Async/Sync and Sync/Async Bridge Scenarios
Configuring Async/Sync Bridge and Sync/Async Bridge without BPM
TABLE OF CONTENTS

1 PREFACE ............................................................................................................................................. 3
1.1 Constraints ....................................................................................................................................... 3
1.2 Definition ......................................................................................................................................... 3
1.3 Intended Audience ........................................................................................................................... 3
1.4 Structure .......................................................................................................................................... 3

2 INTRODUCTION AND PREREQUISITES ...................................................................................... 3
2.1 Introduction ...................................................................................................................................... 3
2.2 Prerequisites ..................................................................................................................................... 3

3 VARIANT 1: ASYN/SYNC BRIDGE BY MEANS OF MODULE PROCESSOR ....................................... 4
3.1 Introduction ...................................................................................................................................... 4
3.2 Async/Sync Bridge ............................................................................................................................ 4
3.3 Test scenario .................................................................................................................................... 4

4 PREPARATION .................................................................................................................................... 4
4.1 User Permissions .............................................................................................................................. 4
4.2 Preconditions .................................................................................................................................... 5

5 JIDOC-RFC-SOAP .............................................................................................................................. 7
5.1 Overview ......................................................................................................................................... 7
5.2 Description ..................................................................................................................................... 8
5.3 Repository Objects ........................................................................................................................... 8
5.4 Configuration Objects ...................................................................................................................... 10
5.4.1 Configuration Overview ............................................................................................................. 10
5.4.2 Integrated Configuration Objects in Integration Directory ......................................................... 10
5.4.3 Test Description .......................................................................................................................... 15

6 VARIANT 2: SYNC/ASYN BRIDGE BY MEANS OF MODULE PROCESSOR ................................... 17
6.1 Introduction ...................................................................................................................................... 17
6.2 Async/Async Bridge ........................................................................................................................ 17
6.3 Test scenario ................................................................................................................................... 18

7 PREPARATION .................................................................................................................................... 18
7.1 User Permissions .............................................................................................................................. 18

8 SOAP - FILE ...................................................................................................................................... 18
8.1 Overview ......................................................................................................................................... 18
8.2 Description ..................................................................................................................................... 18
8.3 Repository Objects ........................................................................................................................... 19
8.4 Configuration Objects .................................................................................................................... 20
8.4.1 Configuration Overview ............................................................................................................. 20
8.4.2 Integrated Configuration Objects in Integration Directory ......................................................... 20
8.4.3 Test Description............................................................................................................................ 26
Pushing the Message ............................................................................................................................. 26
Channel Monitoring .............................................................................................................................. 27
1 PREFACE

1.1 Constraints
The texts, references, and graphics contained in this manual have been compiled with utmost care; nevertheless, it is impossible to guarantee that they are fully without error. SAP cannot assume any responsibility for the correctness or completeness of the following documentation; the user alone is responsible for verifying the information contained therein. SAP will only assume liability for damage arising from the use of this documentation – irrespective of the pertinent legal basis – in the case of intentional or active negligence, under no other circumstances will a warranty be made.

1.2 Definition
This manual describes simple application cases for the de-central adapter engine for process integration and all the configuration steps that are necessary to execute the application cases on the basis of SAP NetWeaver 7.31.

1.3 Intended Audience
This manual is intended to be used by both technology and application consultants.

1.4 Structure
The structure of this document follows the sequence of steps required to configure and run the use cases. In this document, the scenarios are developed in swing client and Integrated Configuration Objects are created. The same scenarios can be developed using Integration iFlows (NWDS/eclipse/IntegrationTool).

2 INTRODUCTION AND PREREQUISITES

2.1 Introduction
The following use case variants are defined in this test:

<table>
<thead>
<tr>
<th>Variant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variant 1: Async/Sync Bridge by means of module processor</td>
<td>This test case describes how to connect an asynchronous system to a synchronous system by means of an async/sync bridge without using BPM</td>
</tr>
<tr>
<td>Variant 2: Sync/Async Bridge by means of module processor</td>
<td>This test case describes how to connect a synchronous system to an asynchronous system by means of a sync/async bridge without using BPM</td>
</tr>
</tbody>
</table>

2.2 Prerequisites
The web service tool used in this test case is SOAP workbench (from Apache) for testing. Any external SOAP Request/Response tool which supports web service can be used for testing. Eg, SOAP UI. Refer [http://www.soapui.org/Getting-Started/web-service-sample-project.html](http://www.soapui.org/Getting-Started/web-service-sample-project.html) for steps to use this tool.
3 VARIANT 1: ASYN/SYNC BRIDGE BY MEANS OF MODULE PROCESSOR

3.1 Introduction
This test case describes how to connect an asynchronous system to a synchronous system by means of an async/sync bridge without using BPM. This is required if you have a sender system that only supports asynchronous message processing to communicate with a purely synchronous receiver application. The async/sync bridge handles the conversion from an asynchronous message sent from the sender to the receiver into a synchronous message, and the other way round for the synchronous reply.

3.2 Async/Sync Bridge
The asynchronous request and response messages are mapped to a synchronous call by means of the module processor. The overall communication sequence looks like below diagram: an asynchronous request message is converted to a synchronous request in the module processor. The synchronous system sends a response which is converted to an asynchronous response message in the module processor which is then passed to receiver.

![Diagram](image)

**Figure 1: Message flow for async/sync scenario by means of the adapter module processor**

3.3 Test scenario
The scenario in this document refers to send IDOC to RFC and response back from RFC to web service.

- PI receives the IDOC from ECC system. JIDOC Sender Adapter is used in PI to get the data from ECC.
- PI does the request mapping and sends the request to BAPI and gets back the response from BAPI via RFC Receiver adapter.
- The response from BAPI is sent to web service via SOAP adapter

4 PREPARATION

4.1 User Permissions
The tester should have permission to log on the PI test system and to open the SOA Monitors in the NetWeaver Administrator. Also the user should have access permissions to R/3 system to perform the testing.

Here PI 7.31 AEX, R/3 system, PI 7.31 Double Stack are used to execute this test. The system environment may differ in your test executions.
Note: The RFC program is created in dual stack server and the request and response is fetched by 7.31 AEX from dual stack system

### 4.2 Preconditions

1. **Configurations in NWA**

<table>
<thead>
<tr>
<th>Actions</th>
<th>System</th>
<th>Details</th>
</tr>
</thead>
</table>
| Settings in Application Resources | PI 7.31 | 1. Open NWA ([http://<host>:<port>/nwa](http://<host>:<port>/nwa))  
- Configuration > Infrastructure > Application Resources  
- Filter for Resource Adapter, inboundRA  
- Click on Properties tab below:  
  - Set value for "MaxReaderThreadCount" between 5 to 10  
  - Set Value of ProgramID as JIDOC_PID_DEMO (This must be used when creating the RFC in ECC system)  
  - Set Local as false (don’t change)  
  - Set values for default GatewayServer as host of PI SystemSet  
  - Save Changes |

<table>
<thead>
<tr>
<th>Actions</th>
<th>System</th>
<th>Details</th>
</tr>
</thead>
</table>
| Create JCO RFC Provider destination | PI 7.31 | 2. Open NWA ([http://<host>:<port>/nwa](http://<host>:<port>/nwa))  
- Configuration > Infrastructure > JCO RFC Provider  
- Create JCO RFC Provider destination with name JIDOC_PID_DEMO in NWA where PIDEV refers to the PI System ID  
- Save Changes |

2. **Configurations at Sender System (SAP ECC)**

This section describes all the configurations needed in the Sender SAP System (ECC) for sending an IDoc to PI

<table>
<thead>
<tr>
<th>Actions</th>
<th>Details</th>
</tr>
</thead>
</table>
| IDOC Creation | The IDoc used in variant1 (JIDOC-RFC-SOAP) is a custom IDoc (ZAIRFLG)  
In SAP ECC system,  
- Goto transaction code(tcode) we31 to create segment (ZAIRFLGSEG) which will have the fields as below:  
  ![Development segments: Display segment definition ZAIRFLGSEG000](http://wiki.scn.sap.com/wiki/display/ABAP/Steps+to+create+custom+IDOC)  
- Goto tcode we30 to create the IDOC (ZAIRFLG)  
Below link can be referred for creation of custom IDOC in SAP ECC system and the custom IDoc created should be imported in ESR [http://wiki.scn.sap.com/wiki/display/ABAP/Steps+to+create+custom+IDOC](http://wiki.scn.sap.com/wiki/display/ABAP/Steps+to+create+custom+IDOC) |
Async/Sync and Sync/Async Bridge Scenarios

| RFC Destination       | On your SAP ECC system in order to send the IDOCs you need to create an RFC destination of type T (TCP/IP) 
|                       | Go to Transaction SM59 create a new RFC destination of Type T 
|                       | • Select the radio button Registered Server Program 
|                       | • In the program ID enter the program ID from inboundRA of NWA 
|                       | • Enter the gateway host and gateway service of your PI server 
|                       | After you configure that you should be able to check the configuration using the test connection button on the RFC destination. |
| Port                   | Goto Transaction WE21 
|                       | • Click on the Transactional RFC from Ports (left tree panel) 
|                       | • Click on Create Button 
|                       | • Give Port Name (SAP<SID>) here or select generate port name and click on continue. 
|                       | • Give the description of the port and select RFC destination of type T created above. Now click on save. |
| Logical System         | • Click on create Button 
|                       | • Give name for your Logical System (<SID>CLNT<ClntNo.>) 
|                       | • Now click on save |
| Partner Profile        | Create Partner Profile with outbound parameter (WE20) to be send to the PI receiver system 
|                       | 1. Go to Transaction WE20 
|                       | 2. Select Partner Type LS 
|                       | 3. Click on Create Button 
|                       | 4. Give Logical System Name which we have just created above as Partner No, Partner Type should be LS, Agent (some valid data), Language (EN). Now click on Save |
Async/Sync and Sync/Async Bridge Scenarios

5. Now create an Outbound Parameter.
   - Select the required Message Type.
   - Select the Receiver Port (which we have created in step 2 in this example).
   - Select Transfer IDoc Immediately option in Output mode for Immediate Testing. Select Basic Type Save.
   - Enter all credentials in the post processing permitted agent tab.

3. The tester should fulfill the ESR Objects and the Configuration Objects as shown in the subsequent sections of this document.

5  JIDOC-RFC-SOAP

5.1 Overview
This test case describes how to connect an asynchronous system to a synchronous system by means of an async-sync bridge.
5.2 Description

SAP R/3 system sends the IDOC using Java IDOC Sender Adapter. PI does the request mapping and sends the request to BAPI and gets back the response of BAPI via RFC adapter. The response received is sent to web service via SOAP Adapter.

To implement the scenario, we need to define two ICO. One for routing the request message from the IDOC to the RFC, and one for routing back the response message to web service.

5.3 Repository Objects

In the Enterprise Service Repository – on PI test system:

- We should have the custom Idoc created as in section 4.2-2 and RFC “SXI DEMO_AIRL_FLIGHT_CHECKAVAIL” under Imported Objects folder. In this scenario the custom Idoc created and imported is “ZAIRFLG_MSG.ZAIRFLG”

Steps to import Idoc:
1. Double click and open the software Component version and provide the details in “Interface import”. The Client and Message Server is of the ECC system in which IDoc is developed

<table>
<thead>
<tr>
<th><strong>Interface Import</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connection Data for Import from SAP System</strong></td>
</tr>
<tr>
<td>System</td>
</tr>
<tr>
<td>Client</td>
</tr>
<tr>
<td>Message Server</td>
</tr>
<tr>
<td>Group</td>
</tr>
</tbody>
</table>

2. Right click on “Imported Objects-IDocs” and click on “Import of SAP Object”
3. Provide the Application Server and System Number in which the custom Idoc is been developed and also provide UserName and password and click Continue and then Finish button
4. Once the Idoc is imported, the structure (fields) of IDOC is as below:
Async/Sync and Sync/Async Bridge Scenarios

Steps to import RFC:
1. Follow the Step 1 as defined in import idoc above.
2. Right click on "Imported Objects-RFC" and click on "Import of SAP Object"
3. Provide the Application Server and System Number in which the RFC is been developed and also provide the UserName and password and click Continue and then Finish button

- The Message Mapping FSACheck_Agency2AirlineRFC_Req (as below) and Operation Mapping FSACheck_Agency2Airline should be created

To do this – call the URL http://<server>:<host>/dir of the Process Integration system and open the link to the Enterprise Services Builder.

The following graphic shows the component view
5.4 Configuration Objects

5.4.1 Configuration Overview

5.4.2 Integrated Configuration Objects in Integration Directory

The first ICO defines the routing from the JIDOC to the RFC. In the Inbound Processing, the JIDOC sender channel (JIDOC_Sender_Test) is specified.
Async/Sync and Sync/Async Bridge Scenarios

On tab **Receiver** of the ICO, the RFC Receiver Business Component is defined.

On tab **Receiver Interfaces**, the Operation Mapping for request mapping is set.

On tab **Outbound Processing**, the RFC receiver channel is specified.
In the RFC receiver channel, the modules AF_Modules/RequestResponseBean and AF_Modules/ResponseOnewayBean are defined in the right sequence in Module tab.

The specified parameters points to the second ICO.
**RequestResponseBean**

**Use**
RequestResponseBean module is used to convert an asynchronous request message to a synchronous request message.
If the thread that enters the module is part of a transaction, the transaction is suspended.

**Entries in Module Configuration**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Sender Adapter</th>
<th>Receiver Adapter</th>
</tr>
</thead>
<tbody>
<tr>
<td>passThrough</td>
<td>If you want the module to forward the message to the next module in the module chain, enter <code>true</code>. The default value is <code>false</code>. The module then calls the messaging system.</td>
<td>If you want the module to forward the message to the next module in the module chain, enter <code>true</code>. The default value is <code>false</code>. The module then calls the <code>ResponseOnewayBean</code> module</td>
</tr>
</tbody>
</table>

**ResponseOnewayBean**

**Use**
A standard module used for converting an inbound message to an asynchronous message. If the thread that enters the module is part of a suspended transaction, the transaction is resumed.

**Parameters** (`interface`, `interfaceNamespace` and `replaceInterface`) mentioned in module Configuration points to the second ICO.

The **second ICO** defines the routing from the RFC to the Web Service. In the *Inbound Processing*, the SOAP sender channel is defined. Note, that in the header of the ICO the corresponding receiver communication component has to be specified as virtual receiver.
(The receiver of the first ICO become Sender and the Sender has to be defined as Virtual Receiver)

In the SOAP sender channel, the *Quality of Service* is set as *Exactly Once*.
Async/Sync and Sync/Async Bridge Scenarios

On tab **Receiver**, the receiver Business system is defined.

On tab **Receiver Interfaces**, the asynchronous response inbound interface is defined as receiver interfa

On tab **Outbound Processing**, the SOAP receiver Channel is set.
5.4.3 Test Description

Testing the scenario JIDOC \(\rightarrow\) PI \(\rightarrow\) RFC \(\rightarrow\) PI \(\rightarrow\) SOAP please follow the steps below:

a) Check whether both (Sender & Receiver) channels being used in this scenario are started. If not then start them via Channel Monitoring of NWA/RWB.

b) Execute transaction “we19” in Sender R/3 system for triggering an IDoc. Enter IDoc Types as “ZAIRFLG”.

c) Double click on EDIDC(Control Record) and enter the Sender, Receiver details and Message Type ZAIRFLG_MSG
d) Double click on IDOC type and enter the record data and click on Standard Outbound Processing

e) The IDoc is immediately sent to XI system. A message confirming the same appears at the bottom of the page. “An Idoc < IdocNo> added”.

f) The same can be confirmed by using transaction “we05” or “we02” and in the Direction box, choose “Outbound”. The list shows all the IDoc that have left the R/3 system.

On Sender Side, Check for “Idoc No.” which should be same as above sent from R/3 system. Get the Msg. ID for this flow of Idoc

This Msg. ID should be checked within Message flowing on Receiver Comm. Channel RFC_AirlineFlight_Receiver

To check on the modules process, goto Message Log of RFC Channel
To check the response from RFC sent to web service via SOAP Receiver Channel can be checked in the communication channel monitoring of channel SOAP_Receiver_ResponseFromRFC.

The response can be checked in SOAP Workbench/external tool. Here in SOAP Workbench the response can be checked as URL mentioned in in Target URL of SOAP receiver channel is of SOAP Workbench.

6 VARIANT 2: SYNC/ASYN BRIDGE BY MEANS OF MODULE PROCESSOR

6.1 Introduction
This test case describes how to connect a synchronous system to an asynchronous system by means of an async/sync bridge without using BPM.

6.2 Sync/Async Bridge
A synchronous call is mapped to asynchronous request and response messages by means of the module processor. In the module processor of the receiver adapter, the synchronous message is converted to an asynchronous request message. The synchronous call waits until the receiver adapter sends a reply.
### 6.3 Test scenario
The purpose of this scenario is to configure SOAP to FILE communication where SOAP is sync and FILE is Async.

### 7 PREPARATION

#### 7.1 User Permissions
The tester should have permission to log on to the PI test system and to open the SOA Monitors in the NetWeaver Administrator. Here PI 7.31 AEX is used to execute the test case. The system environment may differ in your test executions.

SOAP Workbench tool is used to respectively send and receive the messages.

### 8 SOAP - FILE

#### 8.1 Overview
This test case describes how to connect a synchronous system to an asynchronous system by means of an sync/async bridge.

#### 8.2 Description
Web service sends the data to PI (SOAP Sender Adapter used) and PI does the request mapping and sends the request to FILE and the filename placed is with the message ID to correlate with the response message. Once the file is placed, the response (Status) is sent to web service via SOAP Adapter.

To implement the scenario, we need to define two ICO. One for routing the request message from the Web Service client to the File system, and one for routing back the response message.
8.3 Repository Objects

In the Enterprise Service Repository – on PI test system the below screenshot objects are created and used in this test case. The field structure can be referred in the Request and Response Mapping given below during creation.

Request Mapping: MM_SOAP_to_File

Response Mapping: MM_Response_FileToSOAP

The following graphic shows the component view:
8.4 Configuration Objects

8.4.1 Configuration Overview

8.4.2 Integrated Configuration Objects in Integration Directory

The first ICO defines the routing from the Web Service client to the File System. In the Inbound Processing, the SOAP sender channel is specified.
In the sender channel of type SOAP, the Quality of Service is set to Best Effort.

On tab Receiver of the ICO, the receiver communication component is defined

On tab Receiver Interfaces, the Operation Mapping for request mapping is set
Async/Sync and Sync/Async Bridge Scenarios

On tab **Outbound Processing**, the FILE receiver channel is specified.

In the Parameters tab, maintain the target directory and set the file name scheme as "%mid%" (without quotes), so we can set the filename with the message id through variable substitution. Set the construction mode as create (do not append any info after the message id) and enable variable substitution, with a single entry: name="mid", reference="message:message_id" (also, without quotes).
For more details on variable substitution made in File Receiver channel:

In the Module tab, insert the "AF_Modules/RequestOnewayBean" module before the CallSapAdapter module, and set its parameter "passThrough" with value "true"; also, insert the "AF_Modules/WaitResponseBean" module after the CallSapAdapter module.

The second ICO defines the routing of the response message from File to the Web Service client. In the Inbound Processing, the File Sender channel is defined.
In the Parameters tab, maintain the source directory and set the file name as "*" (without extension); set the polling interval for a short time, for example, 10 seconds (a long polling time may cause a timeout in the sync SOAP); make sure you set the "Set Adapter-Specific Message Attributes" flag, with the File Name flag active.
The file picked is been archived.
In the Module tab, replace the "CallSapAdapter" module with the "AF_Modules/NotifyResponseBean" (since the message doesn't need to be sent to the Integration Engine, but rather to the waiting WaitResponseBean); also, before the NotifyResponseBean, insert the "AF_Modules/DynamicConfigurationBean" module, with the following parameters and respective values: name="key.0" value="write http://sap.com/ixi/XI/System/FileName", name="value.0" value="message.correlationId".
The DynamicConfigurationBean module will get the filename ASMA-Adapter Specific Message Attributes (which was previously set by the adapter) and write its value in the message.correlationId attribute, before the message is sent to the waiting process (WaitResponseBean) by the NotifyResponseBean module.

8.4.3 Test Description

Testing the scenario SOAP - > FILE please follow the steps below:

**Pushing the Message**

a) Check whether both (Sender & Receiver) channels being used in this scenario are started. If not then start them via Channel Monitoring of NWA/RWB.

b) Open the external SOAP Request/Response tool being used

c) In the field URL set the below value.


   Set the host and port of the PI server.

   **Note:** The URL specified here is for SOAP Workbench tool when used. If any other external tool is used, the URL should be changed accordingly.

d) Select “Use Authentication”. Then enter your username and password for the PI server.

e) Under the label SOAP delete the existing text. Now copy the content from the payload file below and paste it in the editbox under label SOAP.

   ```xml
   <?xml version="1.0" encoding="utf-8"?>
   <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
   xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <soapenv:Body>
   <ns0:MT_SOAPRequest xmlns:ns0="http://sap.com/xi/SyncAsyncBridge">
   <Row>
   ```

26
f) Click on the Send button. It will push a message.

g) As QoS is Best Effort in SOAP Sender adapter the response is received as below:

![Response Message](image)

**Channel Monitoring**

Open the **PI Communication Channel Monitoring** (NWA → SOA → Monitoring → PI Communication Channel Monitoring). Following screen will appear

![PI Communication Channel Monitoring](image)

Click on the link “Advanced”. The following screen appears. Set the Sender Communication Component “XIVERI_AEX_Sender_SyncAsyncBridge” in the edit box “**Communication Component**” & hit **GO** button.

![Communication Component](image)
As a result we see successful message in the SOAP Sender channel

Goto File Receiver Communication Channel to check the message received.

To check on the modules process, goto Message Log of either SOAP Sender Channel or File Receiver Channel

The response is sent from File to SOAP can be checked in the communication channel monitor of channel FileSenderRes
© 2014 SAP AG. All rights reserved.

SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP BusinessObjects Explorer, StreamWork, SAP HANA, 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 other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects Software Ltd. Business Objects is an SAP company.

Sybase and Adaptive Server, Anywhere, Sybase 365, SQL Anywhere, and other Sybase products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Sybase Inc. Sybase is an SAP company.

Crossgate, m@tic EDDY, B2B 360°, and B2B 360° Services are registered trademarks of Crossgate AG in Germany and other countries. Crossgate is an SAP company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies (“SAP Group”) for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.