# CLONE AND CREATE FORMS

**PRODUCT:** qRules
**LAST UPDATED:** August 22, 2011

## INTRODUCTION

Three qRules command allow form developers to manipulate an InfoPath form's Processing Instructions (PI). These commands are InsertPi, RemovePi and RemoveDbxlPi. InsertPi inserts the specified PI into the DOM, while RemovePi command removes the specified PI from the DOM, if present. RemoveDbxlPi, as the name implies, removes the DBXL processing instructions and therefore requires DBXL. This tutorial will also explain the command Relink, also handy when cloning and creating forms.

## REQUIREMENTS

This tutorial assumes that you have qRules already installed on your machine. For details on this, please see the User Guide included with the installer.

1) Create a new InfoPath form: In InfoPath, design a new, blank form. Select **File** > **Save As** and save the XSN to your local drive. Close the form.

2) Launch the qRules Injector, browse to select the XSN you created and click **Inject**. Close the qRules Injector and open the form in Design mode.

For more information on the steps above, refer to the User Guide. This tutorial also assumes that user has the following:

- InfoPath 2007 or 2010
- DBXL v2.3 or higher
- SQL Server 2005 or 2008

## REMOVE ONE PI AND INSERT ANOTHER

Imagine that you have an InfoPath form that successfully submits to a SharePoint form library, and you wish to migrate to another library. qRules allows you to remove the PI and insert a new one, which is equivalent to changing the XSN that it references. In the steps below, we will illustrate this scenario with a simple example.

1. Create two SharePoint form libraries. We'll call them ShPFormLibA and ShPFormLibB in these steps.
2. Design a form that submits to ShPFormLibA. This form needs to have qRules.
3. Save it as FormA.xsn and publish it to ShPFormLibA. (For the steps necessary to publish a qRules form to SharePoint, please see the User Guide.)
4. Modify the form to change the data connection such that it submits to ShPFormLibB. Save the form as FormB.xsn and publish it to ShPFormLibB. This form does not need qRules.
5. Submit a test form in ShPFormLibA, and then download it locally.
6. Open the downloaded xml file in Notepad and note the PI:

***<?mso-infoPathSolution name="urn:schemas-microsoft-com:office:infopath:ShPFormLibA:-myXSD-2011-07-30T13-53-43" solutionVersion="1.0.0.2" productVersion="14.0.0.0" PIVersion="1.0.0.0" href="http://shp2010-test2/ShPFormLibA/Forms/template.xsn"?>***

This is what we will remove. But we also need to know what to replace it with.

7. Submit a test form in ShPFormLibB, then download it locally.
8. Open the downloaded xml file in Notepad and note the PI:

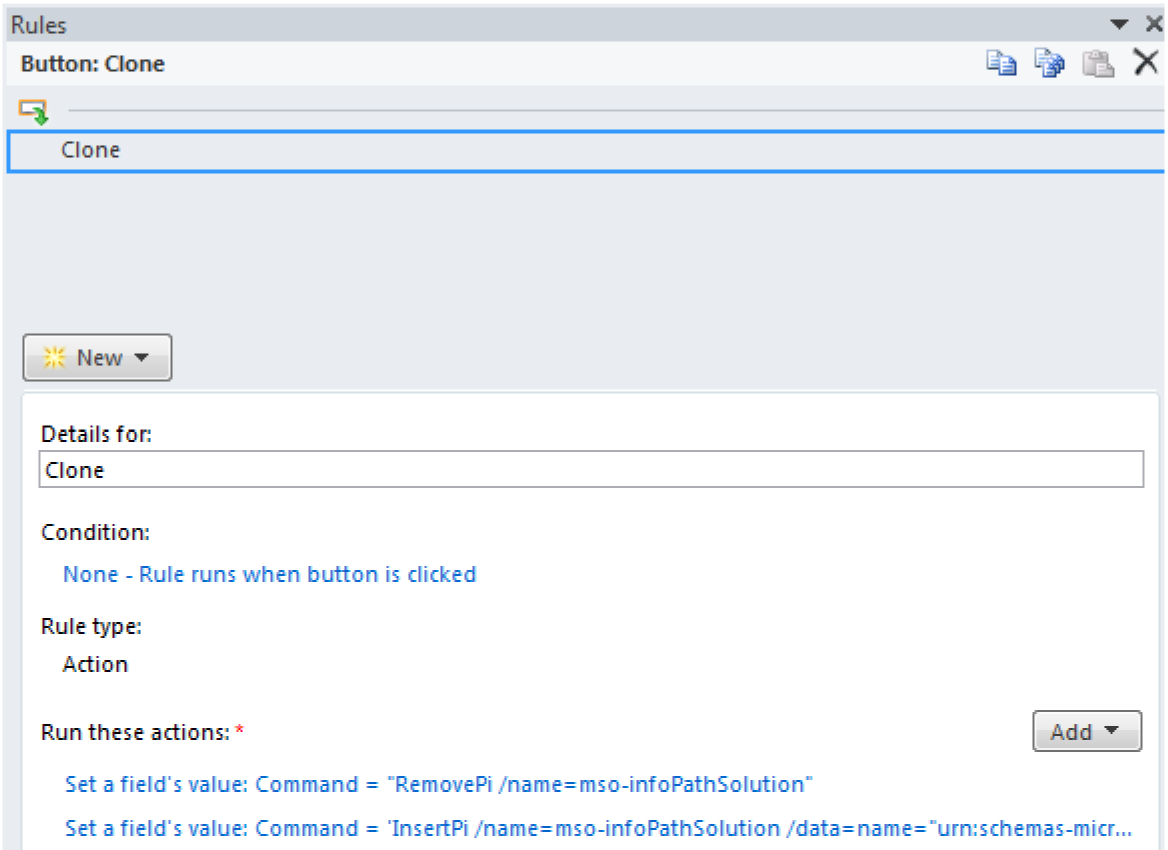
***<?mso-infoPathSolution name="urn:schemas-microsoft-com:office:infopath:ShPFormLibB:-myXSD-2011-07-30T13-53-43" solutionVersion="1.0.0.7" productVersion="14.0.0.0" PIVersion="1.0.0.0" href="http://shp2010-test2/ShPFormLibB/Forms/template.xsn"?>***

9. Add a button to FormA. Label it “Clone”.
10. Add an action to the button which sets the Command node in the QdabraRules secondary data source to the value:

***RemovePi /name=mso-infoPathSolution***

11. Add a second action that sets the Command node in QdabraRules secondary data source to the value:

***InsertPi /name=mso-infoPathSolution /data=name="urn:schemas-microsoft-com:office:infopath:ShPFormLibB:-myXSD-2011-07-30T13-53-43" solutionVersion="1.0.0.7" productVersion="14.0.0.0" PIVersion="1.0.0.0" href="http://shp2010-test2/ShPFormLibB/Forms/template.xsn"***



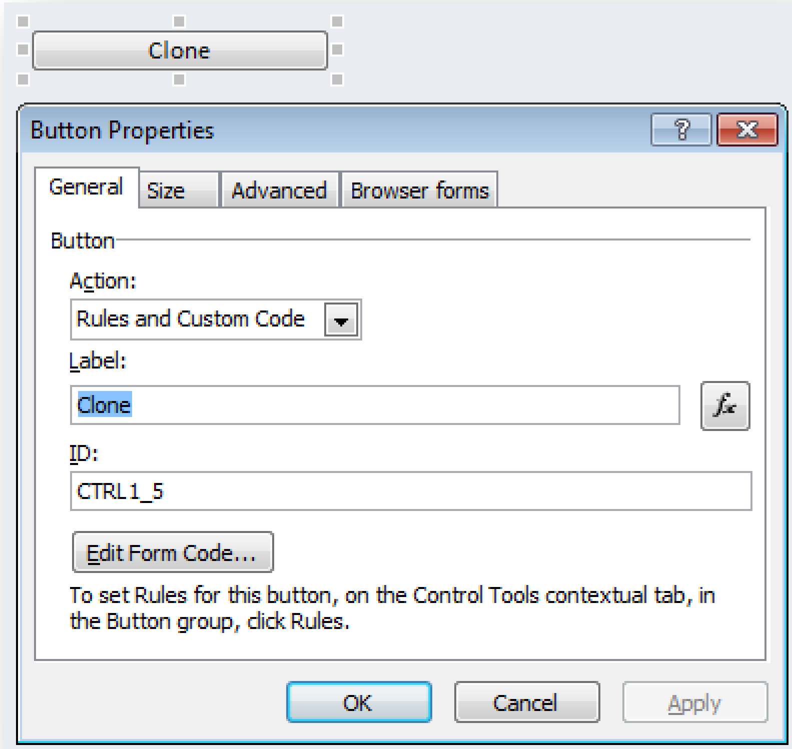

12. Add a third action that submits to a new data connection. This new data connection will submit to ShPFormLibB.
13. Add a fourth action to close the form.
14. Save the updated XSN and publish it to ShPFormLibA.

Now, if you open an existing form in ShPFormLibA and click Clone, the form's PI will get changed, and the form submitted to ShPFormLibB.

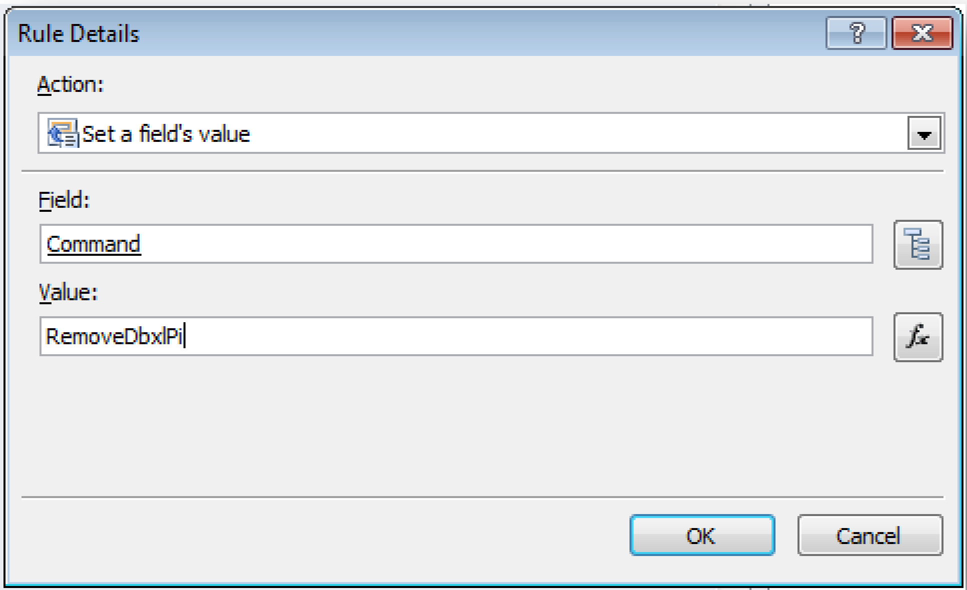
## Working with DBXL

RemoveDbxlPi can be used to clone forms that are stored in DBXL. The usage of the command is fairly easy, as there are no parameters. You can, for example:

1. Add a button and label it "Clone."




2. Add a rule to the button. In this rule, one action will set the Command node in the QdabraRules secondary data source to "RemoveDbxlPi".



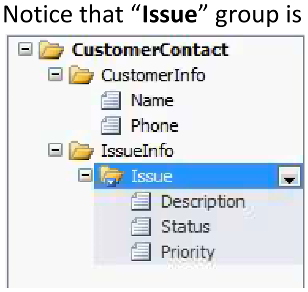

When the user clicks the button, the DBXL PI will be removed. This means that the existing xml will not be overwritten. If the user then clicks a button that submits to DBXL, the form will be submitted to DBXL but with a new Document ID (DocID). This allows you to create a new document based on an existing document.

## RELINK

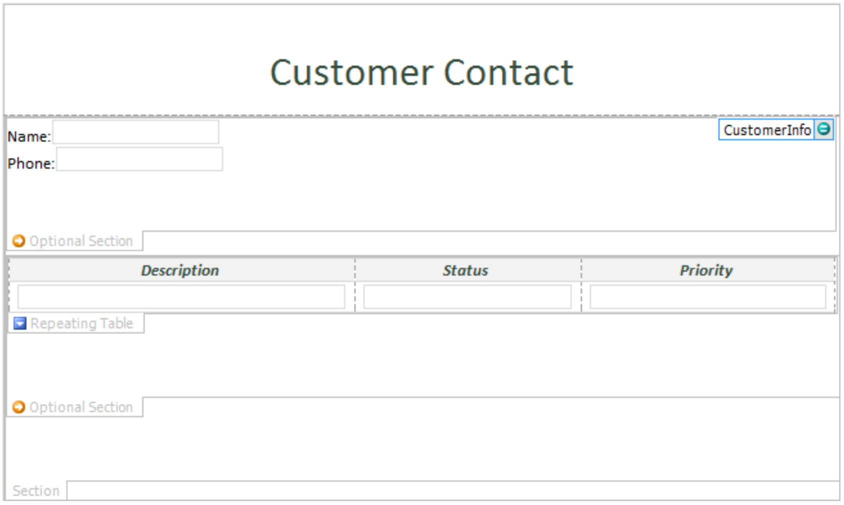
Imagine that you have InfoPath forms created in different versions. Want to have old forms shown in old form template and new forms in new form template? Follow this tutorial and you will be able to create side-by-side versioning of InfoPath form using Content Types.

1. Open a blank form in design mode.

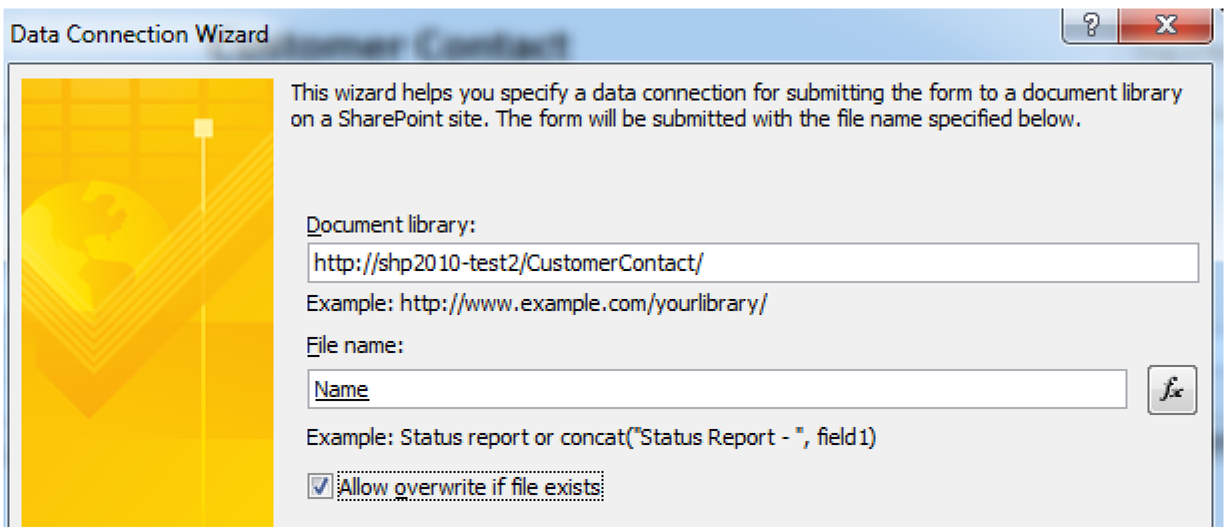
2. Create the following fields in the form. Notice that "**Issue**" group is a repeating section.




3. Drag and drop the fields into the form. Now we are done with creating the form template. Your form will look like this

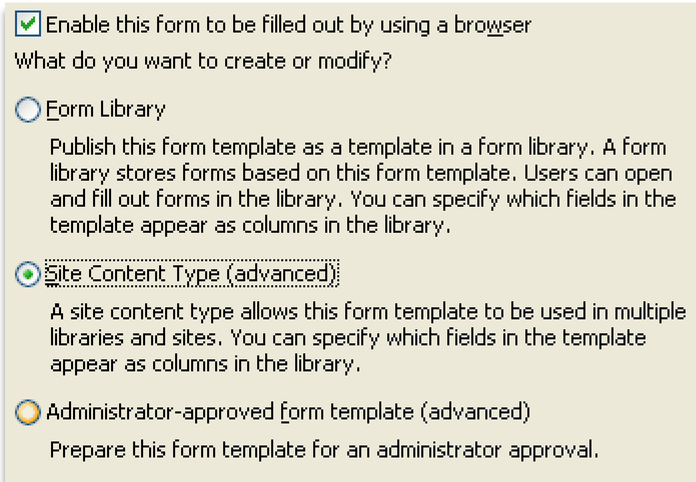



4. Create a Form Library in the SharePoint site called "**CustomerContact**".
5. Add a button which has a submit data connection that submits to the "**CustomerContact**" Form Library in the SharePoint site.



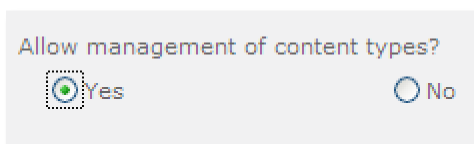

6. Save the form template locally as "**CustomerContact-v1.xsn**". Let's consider this as version 1 form.

7. Publish this version 1 form to SharePoint site and fill the form.
    a.
    b. Go to your InfoPath form template and publish your form to a SharePoint Library.
    c. Enter the location of your Form Library you just created, click **Next**.
    d. The key thing here is to choose "**Site Content Type**", click **Next.**

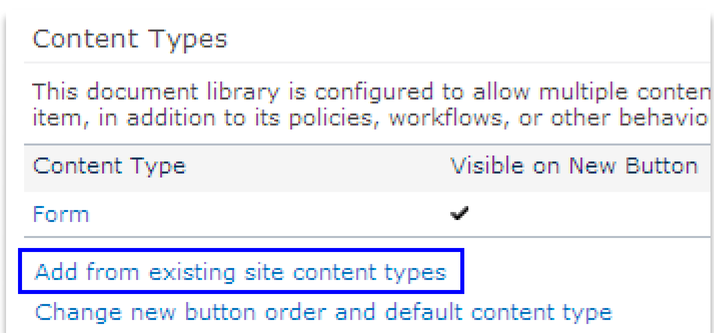



e. Select "**Create a new Content Type**" and click **Next.**
f. Enter the name of the Content Type as "**CustomerContact-v1**".
g. Enter the SharePoint location ("**CustomerContact**" library) and filename ("**CustomerContact-v1.xsn**") for the form template (e.g. http://<your-site>/CustomerContact/Forms/CustomerContact-v1.xsn). Click **Next**.
h. Add the fields you want to promote to the site. Click **Next** and then **Publish**.

8. Add the content type to your SharePoint Library
   a. Go to your SharePoint Form Library -> Settings -> Form Library Settings -> Advance Settings. Select **Yes** to **Allow management of content types**, and click **OK**.

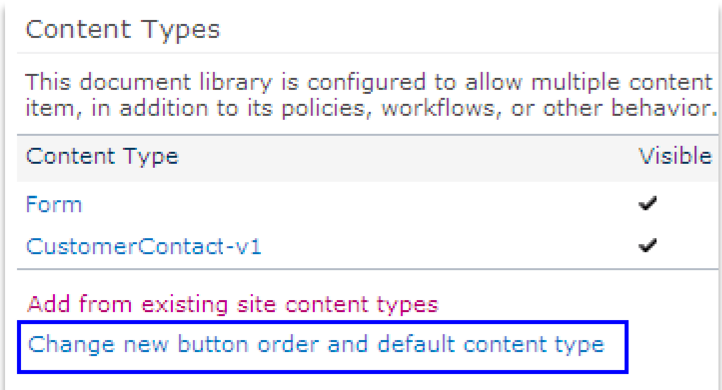



   b. In the Form Library Settings page, under **Content Types**, click on "**Add from existing site content types**".

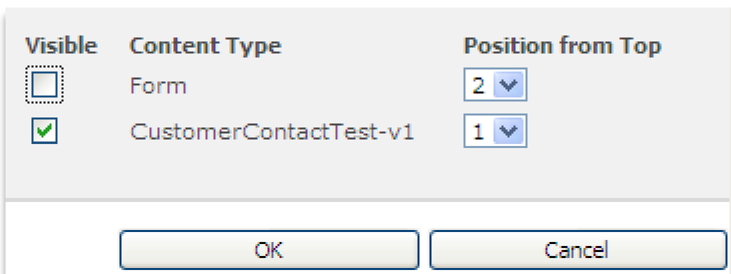


c. Select “**CustomerContact-v1”** and click **Add**. Click **OK**.

d. Back to the form library settings, click “**Change new button order and default content type”**.




e. Choose “**CustomerContact-v1”** as Position 1, which makes it the default content type. Uncheck the visible check box for “Form” and then click **OK**.



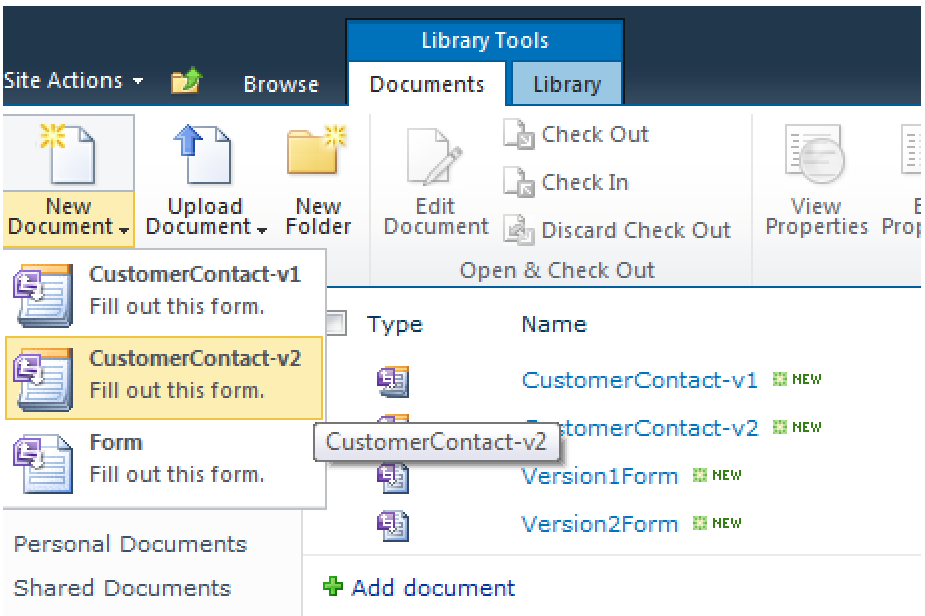

f. Now, click on the “**CustomerContact**” library and click “**Add Document**”

g. Fill out the Customer Contact form and submit it.

9. Create next version (version 2) of the form and name it “**CustomerContact-v2.xsn**”, by following these steps:
    a. Change  a bit of the schema by adding a new field called “**Company**” under “**CustomerInfo**” and make it a required field. Add another new field “**DateTime**” (of datatype DateTime) under “**Issue**” and enter **now()** as the default value for the field.
    b. Clear the previous display, and drag & drop the new schema into the form.
    c. Change the control of the **DateTime** field to calculated value (Expression Box).
    d. Save the form template as “**CustomerContact-v2.xsn**”.

10. Publish this new version (version 2) form to a new SharePoint site– specify a new location for CustomerContact-v2 by following the steps in 5-c through 5-e
    a. Enter the name of the Content Type as “**CustomerContact-v2**”.
    b. Enter the location of your Form Library and the name of the Form Template and publish.

c. Now go to your SharePoint Form Library -> Settings -> Form Library Settings page and under **Content Types** click on "**Add from existing site content types**".
d. Select "**CustomerContact-v2"** and click **Add.**
e. Click "**Change new button order and default content type"** and chose "**CustomerContact-v2"** as Position 2.
f. Now, Click on the "**CustomerContact**" library and click "**Add Document**". Under the New Document (as shown below) you can see that we can choose the templates from both versions. Choose Version 2 template "**CustomerContact-v2"** , fill out the Customer Contact form and save it as "**Version2Form**".




11. Now we have two versions of the InfoPath forms both using their own templates to save information.

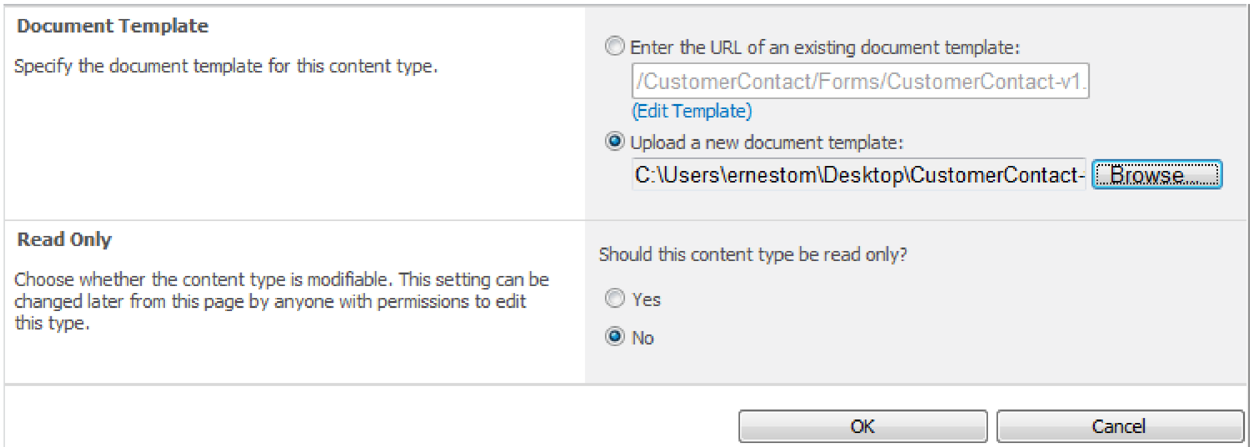
How can qRules help? Imagine that you have an existing set of documents that were submitted using CustomerContact-v1.xsn. You can allow users to selectively relink those documents to the new version.

12. Inject CustomerContact-v1.xsn with qRules.
13. Open CustomerContact-v1.xsn in design mode.
14. Add a button and label it **Relink and Submit**.
15. Add a rule to this button which sets the **Command** node in the qRules secondary data source to:

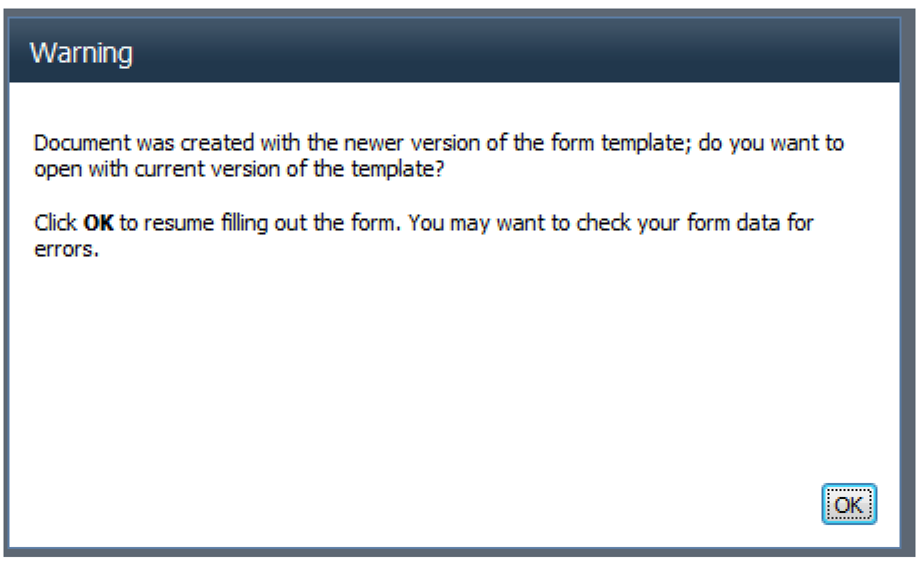
Relink /href=***http://shp2010-test2/Shared%20Documents/CustomerContact-v2.xsn*** (where the URL is the location of the second version of the template, which was published in step 10b.

16. Add a second rule which submits to the SharePoint form library. This is the same data connection that was added in step 5. You may also add a rule that closes the form, if desired.
17. Save and republish the form.

Note that at this point you might get an error in publishing the form. That's because the content type is in use. The work around is to go to Form Library Settings, click on CustomerContact-v1, click on Advanced Settings, and upload the new template via the browser page as seen below.




Back in your form library, you can click an existing document, and then click the Relink and Submit button. This will Relink THAT document to the new template. When that document is reopened, you will see a warning:




After clicking OK, you will see that the form opens using the new template.

## SUPPORT

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

Licensed customers can contact us via **Support@Qdabra.com**.

You can also use the InfoPathDev.com Qdabra Product support forums to request help from the community.

## *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