

SDN Community Contribution

(This is not an official SAP document.)

Disclaimer & Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade. SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.

Applies To

SAP XI 3.0

Expected audience is consultants who mainly work for SAP XI Implementation Project where they are involved in developing interfaces for sending a mail with attachment.

Summary

The objective of this document is to provide solution for sending a mail with payload as an attachment in CSV format. It also demonstrates how to implement an asynchronous execution of WebService from Legacy Application via XI.

By: Krishna Moorthy

Title: Senior Software Engineer, Wipro Technologies Bangalore India.

Reviewed By: Chintamani Chitnis, Systems Manager, NetWeaver Competence Group (NWCG), Wipro Technologies, Pune, India.

Company: Wipro Technologies

Applies To.....	1
Summary	2
1.Business Scenario	4
Objective	4
Overview	4
Description	4
Pre-requisites	4
2. Implementation Steps.....	5
Integration Repository-Design	5
Integration Directory-Configuration	9
Disclaimer & Liability Notice	10
Author Bio	10

1. Business Scenario

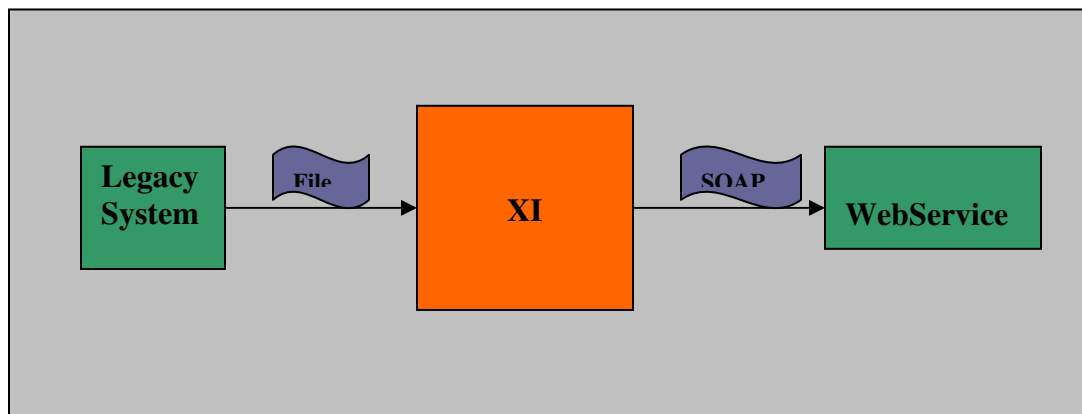
An Outbound Interface, where a mail is sent with payload as an attachment in CSV format through XI.

Objective

The objective of this document is to provide solution for sending a mail with payload as an attachment in CSV format.

Overview

This document is to demonstrate how to implement an asynchronous execution of WebService from Legacy Application via XI. After reading this document you will be in position to implement this scenario using XI with File adapter as a Sender and SOAP adapter as a Receiver. The WebService is responsible for converting XML payload into CSV file and sending a mail with CSV file as an attachment.



Description

File is picked from the legacy system using File Adapter. The file content is sent to XI. Receiver and Interfaces are determined. Mapping between source file structure and target WebService structure is executed. WebService is executed by SOAP Receiver Adapter.

Pre-requisites

- Input file should have all information required by Java mail API. This can be header information. Body or detail part of the file should have content to be sent as attachment.

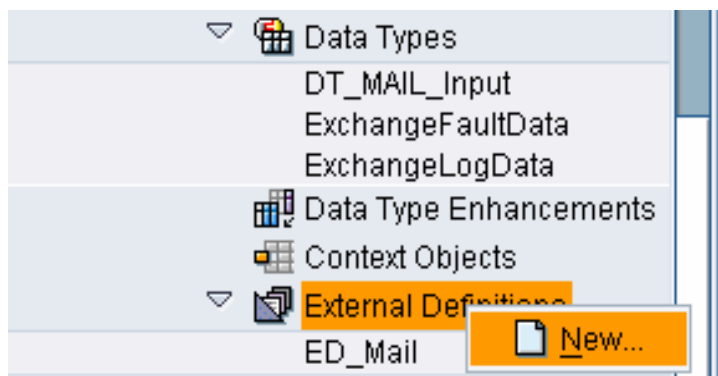
- Java application should be ready that parses XML payload and converts into CSV file. The application should also include code for sending a Mail with attachment using Java Mail API.
- This application should be exposed as a WebService and deployed in XI Server.
- The WSDL file should be provided by WebService Provider that is imported and stored in a local drive.

2. Implementation Steps

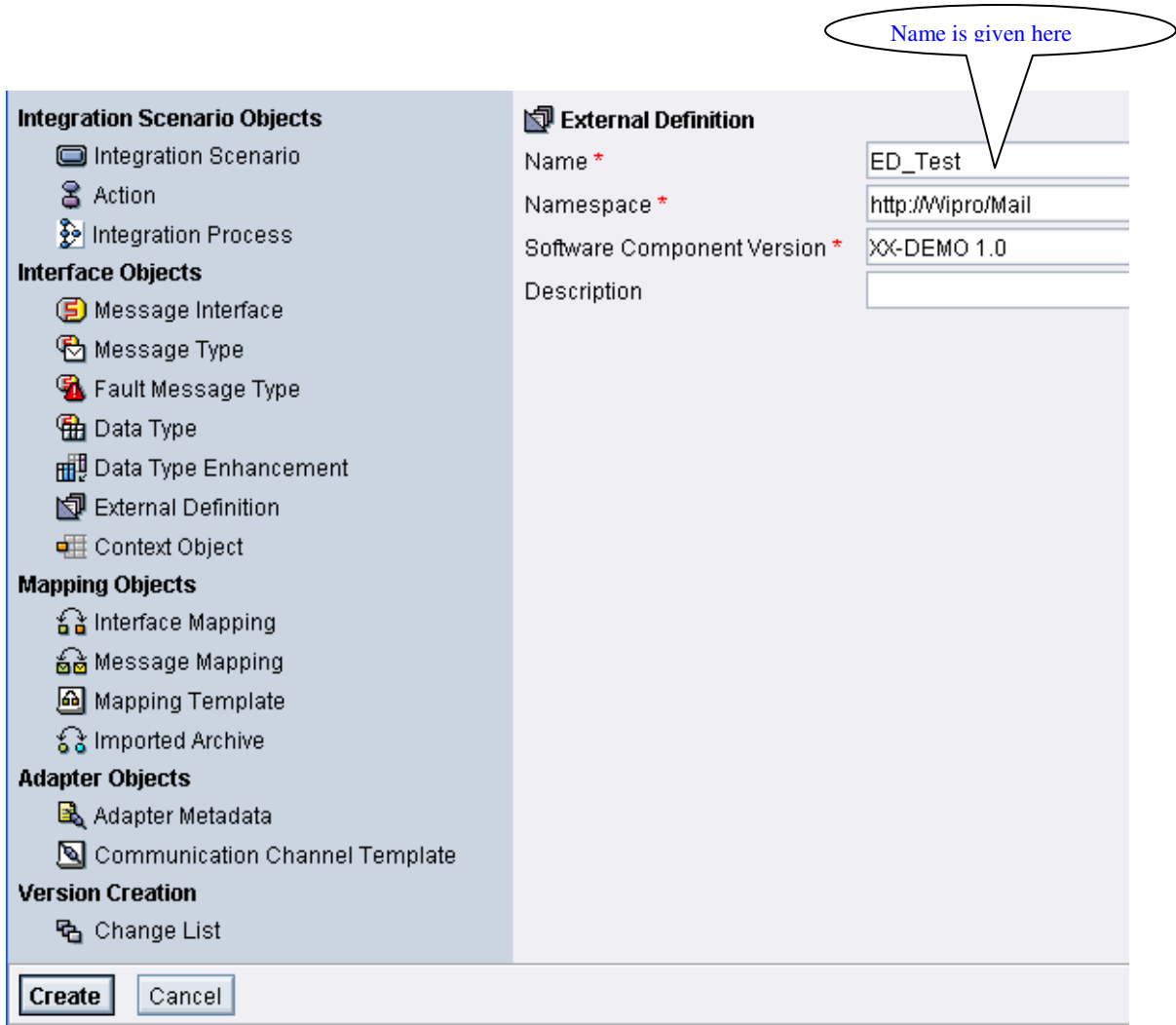
Integration Repository-Design

- 1) Create software component
- 2) Create Namespace under the software component that is created in the above step.
- 3) Now import Web Service metadata in the form of a WSDL file. Create a new External Definition. Import the WSDL into External Definitions in Namespace created. This External Definition acts as a Target Data Type. While importing WSDL file make sure that category is wsdl. Observe the Request and Response message generated, under message tab in the External definition.

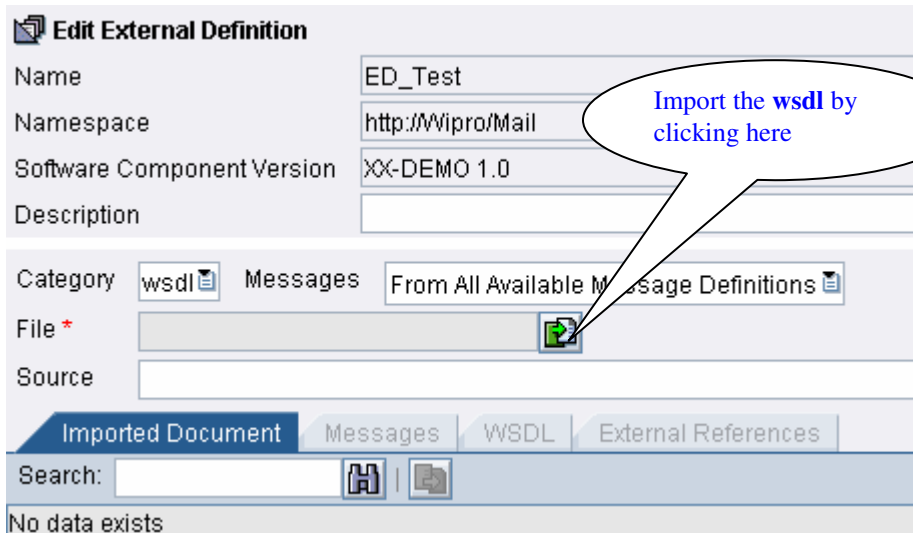
Go to Integration Repository->Software Component Version (i.e. created in step1) ->Namespace (i.e. created in step2) ->Interface Objects->External Definition



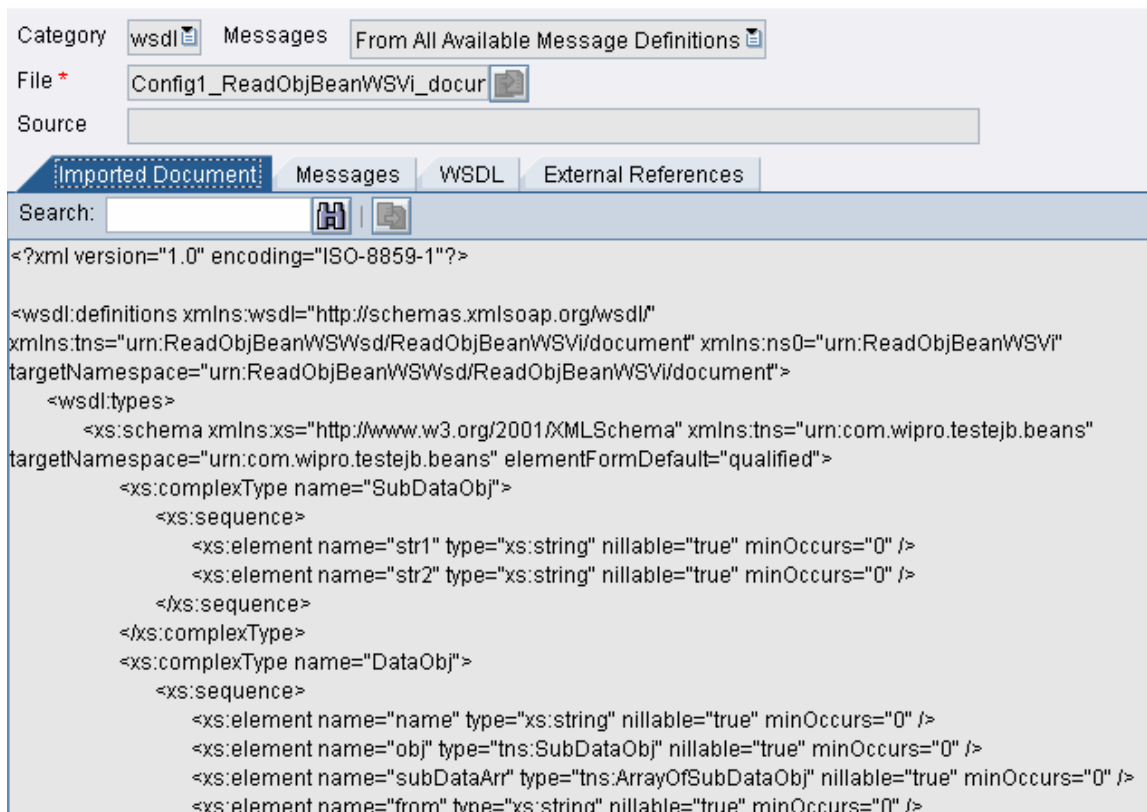
After clicking on “New” following screen will appear.



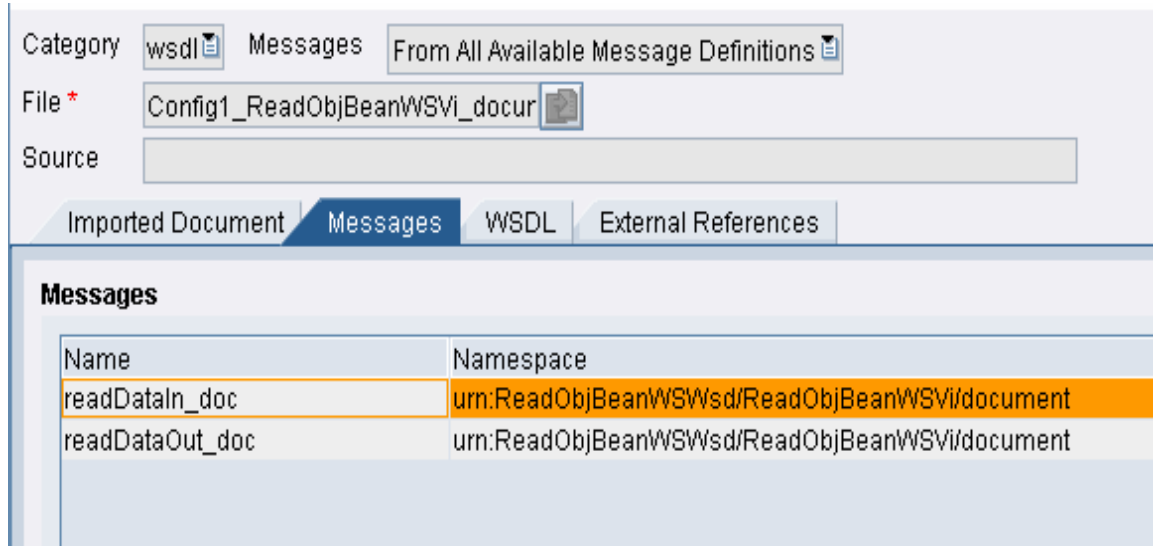
Give new name for External Definition and press Create Button. Then following screen will appear.



After importing valid WSDL, below screen will appear. Make sure that **category** is “wsdl” and **Messages** is “From All Available Message Definitions” in the following screen.



Click on **Messages** Tab and look for Messages in **Messages** window.



- 4) Create Data Type for Input File.
- 5) Create Message Type for Input File.
- 6) Create two Message Interfaces. 1) Outbound, Asynchronous 2) Inbound, Synchronous. In Inbound interface, Input message type should be request/input message for Web Service and Output message type should be response message of Web Service.
- 7) Create a direct one-to-one Message Mapping. Here Source is File Message type and Target is used as input message type of Web Service i.e. from External Definition.
- 8) Now create new Interface Mapping Object. Source Interface is File and Target Interface is WebService.
- 9) Save all and Activate all.

Integration Directory-Configuration

- 1) Create new Configuration Scenario.
- 2) Create Sender and Receiver Communication Channels with respect to specific Business System/Service. Parameters for Sender Communication channel are:
Adapter Type- File, Sender.
- 3) Parameters for Receiver Communication Channel are:
Adapter Type: SOAP, Receiver.
Transport Protocol: HTTP
Message Protocol: SOAP 1.1
Adapter Engine: Integration Server
 Under Connection Parameters provide the following:

Target URL: Enter the complete address (URL) of the Web service provider that you want to send the message to. If the server requires a user authentication, specify the following:

- Specify the *User*.
- Specify the *Password* and confirm.

Hint: Here Target URL means, URL of WSDL.

Display Communication Channel		Status
Communication Channel	CC_SOAP_Receiver	
Party		
Service	BS_File_SOAP	
Description	SOAP Receiver to invoke Webservice	
<div style="display: flex; justify-content: space-around; border-bottom: 1px solid black;"> Parameters Identifiers Module </div>		
Adapter Type *	SOAP	http://sap.com/xi/XI/System SAP BASIS 6.40
<input type="radio"/> Sender <input checked="" type="radio"/> Receiver		
Transport Protocol *	HTTP	
Message Protocol *	SOAP 1.1	
Adapter Engine *	Integration Server	
Connection Parameters		
Target URL *	http://sapntwr.wipro.com:50000/ReadObjBeanWS/Config1?wsdl&style=c	
User		
Password	<input type="password"/> = <input type="password"/>	
<input type="checkbox"/> Use Proxy		

- 4) Create Sender Agreement for File Interface.
- 5) Create new Receiver Determination.
- 6) Create new Interface Determination.
- 7) Create new Receiver Agreement.

8) Save All and Activate All.

To Test the Scenario, follow the steps below.

- 1) Keep the file in the specified directory (as configured in the file adapter).
- 2) Check the mail box of the respective recipients defined in the file. It should have the mail with attachment.

Disclaimer & Liability Notice

This document may discuss sample coding, which does not include official interfaces and therefore is not supported. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing of the code and methods suggested here, and anyone using these methods, is doing it under his/her own responsibility.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of the technical article, including any liability resulting from incompatibility between the content of the technical article and the materials and services offered by SAP. You agree that you will not hold SAP responsible or liable with respect to the content of the Technical Article or seek to do so.

Author Bio

Krishna Moorthy is currently working as a Senior Software Engineer for NetWeaver Practice in Wipro Technologies Bangalore India. He worked as a PeopleSoft Technical Consultant, Mercury Consultant, and ABAP Consultant and now he is playing a role of SAP XI Consultant. His interests are EP and RFID.

Copyright © 2005 SAP AG, Inc. All Rights Reserved. SAP, mySAP, mySAP.com, xApps, xApp, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product, service names, trademarks and registered trademarks mentioned are the trademarks of their respective owners.