



Lab 2 – Make Data Connections Dynamic

Overview

The goal of this Lab activity is to orient you with the DynamicDCs XTP's useful features for migrating your forms to other site locations. With the aid of qRules, the painful process of having to go through and manually update Submit and Query data connections to point to your new SharePoint site can now be eliminated.

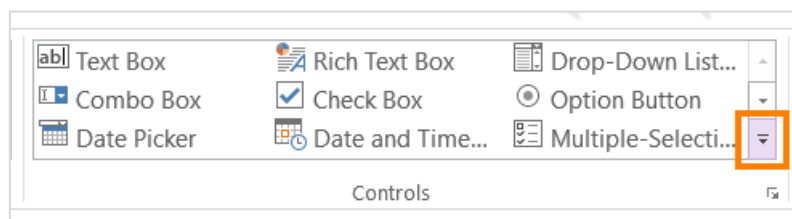
Supported Data Connections

- Submit to Library
- SharePoint List mappings via qRules' SubmitToSharePointList command
- Library query
- List query
- GetUserProfileByName

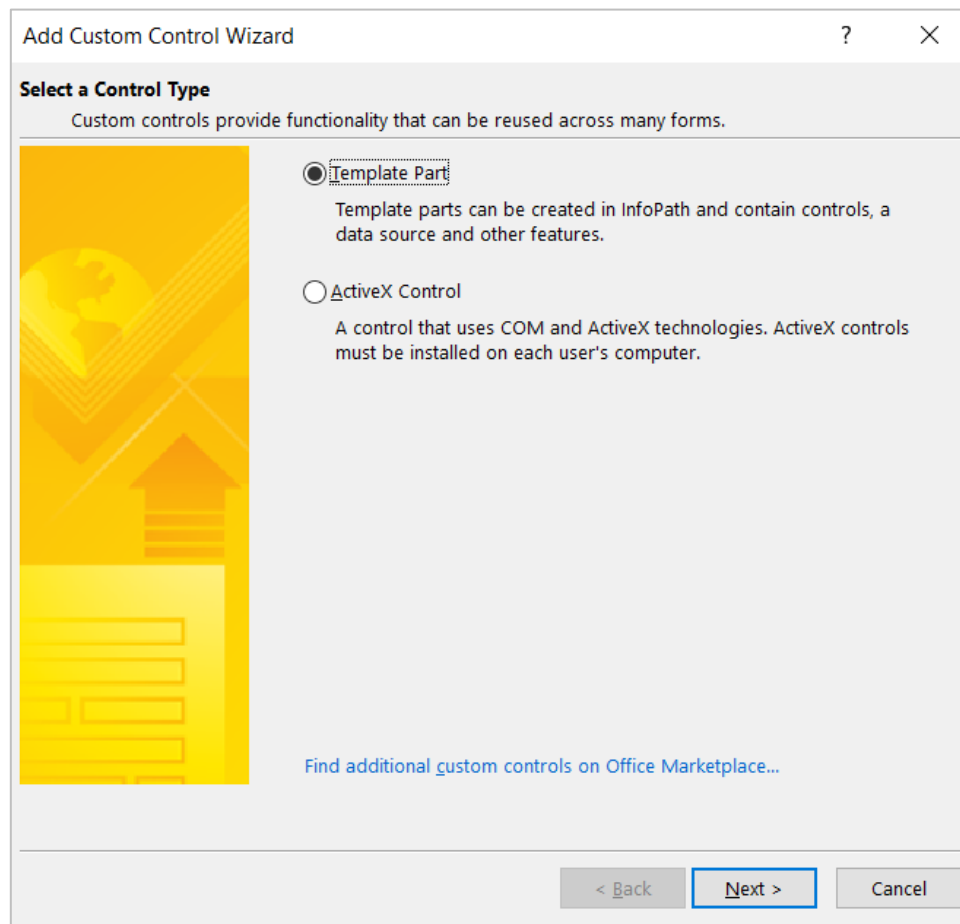
Forms may have one or more of these data type connections mentioned above, except for the GetUserProfileByName, which is often used only once.

Add the Dynamic DCs XTP

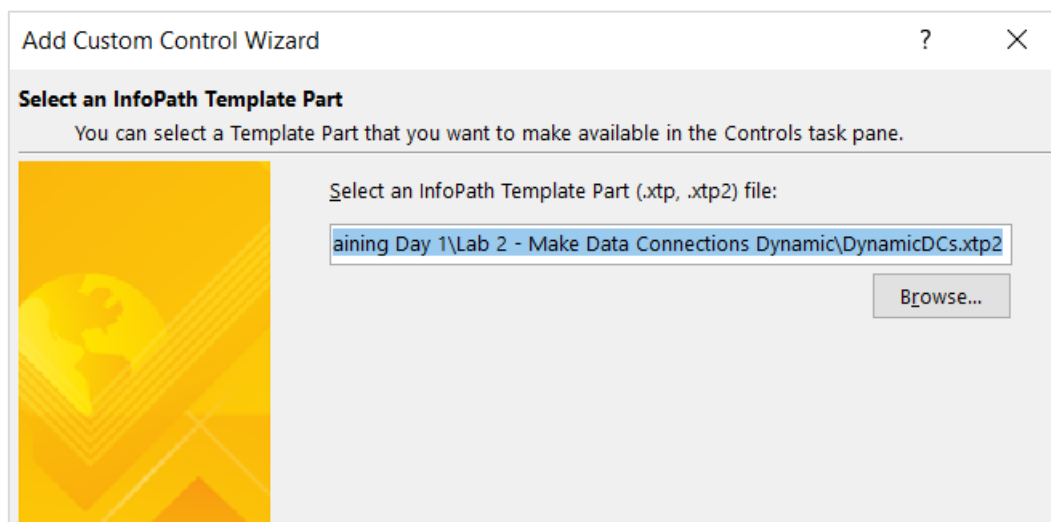
1. Open *Lab 2 - Dynamic DCs Start.xsn* in Design mode.
2. Under the Home Tab, expand the Controls section



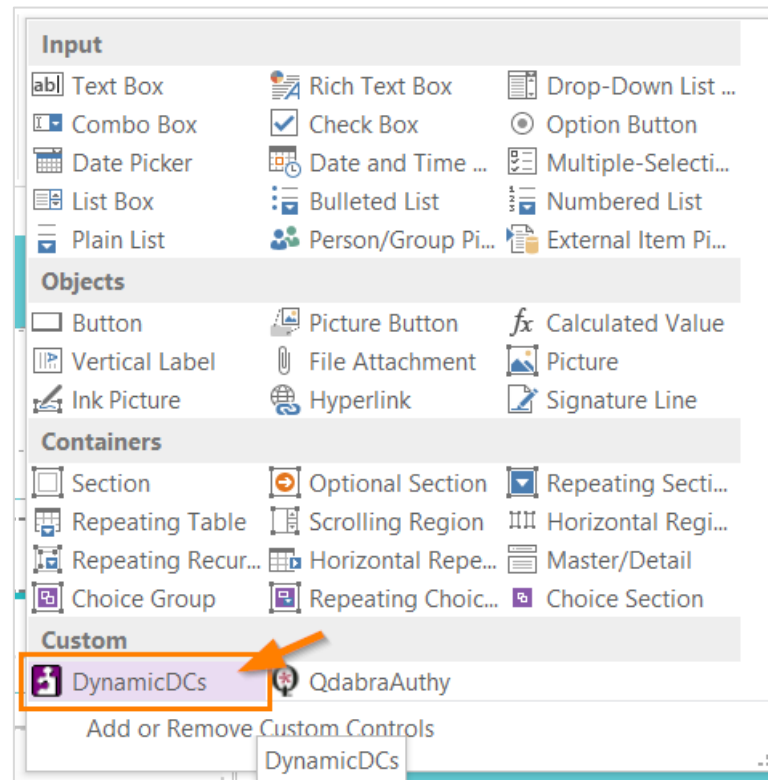
3. Click **Add or Remove Custom Controls**.
4. Click **Add**.
5. Select **Template Part**.



6. Click **Next**.
7. At the next page, click Browse and locate the DynamicDCs.xtp2 file



8. Click Finish and close the Custom Control Wizard. The DynamicDCs XTP should now be available at the Custom Controls section:



9. Point your mouse cursor at the bottom of the form canvas and select the DynamicDCs from the Custom Controls section. This should add the XTP section to your form. All the instructions on how to update the XTP rules are included within the XTP section.

The screenshot shows the Qdabra software interface for designing a form. The top ribbon includes tabs for FILE, HOME, INSERT, PAGE DESIGN, DATA, DEVELOPER, CONTROL TOOLS, TABLE TOOLS, PROPERTIES, and LAYOUT. The main workspace displays a form titled 'Form1 - InfoPath'. The form contains a table with the following structure:

EXPENSE ITEMS					FROM:	TO:
CATEGORY *	DATE *	DESCRIPTION	MERCHANT	RECEIPT	AMOUNT *	
Select...				Attach file		
SUBTOTAL						
LESS CASH ADVANCE					\$0.00	
TOTAL						

Below the table is a 'Submit' button. At the bottom of the form, there is a section titled 'Dynamic DCs XTP' with the following text:

The **Dynamic DCs XTP** is intended for modifying/updating your form's data connections, and supports the following data connection types:

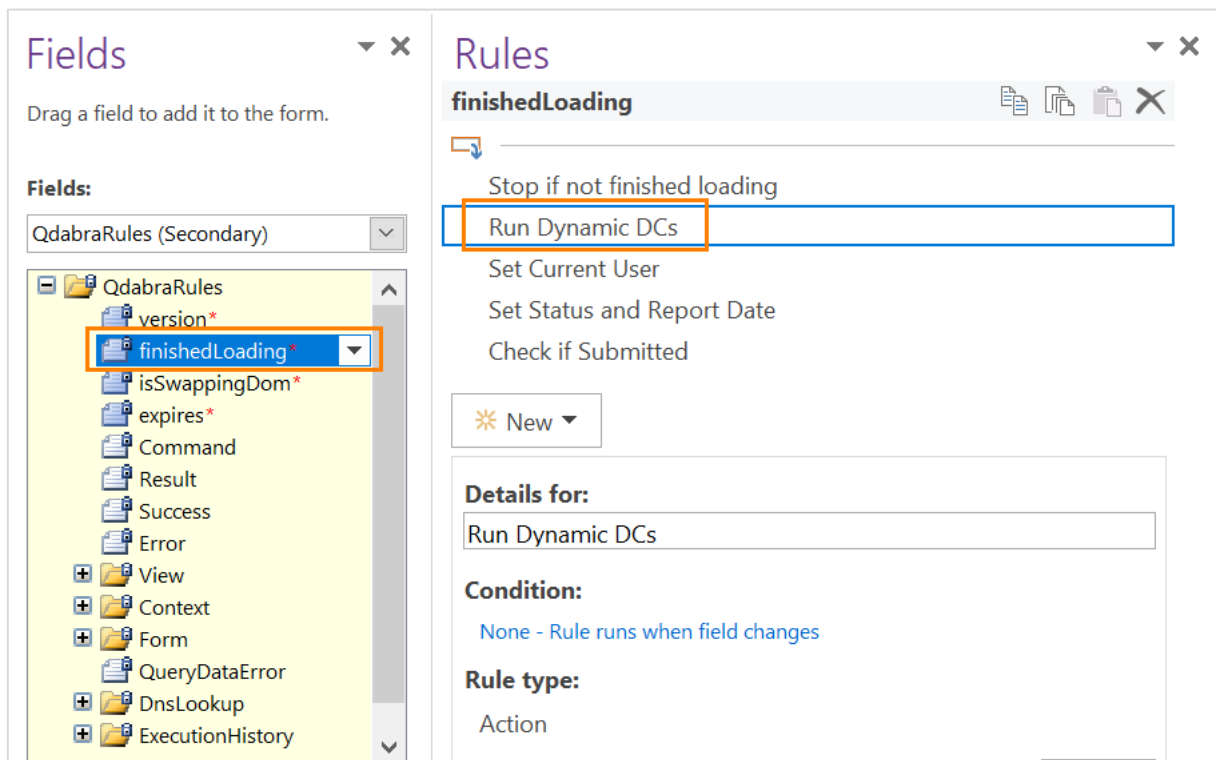
- Submit to Library

On the right side, the 'Fields' pane is open, showing a tree view of the form's structure. The 'ExpenseReport' folder is expanded, showing sub-items: Form, Resource, ExpenseDetails, ReportTracking, and DynamicDCs. The 'DynamicDCs' item is selected.

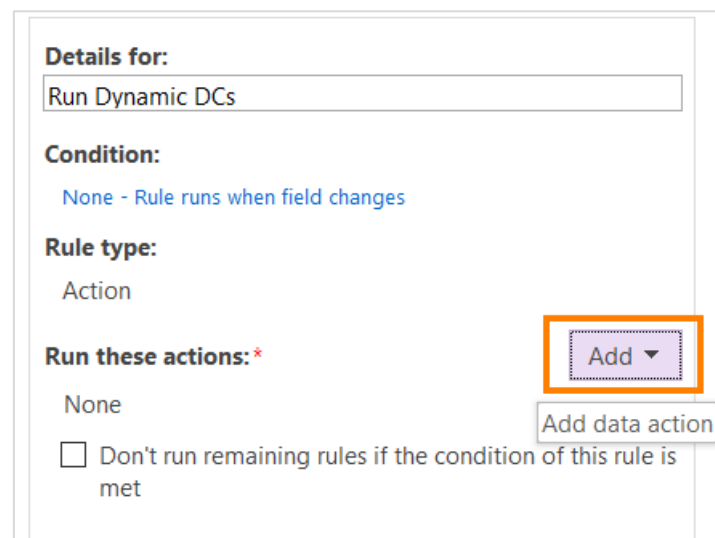
Add qRules Logic

This section assumes that your form already has qRules injected. Inject qRules if you haven't done so; otherwise, proceed with the action steps below.

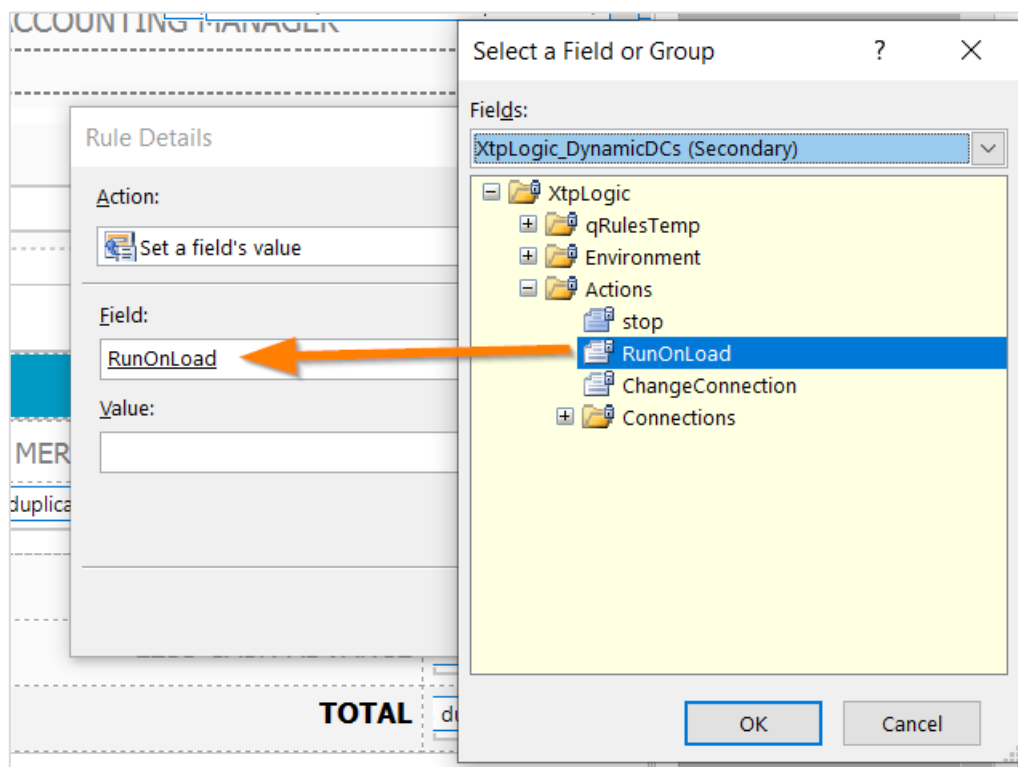
1. At the QdabraRules' finishedLoading node, add a rule and give it an appropriate label such as Run Dynamic DCs (be sure that this rule runs second from the top).



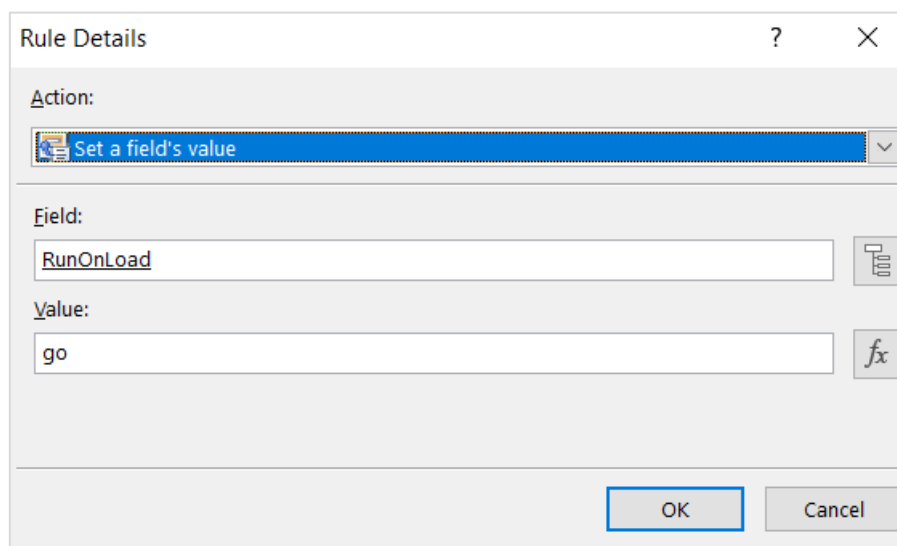
2. Click **Add** to add an action.



3. For the Target field, select the RunOnLoad node located under the Actions group of the XtpLogic_DynamicsDCs data source.



4. Set the field's value to the value **go**.



5. Expand the **qRulesTemp** section and click the rule named '**Set qRules [edit]**' located in the **XtpLogic_DynamicDCs** data source **/XtpLogic/qRulesTemp/tempCommand** node.

Fields

Drag a field to add it to the form.

Fields:

XtpLogic_DynamicDCs (Secondary)

- XtpLogic
 - qRulesTemp
 - tempCommand
 - tempResult
 - tempSuccess
 - tempError
 - Environment
 - Actions

Rules

tempCommand

Set qRules [edit]

Reset self

New

Details for:

Set qRules [edit]

Condition:

tempCommand is not blank

Rule type:

Action

Run these actions: *

Add

Set a field's value: @stop = .

Set a field's value: tempResult = "[Your QdabraRules Result node]"

Set a field's value: tempSuccess = "[Your QdabraRules Success node]"

Set a field's value: tempError = "[Your QdabraRules Success node]"

☐ Don't run remaining rules if the condition of this rule is met

6. Select the first action and point this to the *QdabraRules'* **Command** node.

Fields

Drag a field to add it to the form.

Fields:

XtpLogic_DynamicDCs (Secondary)

- XtpLogic
 - qRulesTemp
 - tempCommand
 - tempResult

Rules

tempCommand

Set qRules [edit]

Reset self

New

Details for:

Set qRules [edit]

Condition:

tempCommand is not blank

Rule type:

Action

Run these actions: *

Add

Set a field's value: @stop = .

Set a field's value: tempResult = "[Your QdabraRules Result node]"

Set a field's value: tempSuccess = "[Your QdabraRules Success node]"

Set a field's value: tempError = "[Your QdabraRules Error node]"

☐ Don't run remaining rules if the condition of this rule is met

Rule Details

Action:

Set a field's value

Field:

Command

Value:

.

OK Cancel

Submit

- Set the next action's value to the *QdabraRules'* **Result** node. And repeat the same for the rest of the actions – setting the tempSuccess field to the QdabraRules Success node and the tempError field to the QdabraRules Error node.

Details for:

Condition:
tempCommand is not blank

Rule type:
Action

Run these actions: * Add ▼

- Set a field's value: Command = .
- Set a field's value: tempResult = "[Your QdabraRules Result node]"
- Set a field's value: tempSuccess = "[Your QdabraRules Success node]"
- Set a field's value: tempError = "[Your QdabraRules Error node]"

☐ Don't run remaining rules if the condition of this rule is met

Here's what it should now look like:

Details for:

Condition:
tempCommand is not blank

Rule type:
Action

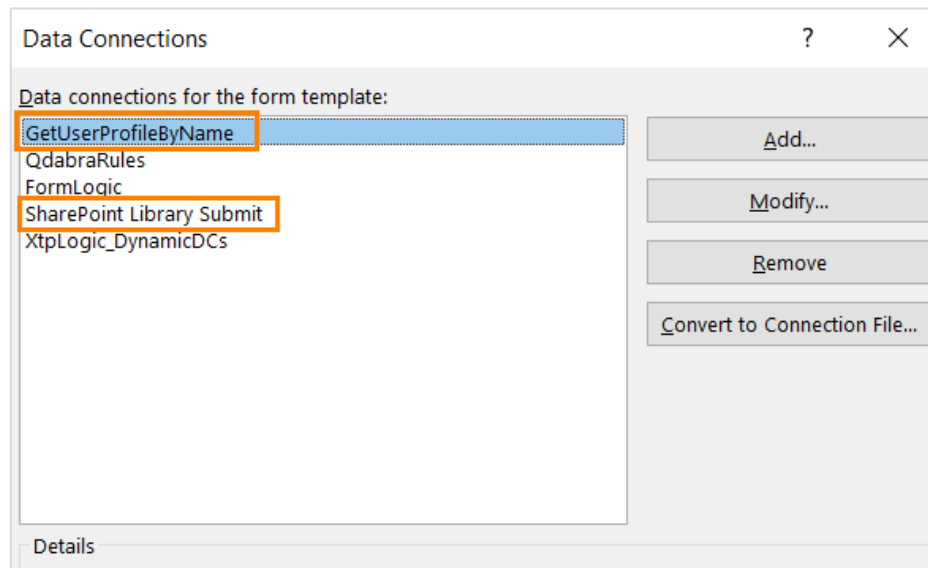
Run these actions: * Add ▼

- Set a field's value: Command = .
- Set a field's value: tempResult = Result
- Set a field's value: tempSuccess = Success
- Set a field's value: tempError = Error

☐ Don't run remaining rules if the condition of this rule is met

Enable the Dynamic Data Connections Logic

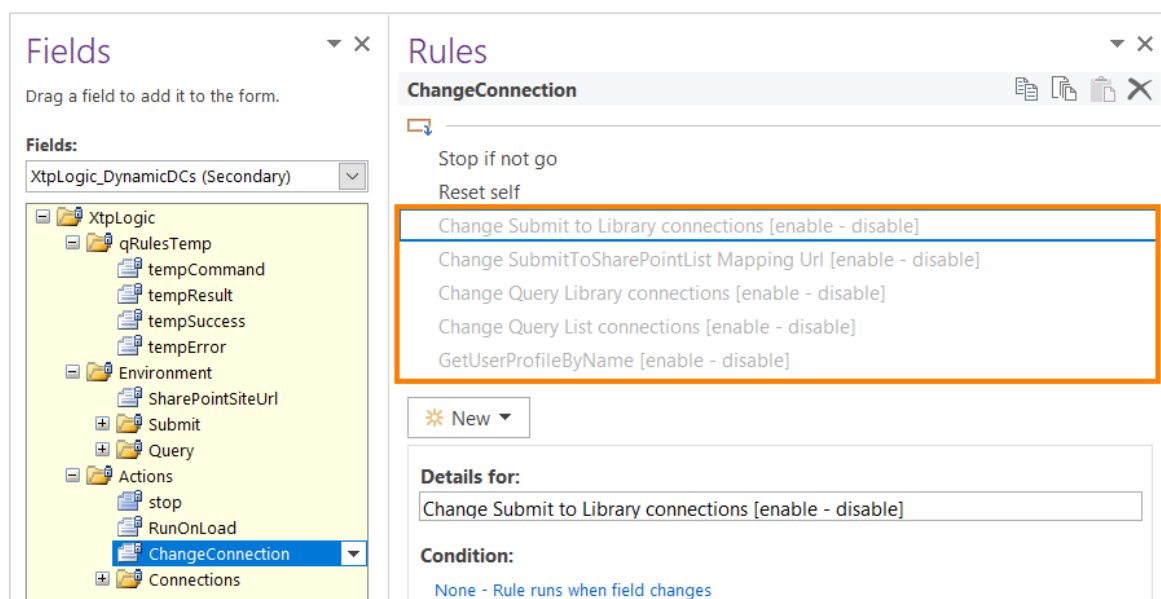
The Expense Report sample has two data connections that we can convert to dynamic:



These data connections falls into supported types below:

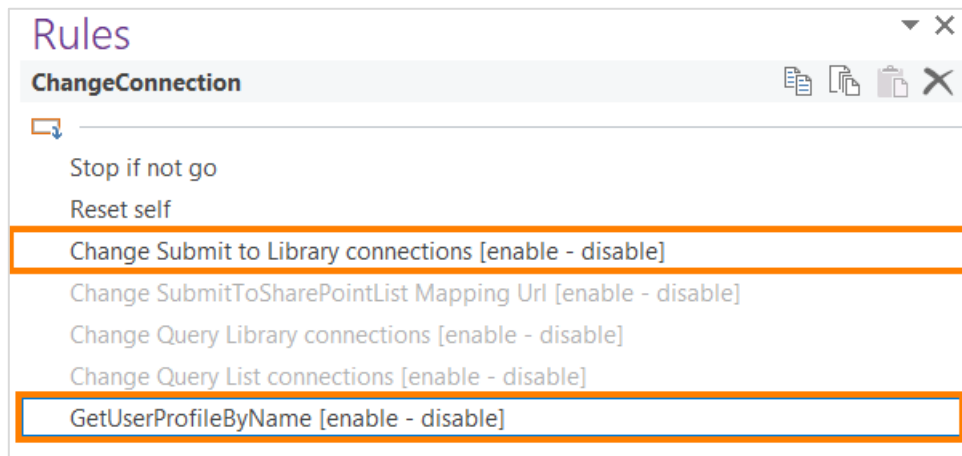
- Submit to Library
- GetUserProfileByName

Under the **XtpLogic_DynamicDCs** data source, proceed to *XtpLogic* → *Actions* → **ChangeConnection** node, notice the last five rules labeled with '[enable - disable]' – which are disabled by default:



These rules trigger the logic for updating the target data connections to point to the new site where FormsViewer is installed.

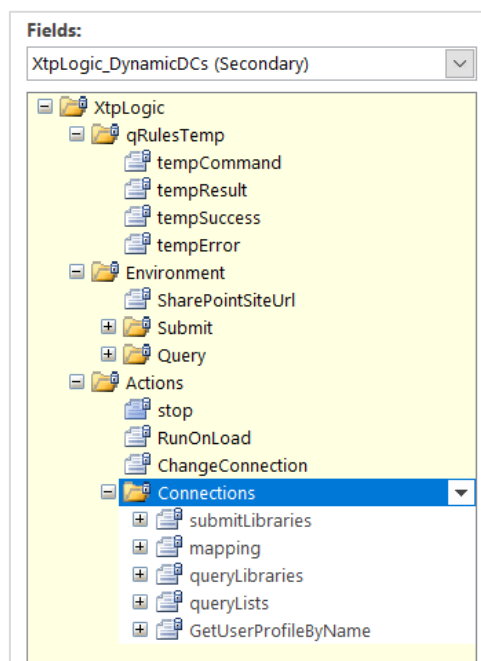
To enable the data connection update logic needed by your form, simply right click on the Action rule and click Enable. In the Expense Report sample, we will enable Submit to Library and GetUserProfileByName:



Modify the Data Connections and point them to the library/list

In this section, we will specify the data connection names and the target library.

1. Proceed to XtpLogic → Actions → Connections group of the of the XtpLogic_DynamicDCs data source.



2. Click on the **submitLibraries** node.
3. Select the first 'SharePoint Library Submit [edit]' rule.

Fields

Drag a field to add it to the form.

Fields:

XtpLogic_DynamicDCs (Secondary)

- XtpLogic
 - qRulesTemp
 - tempCommand
 - tempResult
 - tempSuccess
 - tempError
 - Environment
 - SharePointSiteUrl
 - Submit
 - Query
 - Actions
 - stop
 - RunOnLoad
 - ChangeConnection
 - Connections
 - submitLibraries**
 - mapping
 - queryLibraries
 - queryLists
 - GetUserProfileByName

Rules

submitLibraries

- Stop if not go
- Reset self
- SharePoint Library Submit [edit]**
- SharePoint Library Submit 2 [edit]
- SharePoint Library Submit 3 [edit]

Details for:

SharePoint Library Submit [edit]

Condition:

None - Rule runs when field changes

Rule type:

Action

Run these actions: *

Set a field's value: @dataConnectionName = "[Enter Submit to Library Data Connection Name]"

Set a field's value: @libraryName = "[Enter Library Name]"

☐ Don't run remaining rules if the condition of this rule is met

4. For the first action, modify '[Enter Submit Data Connection Name]' with the name of the Submit to Library data connection in the Expense Report form.

Rule Details

Action:

Set a field's value

Field:

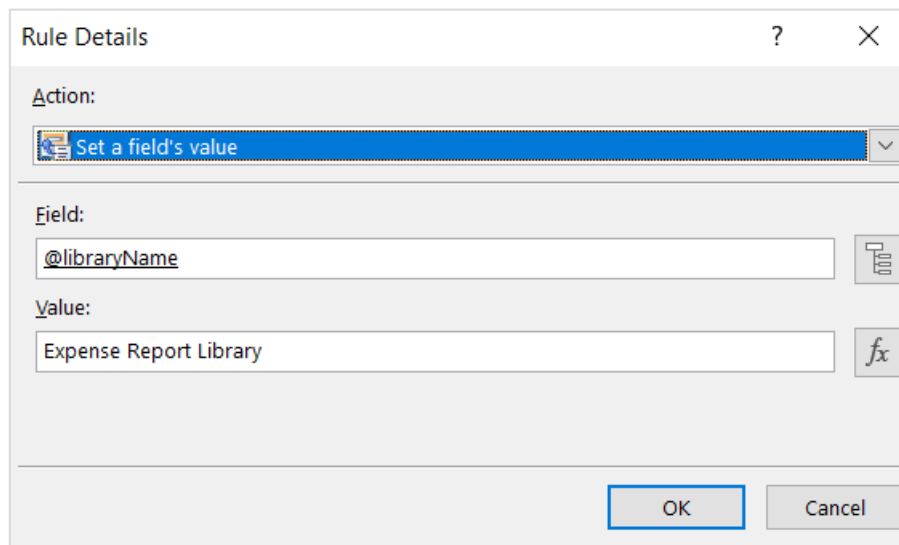
@dataConnectionName

Value:

SharePoint Library Submit

OK Cancel

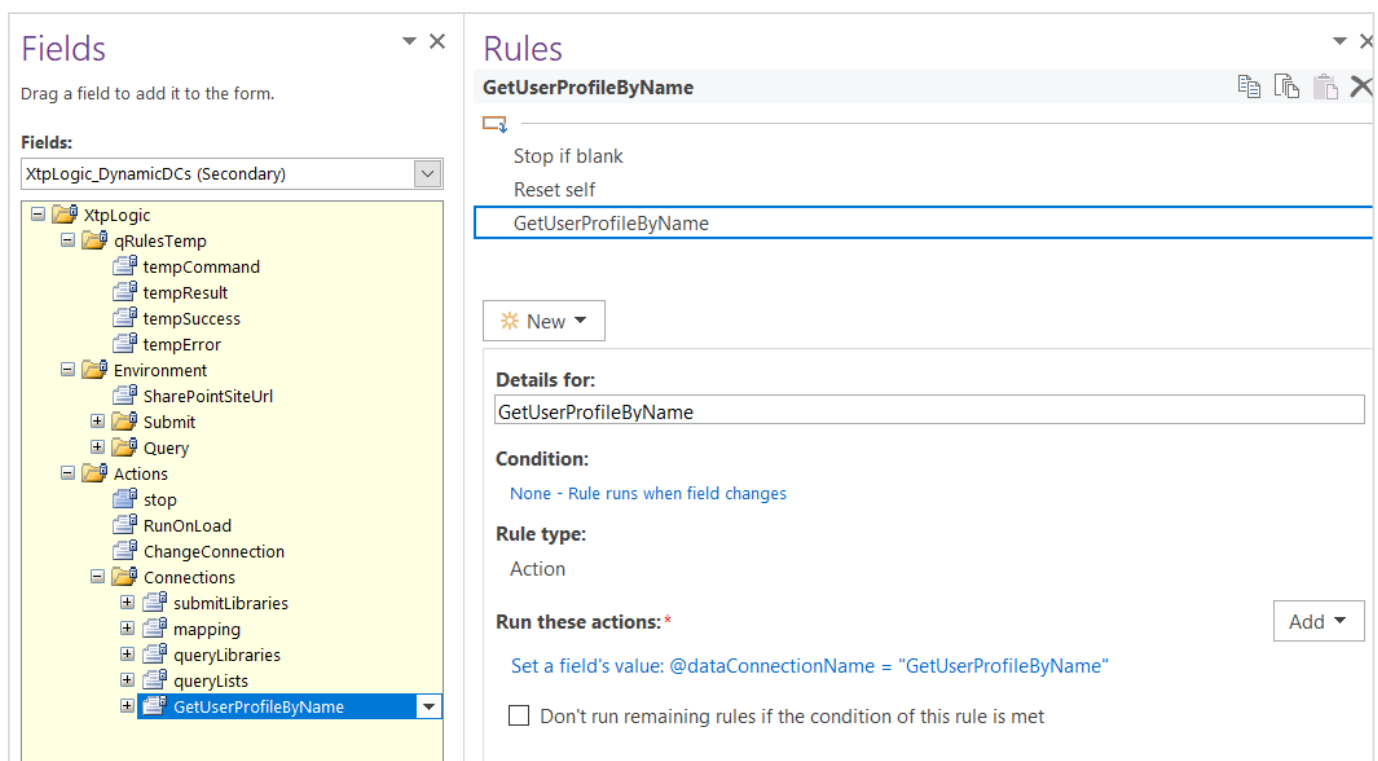
5. For the next action, modify '[Enter Library Name]' to the actual name of the SharePoint Library you are submitting to – Expense Report Library.



The 'Rule Details' dialog box is shown. It has a title bar with a question mark and a close button. The 'Action:' section has a dropdown menu with 'Set a field's value' selected. The 'Field:' section has a text box containing '@libraryName' and a list icon button. The 'Value:' section has a text box containing 'Expense Report Library' and a function icon button. At the bottom are 'OK' and 'Cancel' buttons.

This will automatically redirect the data to the target library located on the site or subsite where your FormsViewer is installed.

6. The `GetUserProfileByName` data connection name are often left as is. If your form's `GetUserProfileByName` data connection uses a different data connection name, proceed to:



The Form Designer interface is shown with two panes. The 'Fields' pane on the left has a title bar with a dropdown and a close button. It contains a 'Fields:' section with a dropdown menu showing 'XtpLogic_DynamicDCs (Secondary)'. Below is a tree view of the form's structure, including 'XtpLogic', 'qRulesTemp', 'Environment', 'Submit', 'Query', 'Actions', 'Connections', and 'GetUserProfileByName'. The 'Rules' pane on the right has a title bar with a dropdown and a close button. It shows a rule named 'GetUserProfileByName' with a list of actions: 'Stop if blank', 'Reset self', and 'GetUserProfileByName'. Below the rule list is a 'New' button. The 'Details for:' section shows the rule name. The 'Condition:' section shows 'None - Rule runs when field changes'. The 'Rule type:' section shows 'Action'. The 'Run these actions: *' section shows 'Set a field's value: @dataConnectionName = "GetUserProfileByName"' and a checkbox for 'Don't run remaining rules if the condition of this rule is met'. An 'Add' button is at the bottom right.

7. Delete the XTP section in the form canvas and the DynamicDCs group in the Main schema

The screenshot displays the Qdabra Software interface. On the left is a form titled "EXPENSE ITEMS" with columns: CATEGORY, DATE, DESCRIPTION, MERCHANT, RECEIPT, and AMOUNT. Below the table are fields for SUBTOTAL, LESS CASH ADVANCE, and TOTAL. A "Submit" button is at the bottom right. A message "XTP section deleted!" is visible. On the right, the "Fields" pane shows a tree structure with "DynamicDCs" selected. A "Microsoft InfoPath" dialog box is open, asking: "This change will rename, remove or modify a field stored in this form. Existing forms that have been filled out will lose the data stored in this field. Do you want to make the change?" with "Yes" and "No" buttons.

8. After making these changes, save the XSN.

Test the form

1. Upload the updated Expense Report form to FormsViewer. You may also try to upload the end form provided in Lab 2 → Lab 2 – *Dynamic DCs End.xsn*
2. Click the Hidden Debug button in the footer section – this is showing as the Form Solution Version at the bottom:


Close

EXPENSE REPORT

NAME

MANAGER

ACCOUNTING MANAGER

REPORT DATE
4/8/2020

BUSINESS PURPOSE*

EXPENSE ITEMS						FROM:	TO:
CATEGORY *	DATE *	DESCRIPTION	MERCHANT	RECEIPT	AMOUNT *		
▼ Select...				Attach file			
					SUBTOTAL		\$0.00
					LESS CASH ADVANCE		\$0.00
					TOTAL		\$0.00

Insert item

Submit

Solution Version: 1.0.1.1405

© 2020 Qdabra Software

- This will redirect you to the form's debug view. See how qRules runs the commands to change the data connection URLs on the fly:

Command		Result
GetInputParameter /key=SPHostUrl	☒ Success	https://qdabra.sharepoint.com/Training/FV2004
Delete /dsname=XtpLogic_DynamicDCs /xpath=/XtpLogic/Environment/Submit/Libraries/Library	☒ Success	
Delete /dsname=XtpLogic_DynamicDCs /xpath=/XtpLogic/Environment/Submit/Mappings/Mapping	☒ Success	
Delete /dsname=XtpLogic_DynamicDCs /xpath=/XtpLogic/Environment/Query/Libraries/Library	☒ Success	
Delete /dsname=XtpLogic_DynamicDCs /xpath=/XtpLogic/Environment/Query/Lists/List	☒ Success	
Insert /dsname=XtpLogic_DynamicDCs /parent=/XtpLogic/Environment/Submit/Libraries /child=Library	☒ Success	1
ChangeSubmitUrl /submit=SharePoint Library Submit /url=https://qdabra.sharepoint.com/Training/FV2004/Expense Report Library	☒ Success	https://qdabra.sharepoint.com/Training/FV2004/Expense Report Library
ChangeConnectionUrl /dsname=GetUserProfileByName /url=https://qdabra.sharepoint.com/Training/FV2004/_vti_bin/UserProfileService.asmx	☒ Success	https://qdabra.sharepoint.com/Training/FV2004/_vti_bin/UserProfileService.asmx
ChangeConnectionUrl /dsname=SharePoint List Submit /url=https://qdabra.sharepoint.com/Training/FV2004/_vti_bin/lists.asmx	☐ Success	

Site Url:
<https://qdabra.sharepoint.com/Training/FV2004/>

4. If you go back to the Data Connections, you'll see that these are originally pointing to a different site:

Data Connections
?
X

Data connections for the form template:

XtpLogic_DynamicDCs
GetUserProfileByName
QdabraRules
FormLogic
SharePoint Library Submit

Add...
Modify...
Remove
Convert to Connection File...

Details

Name: SharePoint Library Submit
Type: Submit data
SharePoint library or list: <https://qdabra.sharepoint.com/Training/FV1908/Expense%20Report%20Library/>
File name (expression): FormName
Allow overwrite if form exists: yes

5. Fill out the Expense Report form and submit – verify that it submitted to the Expense Report Library in your site.

Think about how you can leverage this XTP to your existing forms – you can now deploy your forms to dev or prod environment with less effort by just plugging this tool.

Contact Information

Qdabra Software
218 Main Street,
Suite 731,
Kirkland, WA
98033

Phone: 877.544.2389
Email: Support@Qdabra.com
Website: <http://www.qdabra.com>
Community: <http://www.infopathdev.com>