

# Integrating Microsoft Office 2007 Form Solutions with SAP Web Services

## Authors

Thomas Reimer, Enterprise Technology Strategist, Microsoft EMEA,  
[treimer@microsoft.com](mailto:treimer@microsoft.com)

Juergen Daiberl, Sr. Technical Evangelist, Microsoft Corporation,  
[jdaiberl@microsoft.com](mailto:jdaiberl@microsoft.com)

Tilo Boettcher, Sr. Program Manager, Microsoft Corporation,  
[tilo.boettcher@microsoft.com](mailto:tilo.boettcher@microsoft.com)

## Summary

This paper describes several scalable options for online and offline SAP form solutions. The intent of this paper is to provide suggestions for how to build a forms solution with SAP Web services and Microsoft Office InfoPath® and Microsoft Office Forms Services as part of Microsoft Office SharePoint® Technologies.

## Copyright

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred.

© 2008 Microsoft Corporation. All rights reserved.

Microsoft, Windows, BizTalk, InfoPath, Active Directory, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

## **Applies To**

- Microsoft Office SharePoint Server 2007
- Microsoft Office InfoPath 2007
- Microsoft BizTalk Server 2006 R2
- Microsoft Visual Studio 2005
- SAP NetWeaver

## **Keywords**

Microsoft Office, Microsoft Office InfoPath, Microsoft Office SharePoint Server, Microsoft SAP Integration Scenario, Forms Solution, SAP Web services, SAP Enterprise Services, Business Data Catalog, Enterprise Forms Solution

## **Contact**

This document is provided to you by Microsoft Corporation. Please check the SAP interoperability area at: [www.microsoft.com/sap](http://www.microsoft.com/sap) and the .NET interoperability area in the SAP Developer Network at: <http://sdn.sap.com> for updates or additional information.

## Contents

Copyright .....	ii
Introduction .....	1
Best Practices for InfoPath-based Forms Solutions .....	2
Connecting Forms Solutions to the SAP System .....	4
Using SAP Web Services .....	4
Using SAP Enterprise Services.....	6
Starting Simple – Using InfoPath 2007 as a Forms client .....	7
Sample Process Flow .....	7
Extending the Forms Client Solution – Adding SharePoint Technologies for Scalability .....	10
Sample Process Flow .....	10
Creating a Complete SOA Compliant Managed Forms Solution .....	12
Sample Process Flow .....	12
An InfoPath Forms Solution .....	15
InfoPath Forms Solution Implementation.....	26
Summary.....	39
Resources.....	40
Official Microsoft   SAP Customer Information Center: .....	40
Microsoft and SAP Interoperability White Papers: .....	40

## Introduction

By using XML based electronic forms from Microsoft InfoPath® with SharePoint Server services that are connected to SAP core modules through Web services, productivity, accuracy, and data consistency are significantly improved. Specific benefits include:

- Simplified form access through a central forms repository
- Low total cost of ownership with form administration, deployment, and duplication
- Quick and easy forms solution creation
- Microsoft forms solutions are already licensed in an Enterprise CAL
- Strong end-to-end solution for a dual vendor strategy (SAP and Microsoft)
- A direct connection to SAP legacy and Service-Oriented Architecture (SOA) services with no third party tools by using InfoPath, SharePoint Server, BizTalk® Server, or SAP NetWeaver services
- Form solution operation and management cost savings by using integrated server solutions like Microsoft Systems Center and SAP NetWeaver Solutions Manager
- Microsoft form solutions based on XML and SOA

Since requirements do vary, this paper will focus on the following three scenarios of increasing complexity:

- 1) Lightweight Office InfoPath 2007 forms solution with SAP Web services
- 2) Scalable forms solution with SAP Web Services Access
- 3) Managed forms solution with transactional support

## **Best Practices for InfoPath-based Forms Solutions**

This paper will describe a set of technical solutions designed to bring the right SAP data to the right form user. As a starting point, it is important to understand some basic best practice skills for creating forms solutions in general as well as specifically for SAP.

Forms are used to display structured and unstructured data in a readable format. It is critical that forms are designed to be easily used and understood when displaying information or guiding the user through a sequence of steps in order to effectively complete the form. Often we see forms that are poorly designed and have too much information included on the same page. For example, sometimes in addition to the data, the descriptive and help information are placed on the same page.

The manner in which the form is used must be considered as well. For example, will the form be printed and completed manually, completed online, used on a Tablet PC, or will it be used in multiple formats? Therefore, we need to provide a solution that meets all needs while keeping the information structured and readable with enough space for handwritten insertions.

Forms based on SAP data have the same requirements, but the information must be more structured. This is because SAP services can hold huge amounts of fields and properties (often over 100) when using Web services, Business Application Programming Interfaces (BAPIs), or Intermediate Documents (IDOCs) as a transport mechanism. In most cases, the data is not filtered so we have to deal with lots of entities or data structures. Because of this, our preferred method is to create a process that encapsulates the end points in BizTalk to handle transport, mapping, filtering, and transactions. The SharePoint Server Business Data Catalog (BDC) can handle this type of encapsulation as long as an application definition is defined in the SharePoint Server SDK BDC Editor and imported as a shared application. Currently, the BDC is read-only.

During architectural discussions, Microsoft customers regularly request to build a new style of Web applications using InfoPath capabilities to easily define data sources for databases and Web services. Please be aware that there are some downsides to these solutions. InfoPath has its strength in pattern or forms-oriented design where data can be displayed, verified, signed, and submitted or received, to or from external data sources. However, InfoPath is neither a development environment nor a framework for the Web applications that host the application logic or scalability components.

When you build solutions around SAP, most of the logic will reside in SAP. Therefore, it is important to engage SAP specialists early in the planning process. Also, since external access to the SAP system has to be tested and approved, consider having an executive sponsor to support cross departmental work.

Best practice techniques are listed in the following table.

Do	Avoid
Keep views small (screen size/print form)	Scripting functionality
Use view switching functionality	Significant On_Load processing
Use tables to align controls	Queries that rarely change result data in drop-down list boxes
Set control width to 100% so it scales with column width	Task panes if forms need to appear in the browser
Reuse and adapt existing forms for online completion	Complex forms with 20 or more views
Put instructions in the form layout	Using InfoPath forms as a Visual Basic replacement
Use all the validation user interface notifications and create error messages	Adding a lot of custom code to forms
Build a form design kit with template parts	
Use pre-configured data connections through the data connection library	
Use predefined schemas when building forms	

**Table 1: Best Practice for Forms Design**

## Connecting Forms Solutions to the SAP System

All form solutions discussed in this paper are Service-Oriented Architecture (SOA) compliant. This compliance enables the Web services to be a transport layer and enables the SAP endpoints to handle solution services. Although there are several ways to connect form solutions to the SAP system, we will focus specifically on using InfoPath and BizTalk.

If we use InfoPath alone, we can only connect to SAP side Web Services. This is because InfoPath can only connect to the SAP system by using Web Services or by using ADO.NET database connections to connect to SAP data sources. This can be an issue if an enterprise's existing SAP interfaces require the use of legacy endpoints like BAPIs, RFC calls, or IDOCS that InfoPath cannot access. IDOCS are still used since they are a good way to transport information in an asynchronous and transactional way.

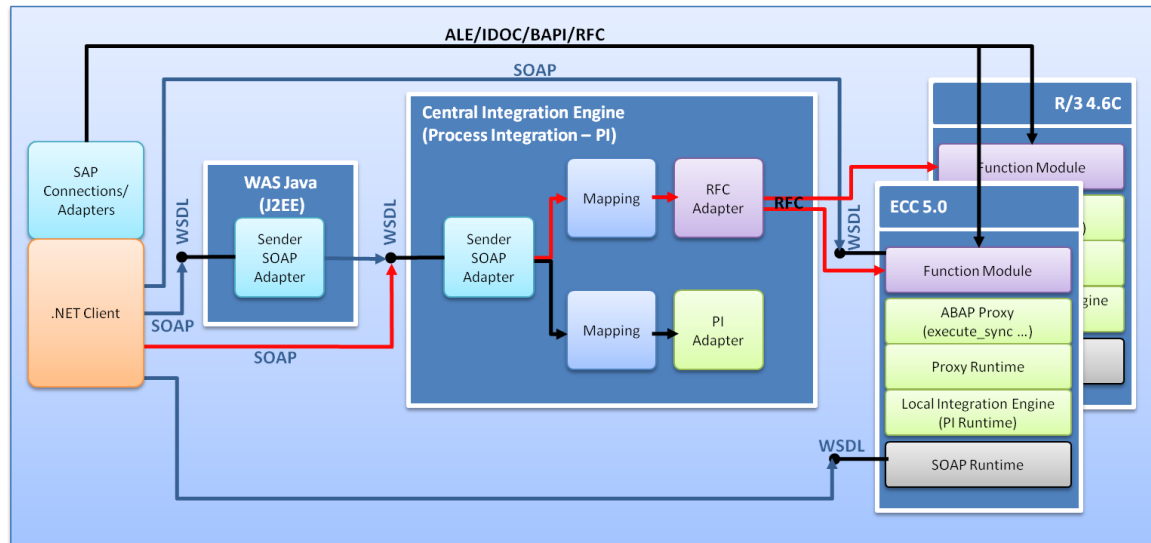
An alternative option would be to use a managed middleware component like BizTalk Server to add a central managed layer to the solution. This extends connectivity to legacy SAP systems such as the SAP Application Server versions prior to 4.6C and 4.7, and provides access to BAPIs, RFCs, IDOCS, and tRFCs. Access to RFCs and BAPIs are especially important since they exist in many SAP business transactions. We have chosen BizTalk Server 2006 R2 as the preferred solution to handle SAP system endpoints.

The next two sections will describe the differences between using SAP Web Services and SAP Enterprise Services to connect to the SAP system.

### Using SAP Web Services

The SAP NetWeaver Application Server versions 6.40 and higher come with a built-in infrastructure that supports Web Services. This infrastructure implements Web Services standards like SOAP 1.1, WSDL, UDDI, WS-I BP 1.0, and WS-Security 1.0. In addition, the Web Service toolset provides a wizard-driven Web Service interface configuration, WSDLs, security, and technical communication features. This toolset also supports Web Service publishing in UDDI registries. As shown in the following Figure 1, SAP Web Services expose existing BAPIs and RFCs to the outside world using the SOAP 1.1 standard.





**Figure 1: .NET Client to SAP R/3 and NetWeaver System Connection Options**

Systems like ECC5 and ECC6 are based on newer implementations of the Web services stack and work out-of-the-box for InfoPath forms solutions. Simply insert the URL of the Web service in your InfoPath Web services wizard and define the set of properties to use to start your first SAP Web services based forms application. Keep in mind that not all companies expose all functions as Web services. There is also several administrative side security and authentication related rules that might limit service access. For complete step-by-step instructions on how to implement a Web service in an SAP environment, please see the Microsoft white paper, "Building .NET Apps using Visual Studio® 2005 and SAP NetWeaver Web Services"<sup>1</sup>.

Web Services Description Language (WSDL) is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. WSDL files can be created in two different ways; as a RPC style or as a DOC style. The RPC style cannot be processed out-of-the-box in InfoPath today. RPC style WSDLs can only be handled with additional code incorporated into the form. However, the WSDL description can be handled in BizTalk Server which is capable of importing custom WSDL descriptions without issues.

When implementing a Web service, the most important item to address is the process description. This is because there are several properties to set. Some of them are mandatory and others can be skipped. Furthermore, the format is important to consider as it is sensitive to all properties including upper and lower case text.

<sup>1</sup>Building .NET Apps using Visual Studio 2005 and SAP NetWeaver Web Services"

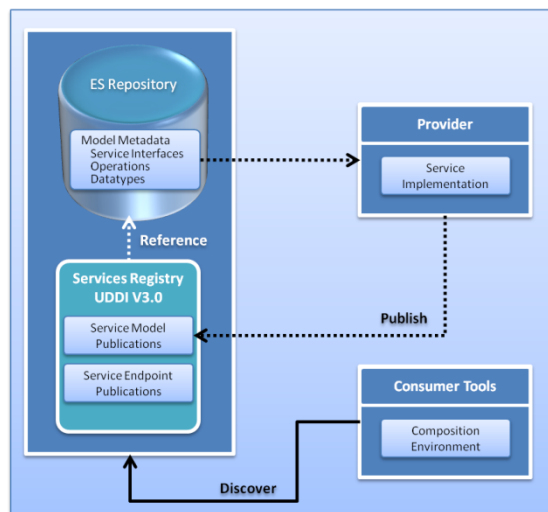
<http://download.microsoft.com/download/5/7/f/57f1490e-8a8d-497b-bbae-ec2a44b3799f/SAPNetweaverWebServicesVS2005.pdf>.

## Using SAP Enterprise Services

SAP Enterprise Services are standards-based Web applications that enable the encapsulation of enterprise functionality and expose it as a reusable business service that can be combined with other services to meet new requirements. Enterprise services can be assembled together to compose new applications or enable new business processes.

Enterprise services are implemented as Web services. This means that their interfaces are defined using WSDL files and that they support other commonly used Web service standards.

A description of SAP Enterprise Services, including the WSDL files, and process component models that show how services work together to automate processes and business scenarios, are contained in the Enterprise Services Repository as shown in Figure 2 below. This repository is a design-time collection of meta data about enterprise services that is used by development tools.<sup>2</sup>



**Figure 2: Enterprise Services Repository Connection using UDDI Compliant Calls**

Enterprise Services can connect to other URLs when the exposed service parameters are changed. For example, the set of Web services described in the following section could be combined into a single Enterprise service exposed through the services registry in a format similar to UDDI 3.0. Since Enterprise Services are implemented as Web services, the method for integrating InfoPath with SAP Enterprise Services will be similar to the method of integrating InfoPath with SAP Web services.

<sup>2</sup> More Information on SAP Enterprise Services can be found at <https://www.sdn.sap.com/irj/sdn/enterprisesoa>.

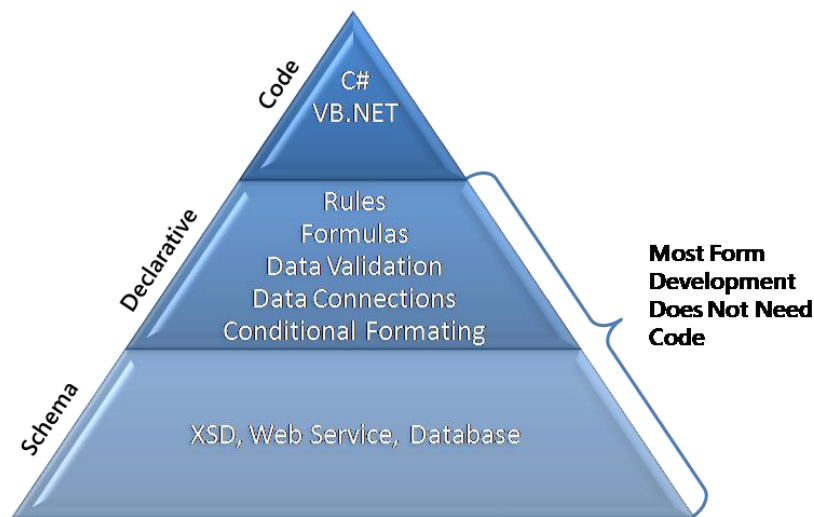
## Starting Simple – Using InfoPath 2007 as a Forms client

When developing lightweight forms solutions with SAP Web services, Office InfoPath 2007 provides a simple starting point.

Developers or experienced users with access to the respective SAP services layer can build customized forms for read-only or read/write access of those services. This can also be accomplished by using managed layer technologies like SAP Process Integration (PI) or Microsoft BizTalk Server.

With form solutions, most forms do not need code customization if using a managed middleware component like BizTalk or Process Integration (PI) where all definitions are stored in a central repository for access, testing, versioning, and reuse.

Figure 3 below illustrates the codeless approach for the majority of form solutions based on InfoPath methodologies.

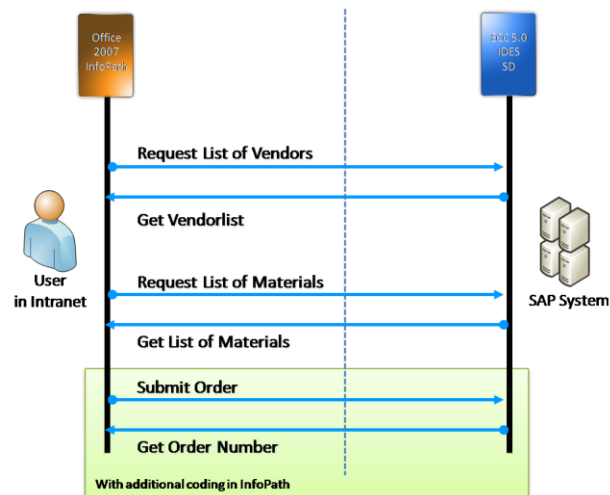


**Figure 3: Minimal Form Code Need**

Forms are typically stored locally, sent through e-mail, or shared on a file share or a Windows SharePoint Services site. A read-only solution provides a code-free, easy to configure way to integrate predefined forms with pre-populated SAP data for offline use. However, a read/write solution requires some additional InfoPath coding to send data from the different form fields to the SAP system.

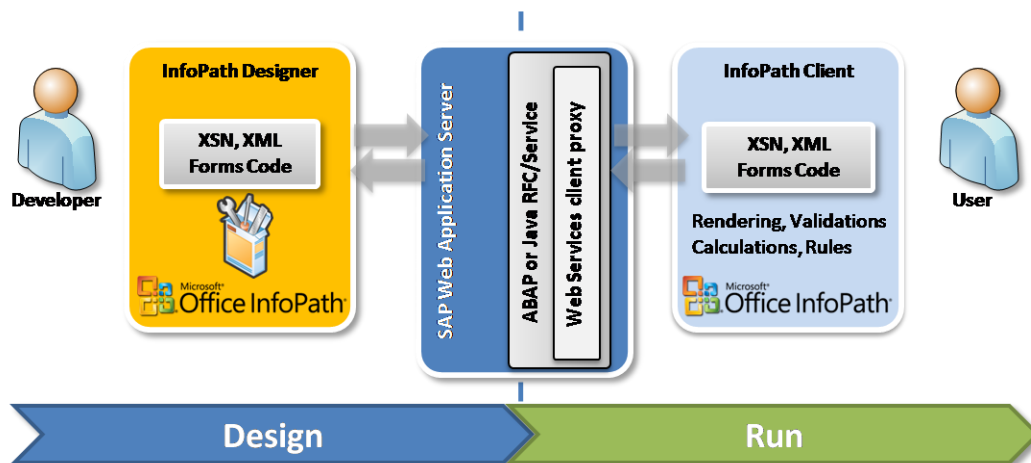
### Sample Process Flow

The following Figure 4 illustrates the most basic forms solution. The InfoPath client communicates directly with the SAP service or optionally with encapsulated services provided by a service bus like BizTalk Server or NetWeaver PI which is highlighted in green below.



**Figure 4. A Simple Forms Solution**

The only required components for this solution are Office InfoPath and the respective SAP services described as Web services. These are NetWeaver Application Server Web services that are exposed through BizTalk or SAP PI. The development process flow is illustrated in Figure 5 below.



**Figure 5. Development Process Flow**

The following table lists the pros and cons for using InfoPath as a forms client solution.

Pros	Cons
Easy to build	Limited access to data sources
No server footprint	No custom SAP Web services can be used without custom coding
InfoPath is often licensed as part of Office 2007 Enterprise CAL	No central solution versioning or endpoint management
Little to medium skill set needed for development	No transactional support
	Limited write back capabilities

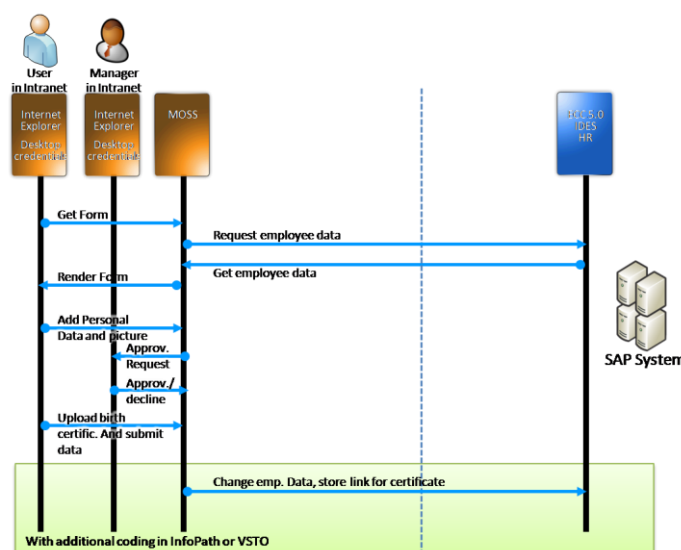
***Table 2. Considerations for using InfoPath as a Forms Client Solution***

## Extending the Forms Client Solution – Adding SharePoint Technologies for Scalability

SharePoint Server adds a significant number of important enhancements to an InfoPath forms client solution to create a scalable enterprise forms solution. It enables better scalability, common access, and distributed workflow as well as functions like archiving, signing, central search, DRM, and collaboration. SharePoint Server also supports online and offline scenarios as well as single sign-on (SSO) with the help of SharePoint Enterprise SSO or the SAP Kerberos login module (SPNego). The SPNego module runs on a NetWeaver Application Server in a Java instance.

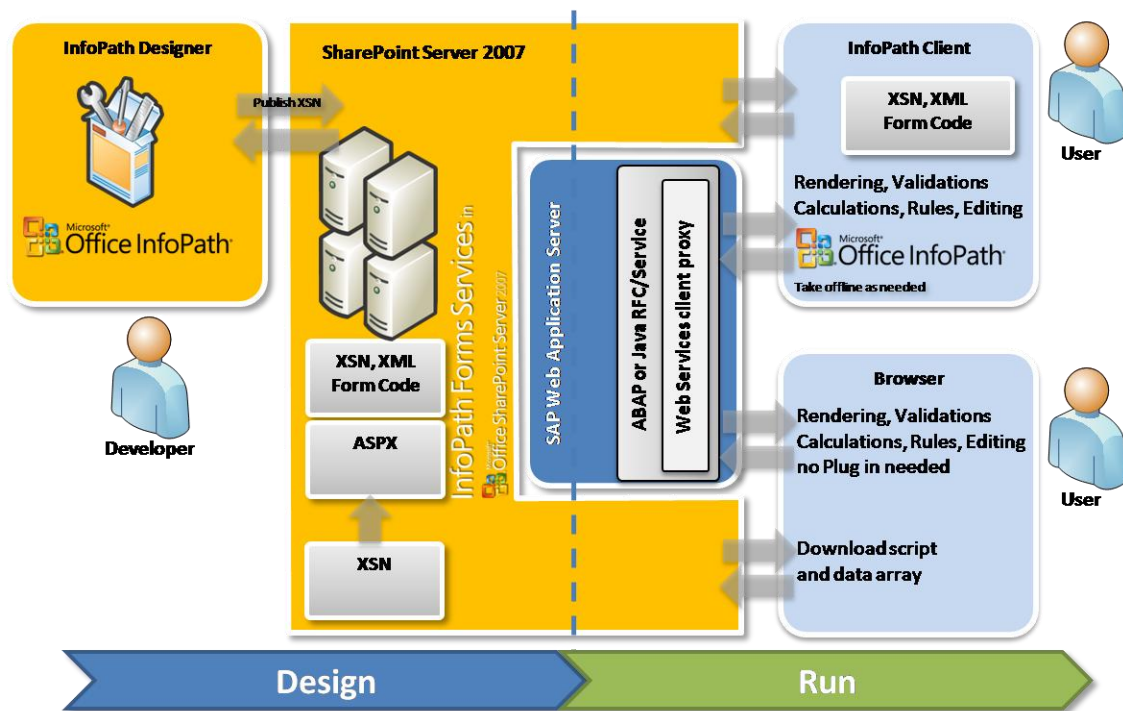
### Sample Process Flow

This new extended scenario includes additional functionality such as an approval workflow, digital signatures, and a central Electronic Records Management (ERM) environment using SharePoint Server. As illustrated in Figure 6 below, this structure includes Microsoft side services for workflow control, SAP services access consistency, and provides additional SAP system functionality beyond the business logic.



**Figure 6. A Forms Solution using SharePoint Server**

In addition to the last solution, with SharePoint Technologies we now have a forms repository and access to the full range of services as shown in the following Figure 7. The solution still provides access to all SAP system functions using the SAP system UI. However, it is now much more convenient to build additional services using forms with browser-based access.



**Figure 7. Development Process Flow with SharePoint Functionality**

During our tests we found that it was helpful to work with local XML document schemas for each form. This provided a local form representation rather than a dynamic creation when simply bound to Web services. The SAP Web services data connection is stored on the server as a Universal Data Connection (UDC) file. Therefore, for online rendering, a client side installation of InfoPath is not needed. However, the forms can be rendered and used online and offline using InfoPath.

The following table lists the pros and cons for using InfoPath as a forms client solution and extending it with SharePoint Server.

Pros	Cons
Central repository for forms, data sources, user management documents	Additional complexity with the service layer for SharePoint
Workflow capabilities	Central user rights have to be planned and deployed in advance using Active Directory® (AD)
Document libraries added for attachments, pictures, PDFs, and multimedia content	Central management requires definition of roles, concepts, and user rights
Rights management	
Digital signatures	
Tracing, tracking	
Versioning	
Browser-based access possible – no footprint on the client for most of the available form options	

**Table 3. Using InfoPath and SharePoint Server as a Forms Client Solution**

## Creating a Complete SOA Compliant Managed Forms Solution

Our preferred structure is a managed forms solution with transactional support and Service-Oriented Architecture (SOA) compliance that connects to SAP Web services. We prefer this because it is manageable, transactional, searchable, SOA compliant, and includes tracing and tracking.

The managed forms solution provides full manageability of the whole process, including component, role, format, and transport port management. Process tracing and tracking are fully integrated with business intelligence. For example, determining where a user is in the approval process or if an order was approved can be answered easily.

This solution can be used in large organizations, but works in medium organizations as well. This is because the licensing costs are scalable. Licensing can start with standard versions and grow into full CALs in an enterprise agreement when needed.

Changes in disparate systems can be managed from a single location by editing the schema of a specific document or by changing the transport port for an approval. This single management location is critical in an SOA environment. For example, this would greatly simplify the process of changing the WCF function to a Web service. For a Microsoft solution, BizTalk Server would be used for this task.

### Sample Process Flow

As shown in Figure 8 below, the SAP layer remains the same as the solutions in the previous sections. All new services were added to the process using an Enterprise Services Bus (ESB). If the workflow process resides in BizTalk instead of Microsoft Office SharePoint Server, digital signing, BizTalk mapping, right adapter choice, employee search, and Business Activity Monitoring (BAM) are now included in the process.

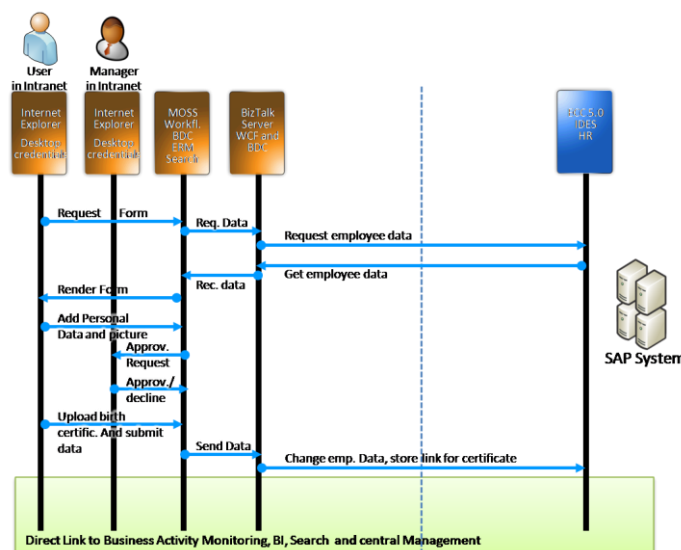
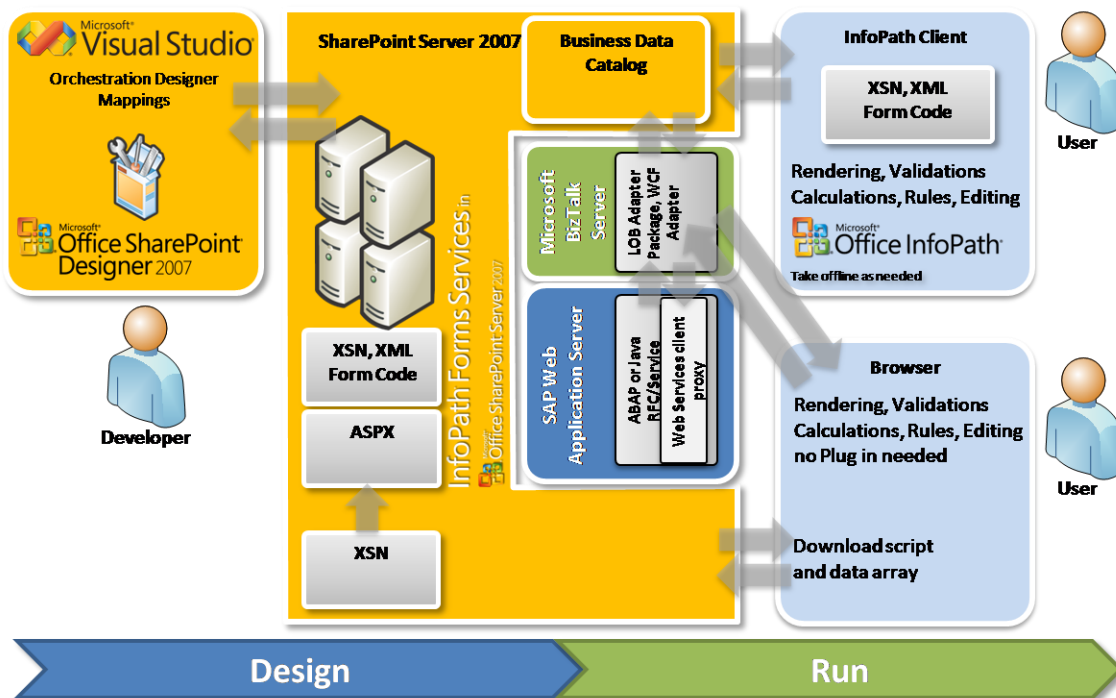


Figure 8. A Forms Solution using SharePoint Server and BizTalk



In Figure 9 below, the BizTalk Server is used as an Enterprise Service Bus to extend the capabilities of the SAP system. The value of this solution is the single point of management for all transactional services. In a forms solution this might also mean that some services can be buffered or decoupled from the SAP system to improve performance and security. The Business Data Catalog exposes all functions and the structure of the connected and pre-defined SAP Services. This provides the ability to access all SAP systems including legacy systems like SAP R/3 4.6 or SAP 4.7.



**Figure 9. Development Process Flow with SharePoint and BizTalk Functionality**

The following table lists the pros and cons for using a complete SOA compliant managed forms solution.

Pros <sup>3</sup>	Cons
Transactional write back possible	Complexity makes it harder to trace errors
SOA compliant solution with Enterprise Services Bus to manage reusable components	Different service layers require expertise, developer skills, and a SOA blueprint in order to add the value provided by an Enterprise Services Bus (ESB)
Easy to add the BI function for workflow, transactions, monitoring, and other key performance indicators if applicable to a forms solution	
Easy to change ports and document definitions for a large scale solution	
Access SAP systems like R/3 4.6 and 4.7 and expose BAPIs/RFC as Web services	

**Table 4: Using InfoPath, SharePoint Server, and BizTalk Server as a Forms Client Solution**

---

<sup>3</sup> All pros from previous table apply.

## An InfoPath Forms Solution

To illustrate the power and flexibility of using InfoPath-based forms solutions, this section will provide an example of a common HR process. Using an InfoPath-based form solution, the process can be built quickly, easily, and can be extended as needed. The following section will provide a walkthrough of the form solution implementation.

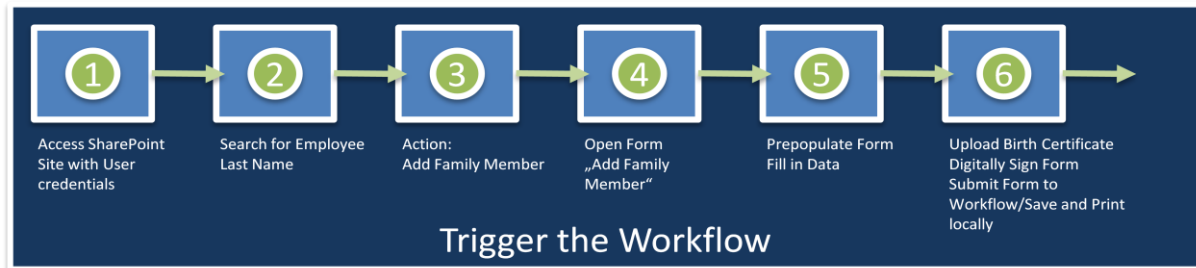
This forms solution includes all the capabilities required for a form-driven workflow including a custom approval workflow component and read/write SAP access. This particular forms solution uses the following features:

- InfoPath and SharePoint Forms Services
  - Web forms and offline forms
  - Contextual rules to control the behavior and layout
  - Digital signing and printing
- SharePoint Form Libraries
  - Persistence of form instances
  - Versioning
  - Ad-hoc queries and filtering
- SharePoint Designer Workflow
  - Declarative design and workflow deployment
  - Extensibility through custom SharePoint Workflow Activities
- BizTalk Server
  - Transactional RFC access
  - Composite Web services
- SAP Web services
  - Read access

The following BAPIs are used in order to retrieve and send data from and to our SAP ECC system:

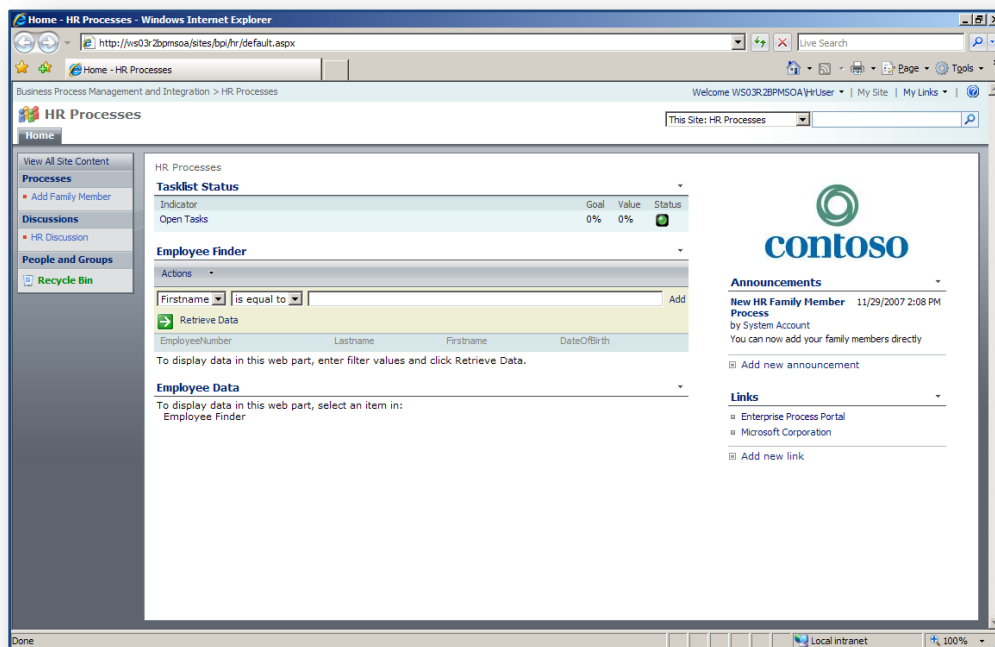
- BAPI\_PERSDATA\_GETDETAILEDLIST: Read personal employee data
- BAPI\_FAMILY\_GETDETAILEDLIST: Read data of family members
- BAPI\_FAMILY\_CREATE: Add family members
- BAPI\_EMPLOYEE\_ENQUEUE: Lock record
- BAPI\_EMPLOYEE\_DEQUEUE: Unlock record

The sample workflow extends the existing SAP process to add new family members by adding a three-step approval workflow and an electronic form with printing and signing capabilities. See the workflow overview in Figure 10 below. The approval workflow sequence is initiated by a new InfoPath form instance submitted by the HR User. Subsequently, the workflow sequence assigns the form to the HR Manager to approve the new family member request. Once approved, the workflow assigns the form to the HR Administrator who will submit the form as a transaction to the SAP system.



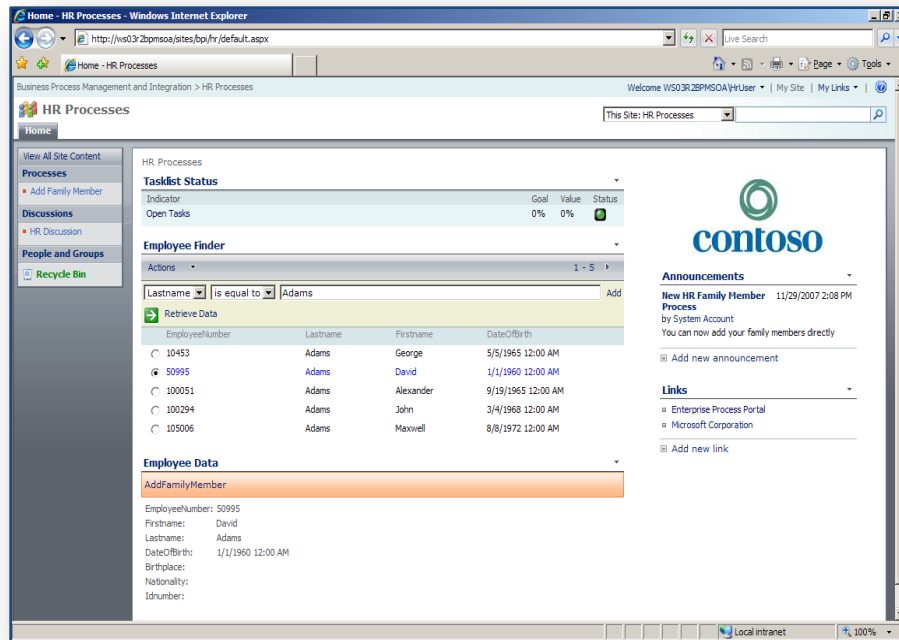
**Figure 10: Workflow Overview**

The HR User can either submit a new AddFamilyMember request by creating a new InfoPath form or by using the Business Data Catalog (BDC) to identify the appropriate employee. With the Employee Finder Web Part, the HR User can add a Firstname and Lastname search filter as shown in Figure 11.



**Figure 11: SharePoint Forms Services Contoso Forms Example**

The BDC retrieves a list of employees based on this filter as shown in Figure 12.



**Figure 12: Contoso HR Web Site Employee Search Results**

The user selects the employee in the Employee Finder Result list to retrieve detailed employee information in the Employee Data Web Part. The Employee Data Web Part also contains the AddFamilyMember BDC action that launches the InfoPath form for the selected employee as shown in the following Figure 13. This BDC action displays the InfoPath form URL that is deployed to InfoPath Forms Services in SharePoint and the attached parameter for the employee number.

**Employee Self Service**  
Add Family Member

contoso

Employee Number: 50995  
Firstname: David  
Lastname: Adams  
Date of Birth: 1960-01-01  
Workflow Status: NEW

**Children**

Firstname	Name	Date of Birth	Place of Birth
Paris	Adams	2007-12-05	Munich, Germany
Ben	Adams	2007-12-11	Munich, Germany

**Add Child**

Firstname:   
Lastname: Adams  
Date of Birth:   
Nationality: German  
Gender: ☒ Male ☐ Female  
Place of Birth:   
Birth Certificate:

[Click here to sign this section](#)

**Figure 13: SAP HR Add Family Member Screen**

If the HR User has launched the InfoPath form using the BDC action, the InfoPath form extracts the employee number parameter and populates the form with the employee and the employee's children's data. Alternatively, the HR User can also create a new InfoPath form by directly entering the employee number in a new InfoPath form.

To enter a new family member, the HR User must enter the name, date of birth, nationality, gender, and place of birth. Optionally, he or she can attach the birth certificate as a binary image file as shown in the following Figure 14.

**Employee Self Service**  
Add Family Member

Employee Number: 50995  
Firstname: David  
Lastname: Adams  
Date of Birth: 1960-01-01  
Workflow Status: NEW

**Children**

Firstname	Name	Date of Birth	Place of Birth
Paris	Adams	2007-12-05	Munich, Germany
Ben	Adams	2007-12-11	Munich, Germany

**Add Child**

Firstname: Rick  
Lastname: Adams  
Date of Birth: 12/11/2007  
Nationality: German  
Gender: ☒ Male ☐ Female  
Place of Birth: Munich, Germany  
Birth Certificate: Birth\_Certificate.jpg (113.36 KB)

[Click here to sign this section](#)

Close Save

**Figure 14: Birth Certificate Form Attachment for Upload to the SAP HR System**

Applying digital signatures is another useful InfoPath and InfoPath Forms Services feature. With it, the HR User can sign the form with a digital certificate as shown in Figure 15 below. This will make the form read-only and prevent further form modification throughout the rest of the approval workflow.

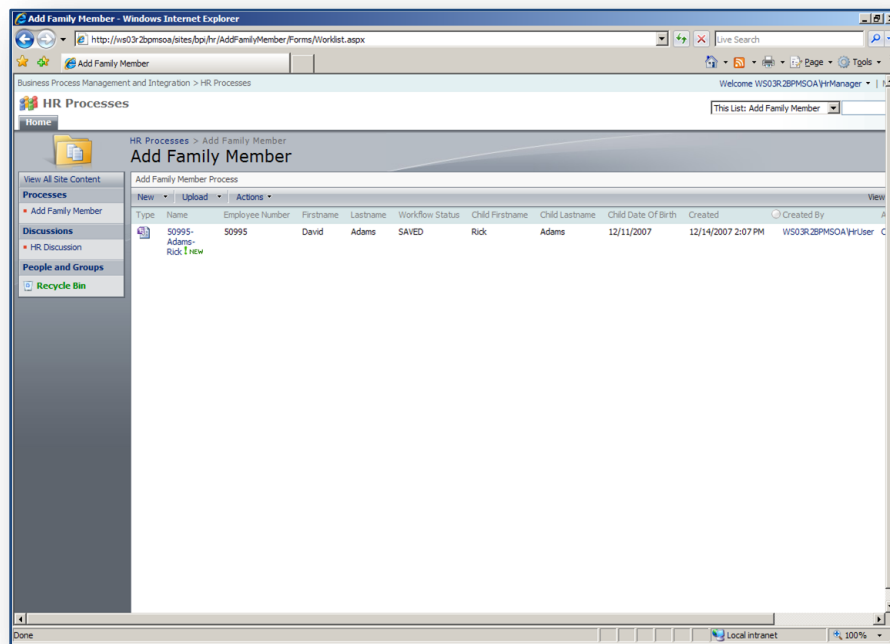
**Figure 2: Form Information Verification**

The default print view is generated automatically for every InfoPath view and can be customized if required. The HR User can print the form from the default print view as shown in Figure 16 below.

**Figure 3: Default Print View**



After the HR User has saved the form, it will appear in the HR Manager's AddFamilyMember Form library as shown in Figure 17 below. The form library is generated based on the InfoPath form template and shows the updated information such as the name of the employee and the child. The workflow process has exclusively assigned the form to the HR Manager so at this stage the user can no longer access the AddFamilyMember request. The underlying SharePoint workflow sets the item level permissions of the form instance to control the access rights.



**Figure 4: Add Family Member SharePoint Form Library**

In this example, the HR Manager only has the option to approve the request by clicking the **Approve** button or to close the form by clicking the **Close** button as shown in the following Figure 18.

This functionality could easily be extended to include a DECLINE/REJECT action. The InfoPath form contains conditional formatting rules based on the workflow status to control the behavior and the layout of the form. For example, the HR Manager cannot update the form data since it is digitally signed. Therefore, the Save button is hidden according to the workflow SAVED status.

**Figure 5: Request Approval Screen**

After the HR Manager has approved the request, the workflow assigns the form to the HR Administrator and it will appear in the HR Administrator's AddFamilyMember Form library as shown in the following Figure 19. Only the HR Administrator can access the form at his stage.

Type	Name	Employee Number	Firstname	Lastname	Workflow Status	Child Firstname	Child Lastname	Child Date Of Birth	Created	Created By
Form	50995-Adams-Rick	50995	David	Adams	APPROVED	Rick	Adams	12/11/2007	12/14/2007 2:07 PM	W503R2BPMSCA\hrAdministrator

**Figure 6: Approved Form Instances**

Based on the Approved workflow status, the HR Administrator can submit the AddFamilyMember request to the SAP system as shown in the following Figure 20. The SAP Submit button calls a BizTalk Web service that generates the required sequence of RFC calls to add the family member entry to the employee record.

50995-Adams-Rick.xml [Signed] Windows Internet Explorer

http://ws03r2bmsoa/sites/bp/hr/\_layouts/FormServer.aspx?xmlLocation=/sites/bp/hr/AddFamilyMember/50995-Adams-Rick.xml&Source=http%3A%2F%2F... Live Search

Print View Powered by: InfoPath Forms Services

## Employee Self Service

### Add Family Member

**Employee Number** 50995

**Firstname** David

**Lastname** Adams

**Date of Birth** 1960-01-01

**Workflow Status** APPROVED

**Add Child**

**Firstname** Rick

**Lastname** Adams

**Date of Birth** 12/11/2007

**Nationality** German

**Gender** ☒ Male ☐ Female

**Place Of Birth** Munich, Germany

**Birth Certificate** Birth\_Certificate.jpg 113.36 KB

**Valid signature - Signed by Administrator** Friday, December 14, 2007 2:03:55 PM [Show Details](#) [Remove](#)

**Figure 20: Form Submittal to SAP**

The Web service then displays a response message from the SAP system as shown in the very bottom of the following Figure 21. Based on this response, the InfoPath form disables the SAP Submit button, sets the workflow status to SAP, and saves the form instance.

**Employee Self Service**  
Add Family Member

Employee Number: 50995  
Firstname: David  
Lastname: Adams  
Date of Birth: 1960-01-01  
Workflow Status: SAP

**Add Child**

Firstname: Rick  
Lastname: Adams  
Date of Birth: 12/11/2007  
Nationality: German  
Gender: Male  
Place of Birth: Munich, Germany  
Birth Certificate: Birth\_Certificate.jpg (113.36 KB)

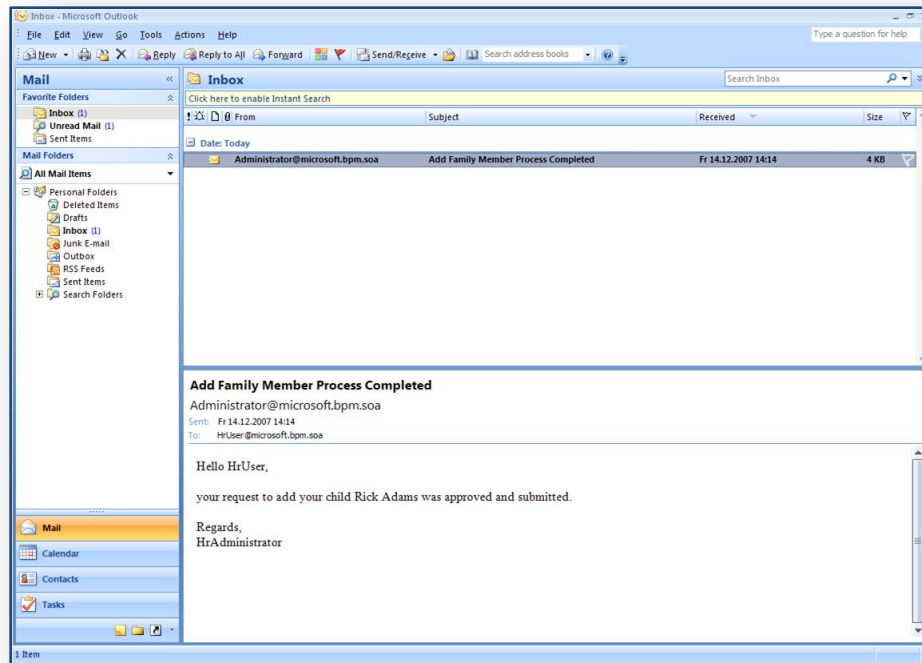
Valid signature - Signed by Administrator  
Friday, December 14, 2007 2:03:55 PM  
Show Details Remove

Close

The new child entry was created successfully.

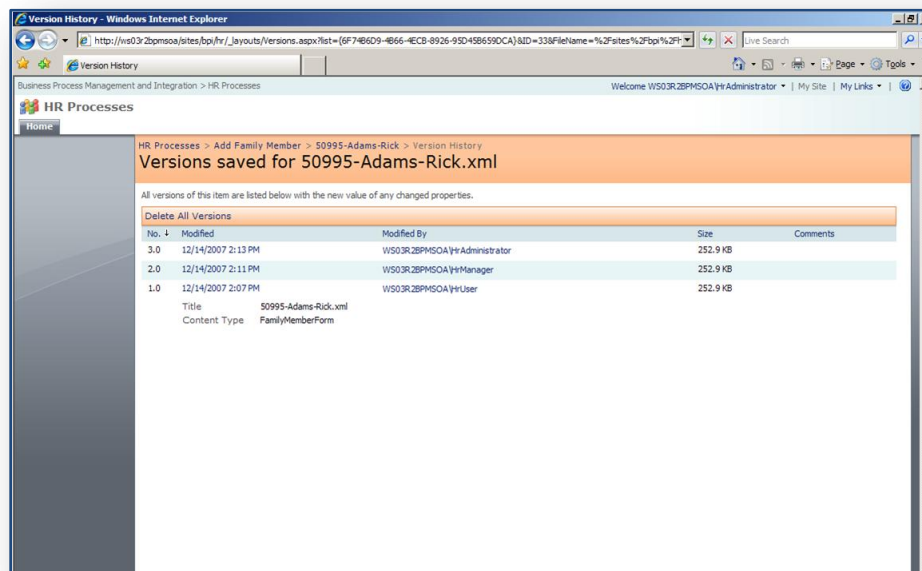
**Figure 21: Completed Add Family Member Request**

The SharePoint workflow sends a notification e-mail to the HR User once the form instance was saved with the SAP workflow status as shown in the following Figure 22.



**Figure 22: Notification E-mail**

The SharePoint library creates a new version of the form instance each time an update is made as shown in the following Figure 23. Using the SharePoint versioning feature, workflow users can track all status changes for each request.



**Figure 23: Form Instance Versioning**

## InfoPath Forms Solution Implementation

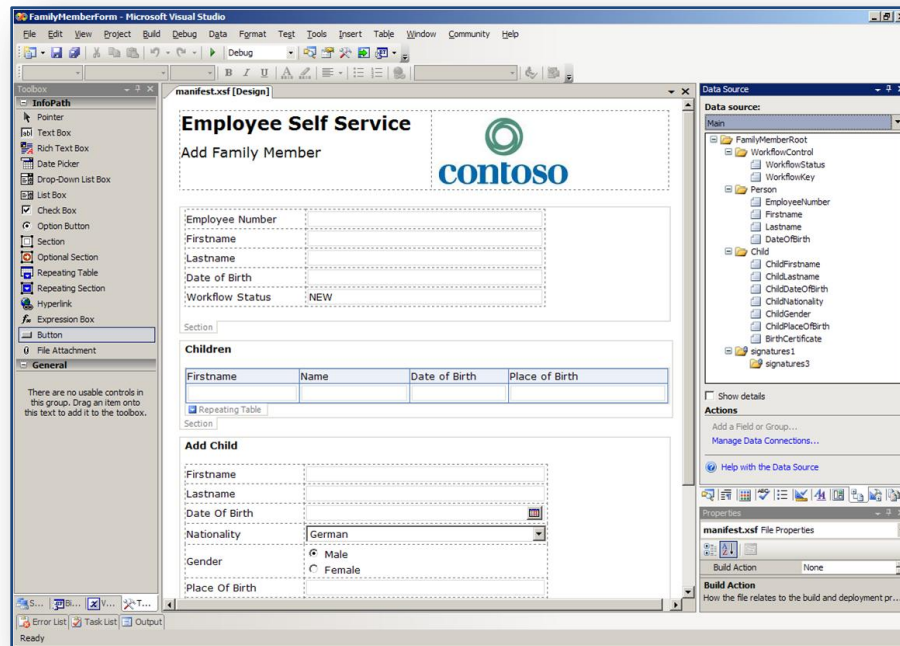
The previous section described a sample InfoPath-based forms solution. This section reviews the implementation of that sample solution.

The sample solution described in this section uses the following elements:

- Visual Studio Tools for Office (VSTO) InfoPath Form Template Solution: FamilyMemberForm.sln
- BizTalk Web Services Solution: BizTalk\_SAP\_Employee.sln
- SharePoint Designer Workflow: AddFamilyMemberWorkflow.xml, AddFamilyMemberWorkflow.xml.rules, AddFamilyMemberWorkflow.xml.wfconfig.xml
- SharePoint Designer Custom Workflow Activities: <http://www.codeplex.com/SPDActivities>.
- SharePoint Business Data Catalog Definition: SAPIDESHR.xml
- Data Connection Library files: BAPI\_FAMILY\_GETDETAILEDLIST.udcx, BAPI\_PERSDATA\_GETDETAILEDLIST.udcx

The InfoPath form template was developed with Visual Studio Tools for Office (VSTO). This form template could also have been developed in InfoPath 2007 with Visual Studio Tools for Applications (VSTA). Although VSTA is a lightweight development tool compared to VSTO, the form template solution in this example requires minimal source code so there is minimal functional restriction when using VSTA.

The form template shown in Figure 24 is based on a dedicated data source that defines the data structure of the form. It includes the employee data that is retrieved from SAP, the data required to add a new child, a node to store the digitally signed form, and a workflow control node. The workflow control node includes the workflow status that controls the behavior of the form and the SharePoint workflow. The workflow key stores a unique ID for each form instance that is used as the physical file name in the underlying SharePoint form library.



**Figure 24: InfoPath Form Template in Visual Studio Tools for Office**

All the core elements of the AddFamilyMember form are designed and configured using the declarative features of InfoPath such as conditional formatting, rules, and data connections. The loading event contains the only code in the form as shown in the following Figure 25. This code increases the CreateFamilyMember Web Service timeout period and retrieves the input parameter if the form is opened by a BDC action.

```
public void FormEvents_Loading(object sender, LoadingEventArgs e)
{
    WebServiceConnection wsc = (WebServiceConnection) this.DataConnections["CreateFamilyMember"];
    wsc.Timeout = 104;

    try
    {
        string employeeid = string.Empty;

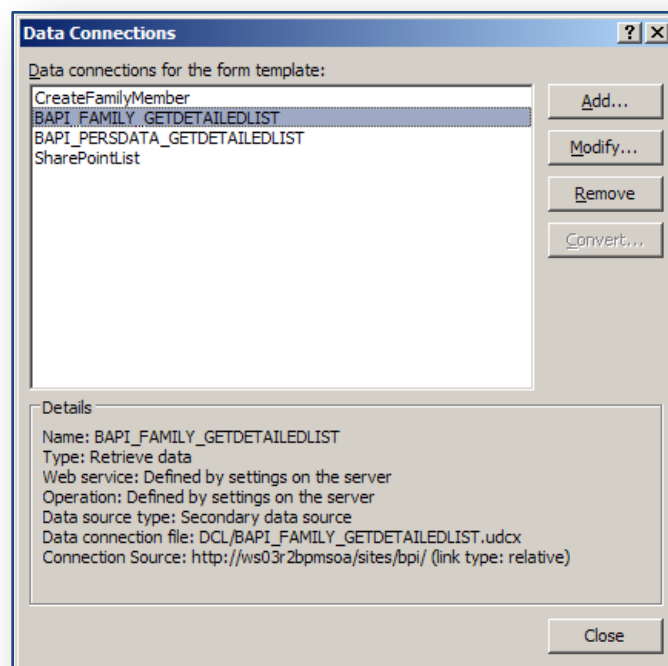
        e.InputParameters.TryGetValue("EmployeeNumber", out employeeid);

        if (employeeid != null && employeeid.Length > 0)
            this.CreateNavigator().SelectSingleNode("/my:FamilyMemberRoot/my:Person/my:EmployeeNumber",
                NamespaceManager).SetValue(employeeid);
    }
    catch (Exception) { }
}
```

**Figure 25. Loading Event Code**

As shown in Figure 26, the form template uses four data connections to connect to Web services and to SharePoint:

- CreateFamilyMember: BizTalk Web Service enables the ability to add the family member entry. Provides write access.
- BAPI\_FAMILY\_GETDETAILEDLIST: A SAP Web service that is used to retrieve existing family member entries from the SAP system. Provides read access.
- BAPI\_PERSDATA\_GETDETAILEDLIST: A SAP Web service that is used to retrieve employee data from the SAP system. Provides read access.
- SharePointList: A connection used to submit form instances to the SharePoint Library such as with the Add Family Member request. Provides write access.



**Figure 26: InfoPath Data Connections**

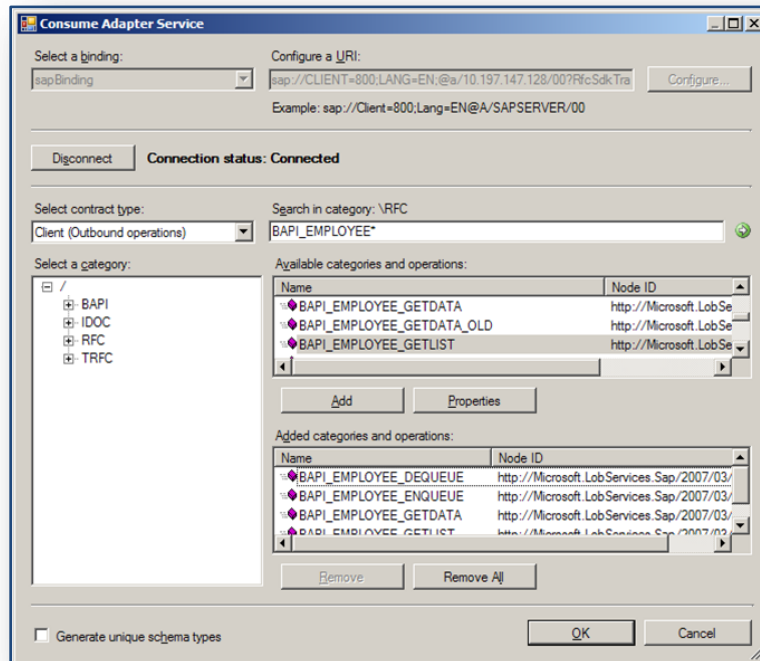
BizTalk Orchestration Designer is a Microsoft Visio 2000–based design tool that enables the user to create business process drawings that can be compiled and run as XLANG schedules. XLANG is an XML-based language. An XLANG schedule describes the business process and the binding of that process to application services.

The BizTalk solution BizTalk\_SAP\_Employee.sln contains three orchestrations that are published as Web services to retrieve employee data from the SAP system and to add children to existing employee records. The BizTalk solution uses the WCF-based SAP adapter that is part of the BizTalk Adapter Pack. For more information, see:

<http://www.microsoft.com/biztalk/en/us/adapter-pack.aspx>.

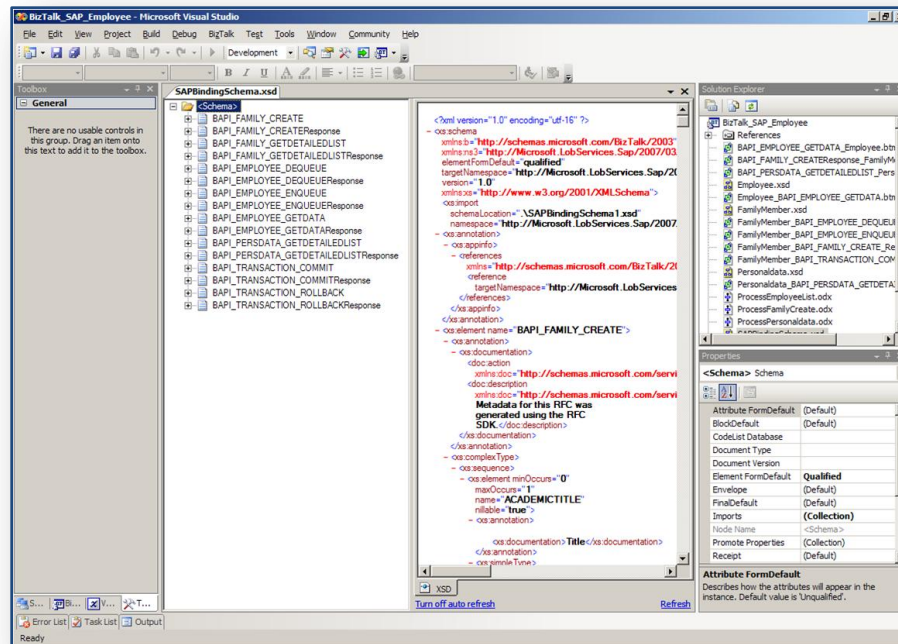


As shown in the following Figure 27, the adapter includes the Consume Adapter Service wizard used to connect to an existing SAP system and to generate the schema definitions for selected BAPIs, RFCs, or IDOCs using the underlying SAP Business Object Repository metadata.



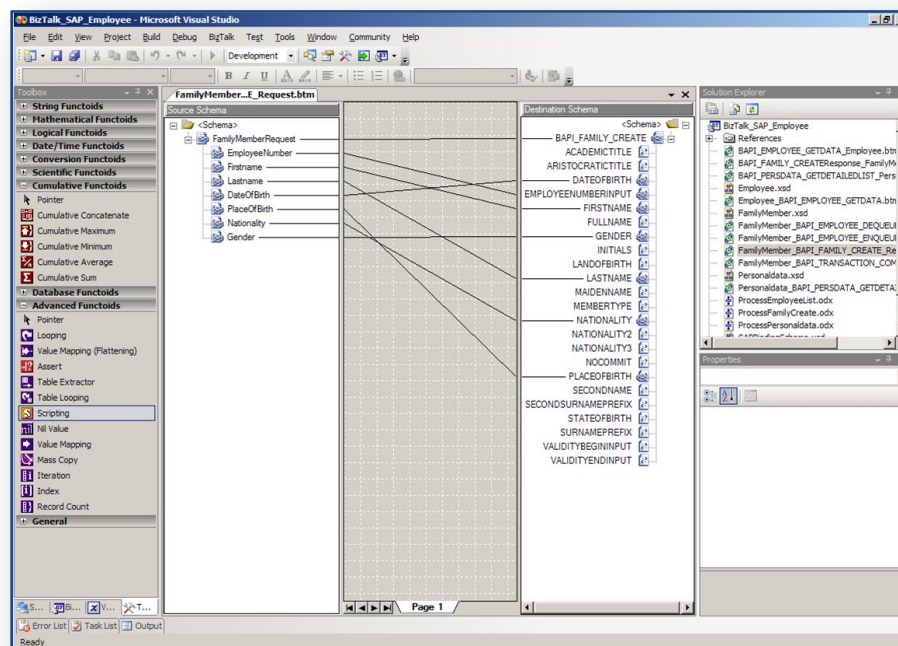
**Figure 27: Consume Adapter Service in a BizTalk Solution**

The adapter generates one BizTalk schema for all selected RFCs as shown in the following Figure 28. To hide the complexity of the comprehensive RFC methods, the BizTalk projects contain separate schemas for the Web service contracts including Employee.xsd, FamilyMember.xsd, and Persondata.xsd.



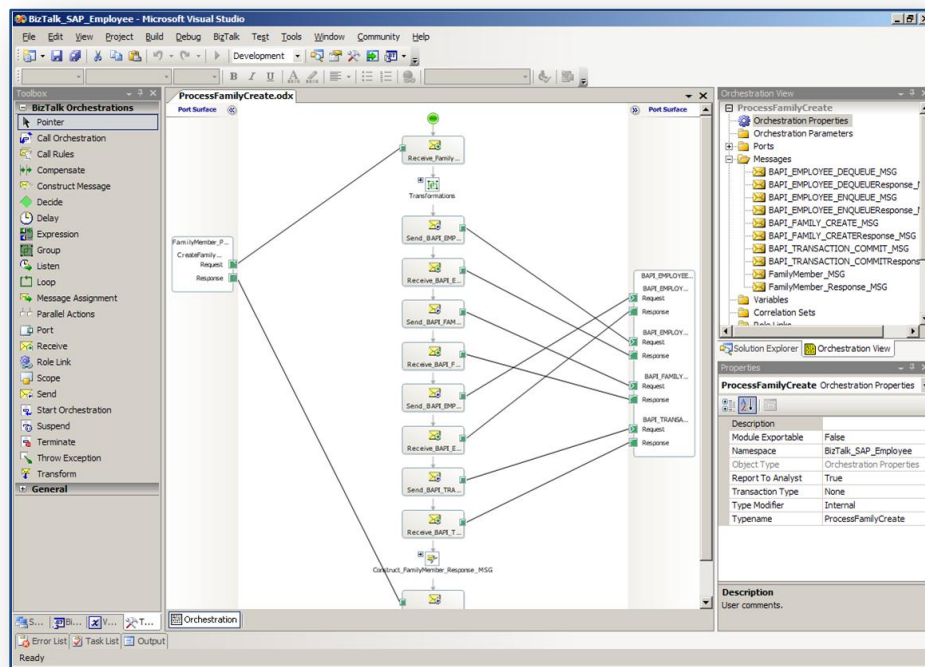
**Figure 28: Generated SAP Binding Schema**

These request and response schemas are mapped according to BAPI calls as shown in the following Figure 29.



**Figure 29: Web Service Request to the BAPI Call Mapping**

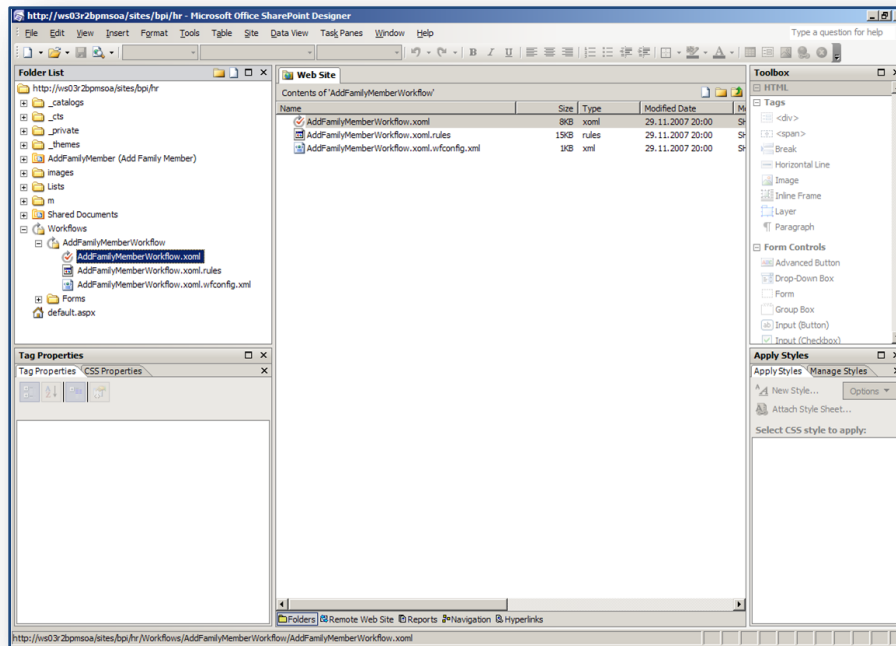
While the orchestrations *ProcessEmployeeList.odx* and *ProcessPersonaldata.odx* simply encapsulate existing SAP RFCs as Web services to retrieve data, the orchestration *ProcessFamilyCreate.odx*, as shown in the following Figure 30, makes several subsequent BAPI calls in a shared SAP connection. In order to add a new family member entry to an existing employee record in SAP, the employee record needs to be locked, updated, unlocked, and committed with the *BAPI\_TRANSACTION\_COMMIT* command.



**Figure 30: BizTalk Orchestration to Call Subsequent BAPIs**

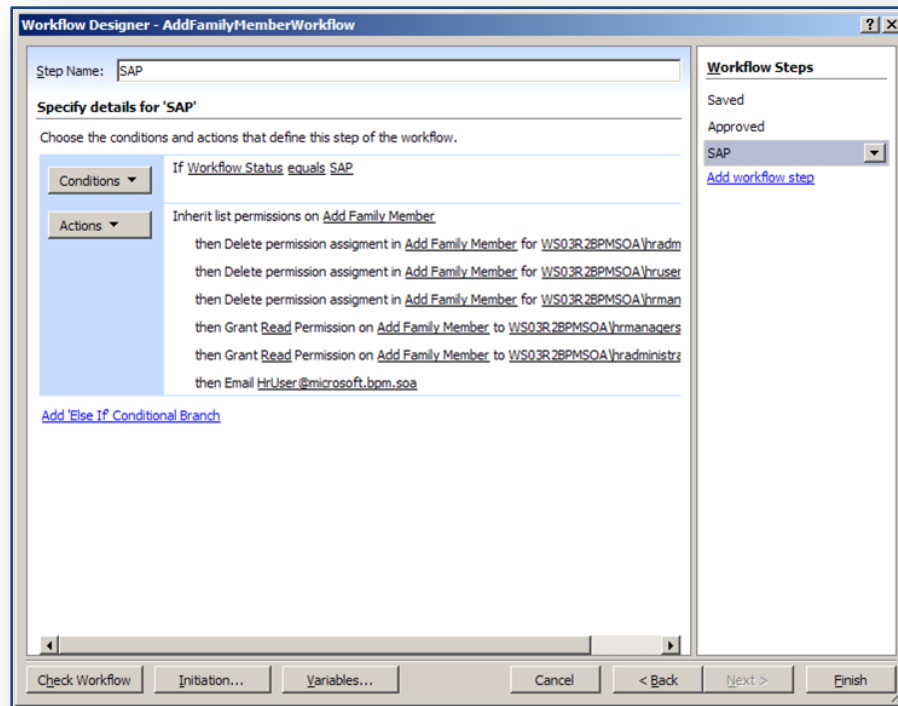
The three-step approval workflow is configured with SharePoint Designer. The SharePoint Designer workflow is attached to the SharePoint Form library that stores the InfoPath form instances as shown in the following Figure 31. The workflow starts when a form instance is created or updated. There are several workflow capabilities and options in SharePoint. For more information, see:

<http://msdn.microsoft.com/en-us/library/aa830816.aspx>.



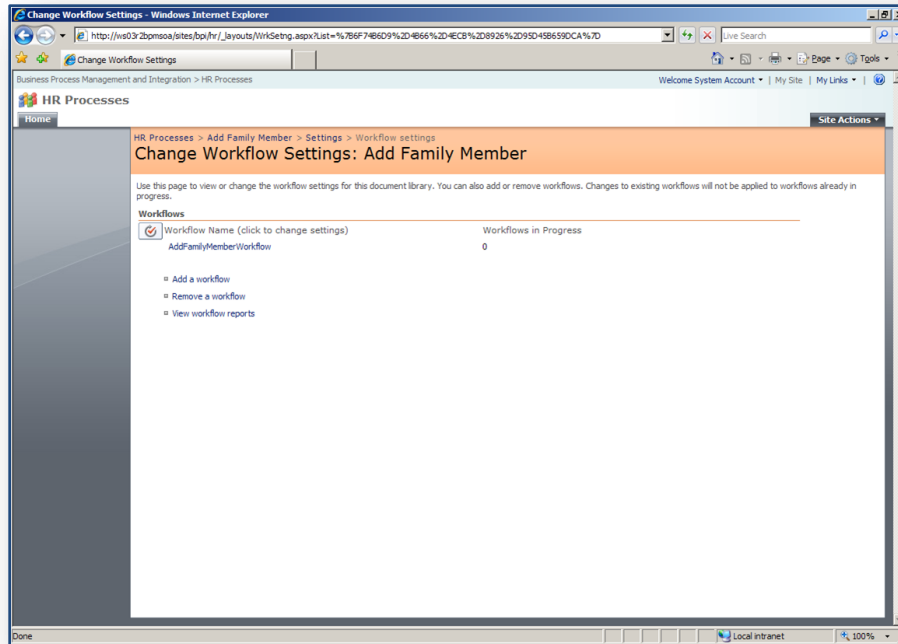
**Figure 31: SharePoint Designer Workflow Files**

In this scenario, the approval workflow uses a stateless workflow pattern that consists of steps for each workflow status including Saved, Approved, and SAP. Each workflow step is identified by a condition that evaluates the workflow status. In this way, a set of activities is applied to each InfoPath form instance based on the workflow status as shown in the following Figure 32. These activities include deleting and granting item level permissions and sending notification e-mails.



**Figure 32: SharePoint Designer Workflow Settings**

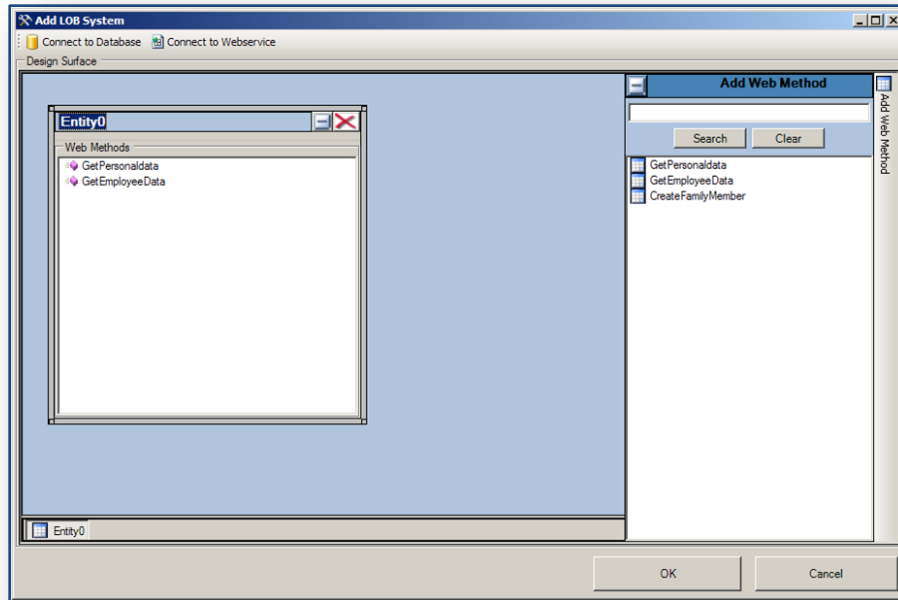
Once a SharePoint workflow is deployed and attached to a library, it can be configured and controlled in the workflow settings. For example, removing a workflow, terminating workflow instances, or inspecting the workflow history can now take place as shown in the following Figure 33.



**Figure 33: SharePoint Workflow Settings**

The Employee Finder Web Part is based on a SharePoint Business Data Catalog (BDC) definition. The BDC definition is a metadata structure that defines how the underlying Web services are represented within SharePoint.

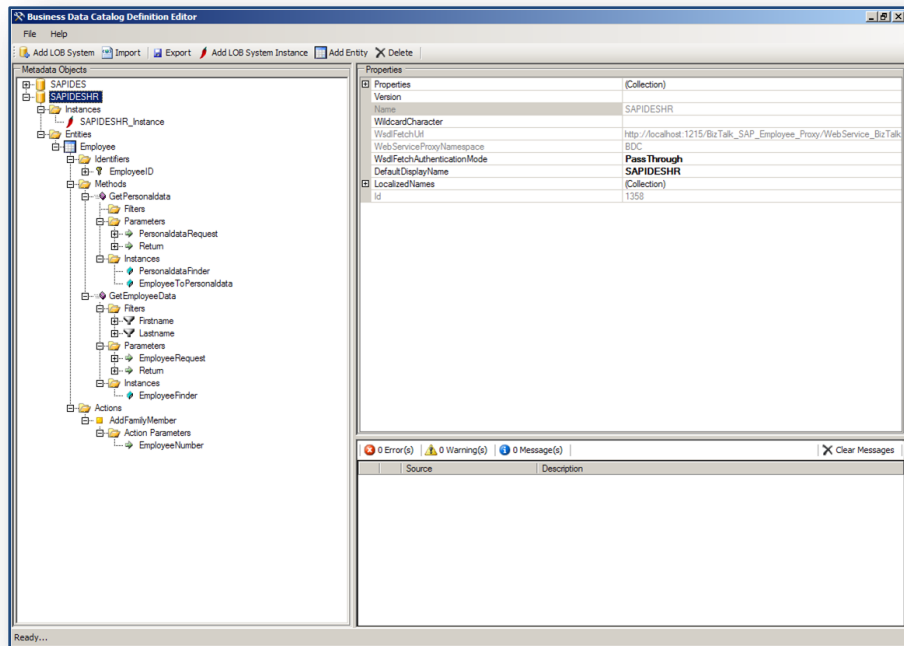
The Employee Finder leverages the Web services `GetPersonalData` and `GetEmployeeData` that are provided by BizTalk Server using the WCF based Line of Business Adapter for SAP as shown in the following Figure 34. Based on the WSDLs of these imported Web services, the BDC definition includes custom metadata for the employee filter, the association between the employee list and the employee details, and the action that opens the InfoPath form.



**Figure 34: Adding Web Methods to a BDC Entity**

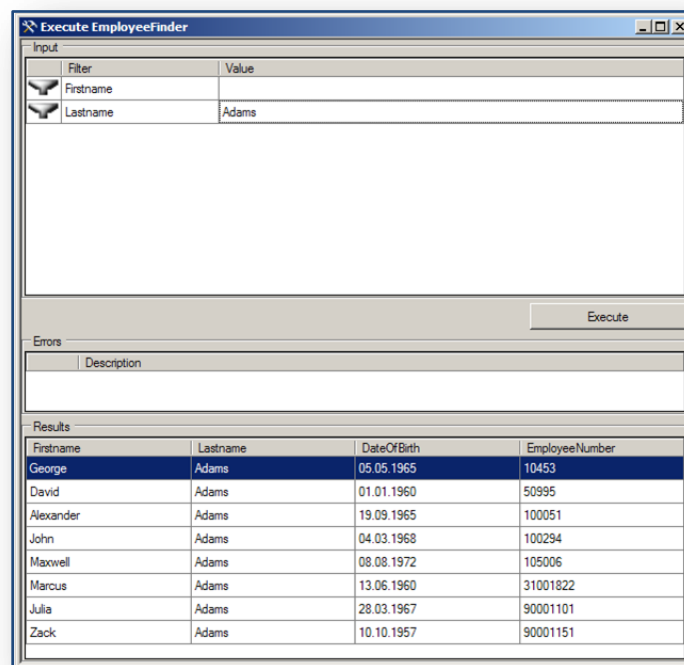
As shown in the following Figure 35, the BDC definition editor is part of the SharePoint Server 2007 Software Development Kit (SDK). This SDK can be found at:

<http://www.microsoft.com/downloads/details.aspx?FamilyId=6D94E307-67D9-41AC-B2D6-0074D6286FA9&displaylang=en>.



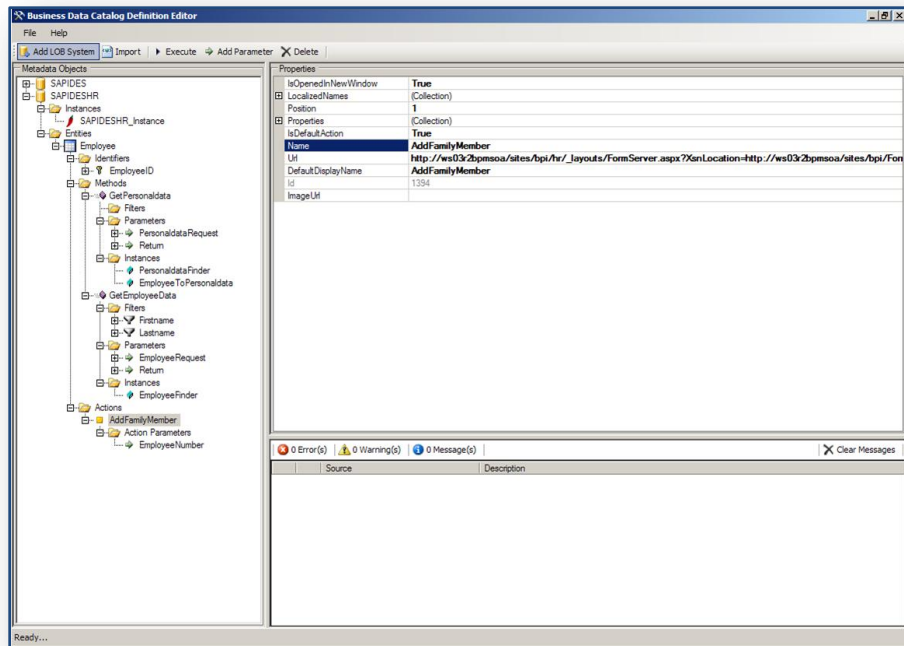
**Figure 35: Business Data Catalog Definition Editor**

In addition to the BDC definition creation capabilities, the tool provides several ways to test the definition before it is deployed to SharePoint as shown in Figures 36, 37, and 38.



**Figure 36: BDC Method Instances Test**





**Figure 37: BDC Action to Launch the InfoPath Form**

**Execute AddFamilyMember**

Select an Entry Instance

Name	Type	Value
EmployeeID	System.String	00001700

Errors

Description
-------------


Results

EmployeeNumber	Firstname	Lastname	DateOfBirth	Birthplace	Nationality	Idnumber
1700	Martin	Anstett	01.05.1960		DE	

Print View Powered by: InfoPath Forms Services

## Employee Self Service

### Add Family Member



Employee Number

Firstname

Lastname

Date of Birth

Workflow Status

#### Children

Firstname	Name	Date of Birth	Place of Birth
New	Anstett	2007-11-28	Munich
New	Anstett	2007-11-29	Munich, Germany

about:blank

**Figure 38: Test BDC Actions**

We have not described Single Sign-On setup in this paper since it depends on components in the infrastructure that might vary across installations.

The easiest InfoPath authentication scenario would work with the SAP system username and password authentication. An integrated approach could be the use of Kerberos or using the user's logon credentials and sending them to the SPNego Module of SAP NetWeaver<sup>4</sup>.

A good starting point for scenario two is the Enterprise Single Sign-on shipping with MOSS. In this scenario you would have to include your MOSS farm into an AD. BizTalk Server uses the same mechanisms like MOSS in terms of SSO so this solution looks very similar. We recently published an overview paper about the different approaches for SSO where we described the different SSO approaches like Enterprise SSO from Microsoft or typical scenarios for using the SPNego login module from SAP. This document can be found at:

<http://download.microsoft.com/download/c/6/c/c6c42b9f-66f4-47b3-99be-8e5afa1ddc9a/SSO%20with%20MS%20and%20SAP.pdf>.

---

<sup>4</sup> User must run the Java Stack in order to be able to run the SPNego login module.

## Summary

Microsoft Office InfoPath 2007 combines the familiar Microsoft Office system environment with a rich set of tools that enable you to build forms solutions that validate data, integrate into business systems, and guide users through an easy form completion experience. By using InfoPath Forms Services with SharePoint Server, you can extend your SAP business processes beyond the corporate firewall to provide electronic forms for use in the Microsoft Office system, Web browsers, and mobile devices. In addition, a modern Web services environment needs a manageable services layer that can be provided by adding BizTalk Server to the solution.

The tight integration between Office InfoPath 2007 and the other programs included in the 2007 release of the Microsoft Office system provide new ways to capture and consume information. InfoPath is built on industry standards like XML Schema, XSLT, and others. It provides an easy way to standardize the data collection processes across your entire organization.

## Resources

Official Microsoft | SAP Customer Information Center:

<http://www.microsoft.com/isv/sap/default.aspx>

Microsoft and SAP Interoperability White Papers:

Microsoft

Customer Information Center:

<http://www.microsoft.com/isv/sap/technology/interop/default.aspx>

TechNet

<http://technet.microsoft.com/en-us/interopmigration/bb544954.aspx>

SAP

SAP Developer Network

<https://www.sdn.sap.com/irj/sdn/dotnet>