to <http://servername:port/QdabraWebService/Documents/> where servername:port is the <machine name>:<port> where DBXL is installed and QdabraWebService is the virtual directory name that you used while installing DBXL.



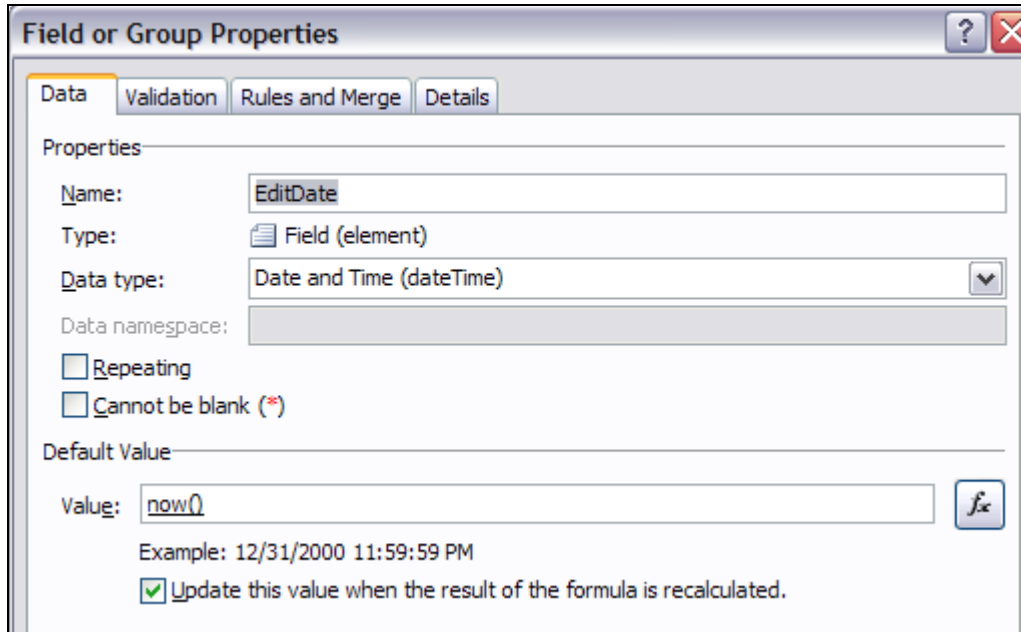
## SET DEFAULT VALUES

- 1. Double click the **EditDate** field in the Data Source Task Pane.





2. Select the *fx* button to open the formula editor.
3. Set the default value to **now()**:



4. Double click the **Link** field in the Data Source Task Pane.
5. Select the *fx* button to open the formula editor.
6. Paste the following formula into the formula editor to create a link to previous versions:

```
concat(xdXDocument:GetXDOM("ViewOptions")/ViewOptions/DownLevelLinkUrl,
substring-before(substring-after(/processing-instruction() [local-name(.)
="QdabraDBXL"], 'docid="'), ''), "v-", (count(..preceding-
sibling::my:Workflow) - 1), "/", substring-before(substring-
after(/processing-instruction() [local-name(.) = "QdabraDBXL"],
'docid="'), ''), "v-", (count(..preceding-sibling::my:Workflow) - 1),
".xml")
```

**Note:** This formula presumes your data connection to the **ViewOptions.xml** file is called **ViewOptions** and that you maintained the suggested structure for the **WorkflowHistory** group.

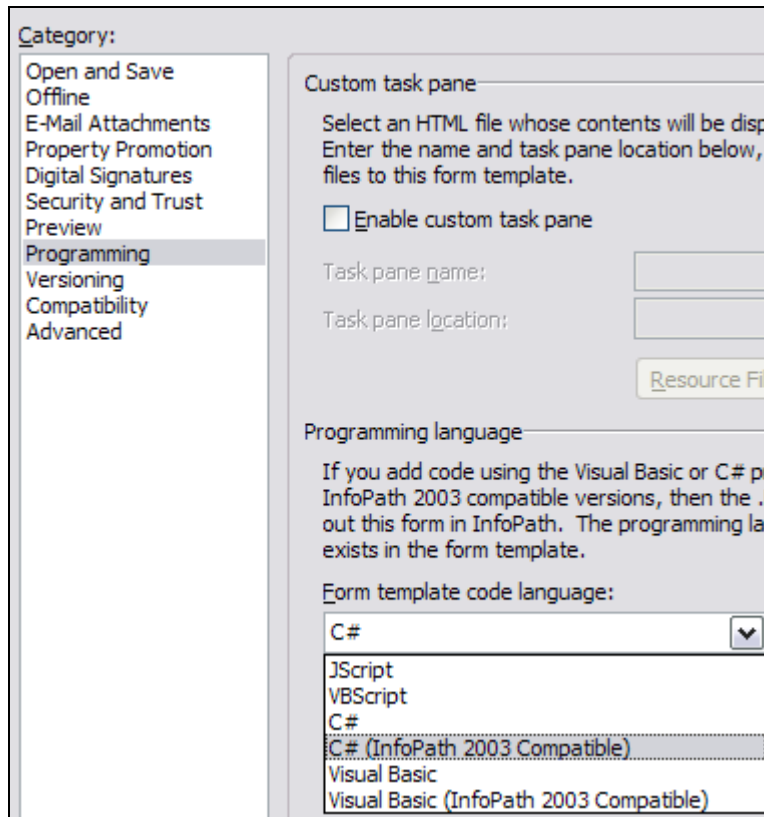
## ADD CODE TO THE FORM

We will add code that will add a new **Workflow** node each time a filled out form is opened. Please note that this is sample code, without error handling. Also, this code does not deal with the *xsi:nil* attribute. If you do not have a default value set for a field with a data type of Date, Decimal, or other types that InfoPath adds the *xsi:nil* attribute to, you will need to add the attribute yourself when you clone the node from the sample data.

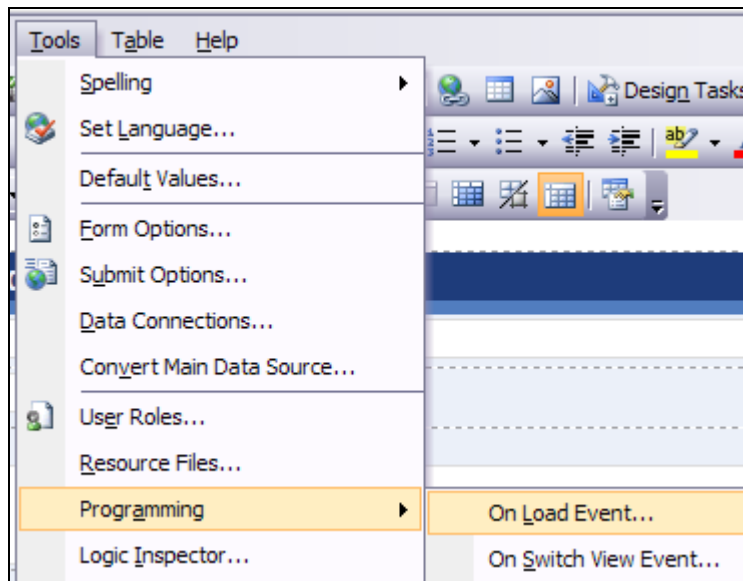
1. Set your form to use C# (InfoPath 2003 Compatible) for its programming language:
  - a. Go to **Tools/Form Options**.
  - b. Select **Programming** in the left hand Category pane.



- c. Select C# (InfoPath 2003 Compatible)



- 2. Add an On Load event to your form:
  - a. Go to **Tools/Programming/On Load Event**:



VSTA will open. Add the following code in the On Load event:



```
string formUri = thisXDocument.URI.ToString();

string version = formUri.Substring(formUri.Length - 7, 3);

if (!thisXDocument.IsNew && !thisXDocument.IsReadOnly &&
!version.Equals("v-0"))

    AddWorkFlowNode();
```

We don't want to add a new work flow node if the form is new or when we open a previous version, so the if condition checks to see if the form is new, or read only – with the exception of the newest version, versions open as read only from DAT. We also check the form URI, to see if it is the newest version, and we don't add a work flow node if that is the case.

3. Add another variable declaration at the top of the class, beneath the line `private Application thisApplication;`

```
private IXMLDOMDocument2 sampleDataDom;
```

4. Add the following functions to your code:

```
private void AddWorkFlowNode()
{
    IXMLDOMNode previousWorkFlowNode =
thisXDocument.DOM.selectSingleNode("/my:myFields/my:WorkFlowHistory/my:Wo
rkFlow[1]");
    IXMLDOMNode workFlowNode =
GetNodeFromSampleData("/my:myFields/my:WorkFlowHistory/my:WorkFlow",
"xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"
xmlns:xhtml=\"http://www.w3.org/1999/xhtml\"
xmlns:dfs=\"http://schemas.microsoft.com/office/infopath/2003/dataFormSol
ution\" xmlns:tns=\"http://qdabra.com/webservices\"
xmlns:s1=\"http://schemas.microsoft.com/sharepoint/soap\"
xmlns:s2=\"urn:schemas-microsoft-com:rowset\" xmlns:s3=\"#RowsetSchema\"
xmlns:my=\"http://schemas.microsoft.com/office/infopath/2003/myXSD/2008-
09-06T21:16:50\"
xmlns:xd=\"http://schemas.microsoft.com/office/infopath/2003\"");
    if (previousWorkFlowNode != null && workFlowNode != null)
    {

previousWorkFlowNode.parentNode.insertBefore(workFlowNode,
previousWorkFlowNode);
    }
}

public IXMLDOMNode GetNodeFromSampleData(string XPath, string
formNamespaces)
{
    IXMLDOMNode nodeFromSampleData =
GetNode(GetSampleDataDom(formNamespaces), XPath);
    if (nodeFromSampleData == null)
    {
        return null;
    }
}
```



```
    }
    return nodeFromSampleData.cloneNode(true);
}
public IXMLDOMDocument2 GetSampleDataDom(string namespaces)
{
    if (sampleDataDom == null)
    {
        sampleDataDom =
(IXMLDOMDocument2)thisXDocument.CreateDOM();
        sampleDataDom.async = false;
        sampleDataDom.validateOnParse = false;
        sampleDataDom.setProperty("SelectionNamespaces",
namespaces);
        sampleDataDom.load("sampledata.xml");
    }
    return sampleDataDom;
}
public IXMLDOMNode GetNode(IXMLDOMDocument2 xml, string xpath)
{
    IXMLDOMNode node = xml.selectSingleNode(xpath);

    return node;
}
```

6. You must modify the following functions to match your form:

a. AddWorkflowNode()

i. Change the highlighted xpath in the first line from

(`"/my:myFields/my:WorkflowHistory/my:Workflow[1]"`) to the correct xpath to the first repeating **Workflow** group in your form.

ii. Change the highlighted namespaces in the second line to match the namespaces used in your form. You can save the form as source files and copy the namespaces from **myschema.xsd** to get the correct ones. You can escape the quote characters with a \. Find and replace is helpful for that – paste the namespace into a text editor and search for double quotes. Replace with \" to escape the double quote characters properly.

#### ATTACH TEMPLATE AND TEST

1. Attach the new template to the Document Type in DAT.
2. Create a new document from the Document Type and submit it:



### My Ugly Sample Form

**Issue:** Test Issue

**Date:** 10/22/2008

**Description:**  
I think this form is ugly

**Additional Information:**

Status: New

Assigned To: Hilary Stoupa

Comment: Assigning to myself, as it is my fault form is ugly.

**Historical Information**

3. Open the same document created in step 2, add or change some information, and submit it again:

### My Ugly Sample Form

**Issue:** Test Issue

**Date:** 10/22/2008

**Description:**  
I think this form is ugly. This is the second version.

**Additional Information:**

Status: On Hold

Assigned To: Hilary Stoupa

Comment: Still Ugly

**Historical Information**

Edit Date:  
10/22/2008 3:28:56 PM

Status:  
New

Assigned To Name:  
Hilary Stoupa

Modified By Alias:  
HilaryS

Modified By Name:  
Hilary Stoupa

Comment:  
Assigning to myself, as it is my fault form is ugly.

Link: <http://pledge/Qdabrawebservice/Documents/275v-0/275v-0.xml>

Note that previous version data shows in the Historical Information section in the sample form.



4. Open the same document again, and select the link to the first version of the form (bottommost link) to open the original form. Note that the version numbering may seem counter-intuitive, as the lowest number is the most recent version and the highest number is the oldest version.