



INFOPATH TO K2 SMARTFORMS MIGRATION

2013-11-15



Contents

Introduction.....	2
Step 1: Create SmartObject data-model.....	3
Line Of Business for data-model	3
Converting the Main data-source.....	3
Converting Secondary data-sources.....	5
SmartObject Method data-source	5
SOAP Web Service.....	5
REST Web Service	5
SharePoint List or Library	5
SQL Server database.....	5
XML	6
Step 2: Design Views and Forms.....	7
Inspect InfoPath Form	7
Laying out the foundation for the SmartForm – Views	7
Assemble the Views - SmartForms.....	8
Step 3: Build New Process	9



INTRODUCTION

This document provides high-level guidelines to migrate from an existing InfoPath centric K2 application to a K2 smartforms based form technology.

This document assumes that the reader is familiar with InfoPath and how it integrates into K2 blackpearl process definitions, as well as experienced with K2 smartforms and SmartObject technologies. The target audience is anyone tasked with determining an approach on how to move from InfoPath based forms to K2 smartforms.

NOTE: *If you do not possess the appropriate level of K2 experience and/or need assistance in planning and implementing your K2 smartforms solution, please contact your local K2 Consulting Services in order to understand what K2 product training and engagement offerings are available to assist with your needs.*

There is no ability to directly convert an InfoPath form to a K2 smartform, this document focuses on the steps involved in the migration of an InfoPath application to SmartForms:

1. Create SmartObject data-model to represent InfoPath data-sources.
2. Design Views and Forms to represent the new user-interface.
3. Build a new process definition (workflow) to use the SmartObjects and Forms generated by step 1 and 2.

This is a migration and not a conversion and will result in a new set of forms and workflows being deployed, permitting in-flight process instances to complete in the technology and architecture in which they were designed.



STEP 1: CREATE SMARTOBJECT DATA-MODEL

The first step in migrating the InfoPath application to K2 smartforms, is to create a SmartObject based data-model for the data-sources defined in the InfoPath template (XSN) document. The reason for this is that K2 smartforms use SmartObjects as its only data application layer (DAL) technology.

In the InfoPath template document, the following data-sources are present:

- **Main** – this is the XML based data tree, and the root of this tree is usually named 'myFields'. This XML data tree, or document, is the information sent to the process instance when it is started, and also used to populate the InfoPath template when a user opens a task created by a Client Event.
- **Secondary** – this is the integration defined for the InfoPath template. The following options are available:
 - SmartObject Method
 - SOAP Web Service
 - REST Web Service
 - SharePoint Library or List
 - Database (SQL Server)
 - XML document

Each of these data-sources need to be converted to use the SmartObject technology, with the exception of the *SmartObject Method* secondary data-source.

LINE OF BUSINESS FOR DATA-MODEL

Before the SmartObjects can be created, a line-of-business (LOB) system will have to be selected to contain the data-model. Using a relational database management system (RDMS) such as SQL Server and Oracle are great options, since it provides the designer with the flexibility to map out the data-model in any structure that is needed, while also providing good performance for runtime. SharePoint Lists can also be used, but is not recommended for transactional data.

For the purpose of this document, we will use SmartBox SmartObjects in order to keep this guide simple.

CONVERTING THE MAIN DATA-SOURCE

Start creating the new data-model in SmartObjects as follows:

1. Examine the entire XML tree structure of the main data-source of the InfoPath template – i.e. expand all groups (folders) and subgroups (sub-folder).
2. Decide how to group all these fields in business entities, and create a [logical data model \(LDM\)](#).

Things to keep in mind when creating the LDM:

- There should be 1 'master' business entity for the application. This business entity should contain the most important information.



- All other business entities should have a logical relationship with this business entity – i.e. contain a field that stores the ‘master’ business entity’s unique identifier.
- All business entities should have a unique identifier – use either an integer or GUID value.
- The business entities should have short, but descriptive names.
- Don’t give business entities more than 20 fields. Rather break them up into separate business entities.
- The field data types of the business entity should be the same as those in the main data-source of the InfoPath template.
- Consolidate groups and subgroups where possible.
- Repeating groups/sections should always be a separate business entity from the master business entity.
- Only include business entity fields in the LDM that should be stored for reporting or auditing purposes.
- Do not include the fields of the `_K2` group in the LDM.
- Do not include fields (in the LDM) used only for:
 - Integration, such as the SharePoint URL or Library name.
 - Expressions, such as showing totals or averages (unless they should be stored)
 - Rules, such as hidden values to calculate results for logical rules.

Once the LDM has been clearly defined, it can be implemented into a physical data model (PDM). This requires the following steps:

1. Create the structure of the PDM in the preferred LOB.
 - a. For example, if SQL Server or Oracle is used, then the PDM will be implemented as tables for the entities, with primary keys (for unique identifiers) and foreign keys (for relationships).
2. Register a Service Instance (using the SmartObject Service Tester tool) to connect to the new data-model.
 - a. NOTE: if K2 doesn’t provide an out-of-the-box (OOTB) Service Broker for the LOB system, then a custom Service Broker will have to be built prior to registering the Service Instance.
3. Using the new Service Instance, create a SmartObject for each business entity in the PDM, in a K2 design tool such as *K2 Studio* or *K2 for Visual Studio*.

Once the PDM has been built in SmartObjects, these SmartObjects can then be deployed to the K2 Server, and Views and Forms can be built.



CONVERTING SECONDARY DATA-SOURCES

'Converting' the secondary data-source is much simpler than the main data-source, as the LOB system already exists. It is just a matter of connecting them with K2, and building SmartObjects for them.

SmartObject Method data-source

There are no additional steps required, since the SmartObjects already exists. They can just be reused for SmartForms.

SOAP Web Service

K2 provides an OOTB Service Broker that can be used to connect to SOAP (aka XML) web services, and consume them in K2 as SmartObjects.

Here are the steps required to convert SOAP web service data-sources:

1. Register a Service Instance (using the SmartObject Service Tester tool) of the *Endpoints WebService* type to connect to the SOAP web service.
2. Using the new Service Instance, create a SmartObject for the methods that were used. It would be best to consolidate multiple web service methods into a single SmartObject, but can also be split into multiple SmartObjects.

REST Web Service

K2 currently does not have a Service broker for REST endpoints, however, it is on the product roadmap for a future release. If REST endpoints are used, then they can be wrapped by a WCF web service, which can then be used via the *Endpoints WCF Service Broker* that ships OOTB with K2 blackpearl.

SharePoint List or Library

K2 provides an OOTB Service Broker that can be used to connect to SharePoint Lists and Libraries, and consume them in K2 as SmartObjects.

Here are the steps required to convert SharePoint List/Library data-sources:

1. Register a Service Instance (using the SmartObject Service Tester tool) of the *SharePoint Content* type, to SharePoint site that hosts the list or library.
2. Using the new Service Instance, create a SmartObject for each list and library that were used.

SQL Server database

K2 provides an OOTB Service Broker that can be used to connect to a SQL Server database, and consume them in K2 as SmartObjects.

Here are the steps required to convert SharePoint List/Library data-sources:

1. Register a Service Instance (using the SmartObject Service Tester tool) of the *SQL Server Service* type, to SQL Server instance that hosts the database.
2. Using the new Service Instance, create a SmartObject for each table, view and stored procedure that were used. It would be best to consolidate multiple tables,



views, and stored procedures into a single SmartObject (where it makes sense), but can also be split into multiple SmartObjects.

XML

K2 currently does not have a Service broker for XML documents. If XML documents are used, then they can be wrapped by a WCF web service, which can then be used via the *Endpoints WCF Service Broker* that ships OOTB with K2 blackpearl.



STEP 2: DESIGN VIEWS AND FORMS

INSPECT INFOPATH FORM

Start with the submit InfoPath view, usually the default view, to determine View layouts.

1. Note different sections, repeating sections/tables.
 - a. Separate data sources should have already been provisioned in the LDM as per above for each of these. Repeating sections potentially already have an underlying secondary data source.
2. Note drop-downs, people pickers.
 - a. Note the current underlying data source for the drop-downs and people pickers.
3. Note structure, images and content.
4. Step through each View of the InfoPath form.
 - a. See the natural progression and note any sections, repeating sections/tables, drop-downs, people pickers as well as the task action section i.e. Action Buttons, drop-downs, calendars, comments fields etc.

LAYING OUT THE FOUNDATION FOR THE SMARTFORM – VIEWS

The first step is to build out the K2 smartform Views using the information collected above, trying to mirror the InfoPath form, and also our Logical Data Model.

Things to keep in mind when building views:

- In K2 smartforms the concept of Views differs from InfoPath Views as different SmartForms Views are tied to different data sources (SmartObjects) and a SmartForms can contain multiple Views on a single display, which can be shown or hidden, where InfoPath Views are typically all tied to the same data source(s) and there can only be one View per display and different Views need to be “paged” to using rules.
- There will typically not be a 1:1 mapping of InfoPath Views to SmartForms Views but rather sections/tables/repeating sections-tables will have a more logical mapping to SmartForms Views.
- Build Views based on LDM defined earlier.
 - Item Views to capture data
 - List Views to display data
 - Editable List Views to capture “repeating data” in row format.
 - Capturing comments is a good example
 - Flat, no structure
 - More on structured repeating data later
- Structure is defined using Main Table, table controls and Views.
- Drop-down lists (DDL) can be “cascaded” OOTB with K2 smartforms.
 - DDL can be filtered based on a value of another DDL on the same form
- Picker control is the equivalent of the People Picker in InfoPath.
 - Any SmartObject with a list method may be tied to a Picker Control
 - Not constrained to AD Users.
- Repeating tables/sections(structured)



It has to be noted that there are different ways to address repeating sections in K2 smartforms. Common solutions are:

- Master/Detail
 - 2 Views against the same data source
 - 1 View is a list View to list all items added
 - 1 View is an item View to add items and item details
 - Views are on the same Form and can both be seen at the same time
- List/Pop-up
 - Similar to the Master Detail
 - The Item View is a pop-up Sub-form
- Once you are ready to start building your Views it is a good idea to have your InfoPath form open in design mode on one monitor and your K2 smartforms Designer on another for quick reference.
- If different controls are desired for the same data source fields, DDL should be textbox on subsequent displays to end-users, then different Views need to be constructed OR if only a few controls are required to be different both controls, DDL and textbox, can be placed on the same View and their Visibility property maybe toggled (Show/Hide) using SmartForms rules/actions.
- Usually a separate View is built for the Workflow/Action section.

ASSEMBLE THE VIEWS - SMARTFORMS

When the majority of Views are built a Form can be assembled. A Form is the framework into which the pre-built Views will be dropped.

Things to keep in mind when assembling forms:

- Multiple forms may be used for one solution
 - Sometimes it is necessary to make use of pop Sub-forms
- Views can be arranged vertically or horizontally
- Views will be displayed in the order that they are added when visible
- Form level rules tie Views together on a Form
- Different States define different “behaviors” of Forms/Views/Controls
 - More on states in the process section
- Form Validation rules can be used to do a “full-sweep” validation on Control events(like button click)
 - Validation can be done across all Views in one rule
 - It can be set that only controls that are visible, enabled and not read-only be validated



STEP 3: BUILD NEW PROCESS

The final step in the refactoring process is to rebuild the process. It is recommend to keep the below considerations in mind when modifying the process:

- Investigate and note the following in the InfoPath process.
 - Outcomes/Actions for activities with client events
 - Notifications on client events
 - Event level escalations
- Create a new KPRX.
 - A suffix may be appended to the new process to distinguish between processes
 - Old – ChangeRequest
 - New – ChangeRequest.SF
- Use multiple instances of K2 for Visual Studio/K2Studio to display the process on different monitors for easy reference.
- Choose to integrate processes with K2 smartforms Process integration or adding an Item reference or not.
- A standard reference to Load method of main Data Source could also be used if K2 smartform Process Integration is not used.
- Copy Activities, one by one (for systematic approach), from old process to new process.
 - Remove InfoPath client event
 - Artifacts that stay intact after removal
 - Destination rules
 - Activity escalation rules
 - Add a SmartF Client Event to the activity
 - Select the Form created earlier
 - Create a new state for the activity via the Wizard for each client or
 - Use a single state with additional logic to determine which Worklist Item to open and Submit
 - Pass Data to the form
 - Pass Data to the process
 - Reconfigure the notifications
 - Reconfigure Escalations