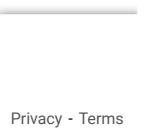




PROXY

File to Inbound Proxy Scenario SAP PI/PO Single Stack

POSTED ON MAY 12, 2019 BY ISURU FERNANDO



Privacy - Terms



Additionally, you can also refer my articles on [Outbound ABAP Proxy](#) and functionality of [Fault Messages](#) for Inbound ABAP Proxy interfaces.

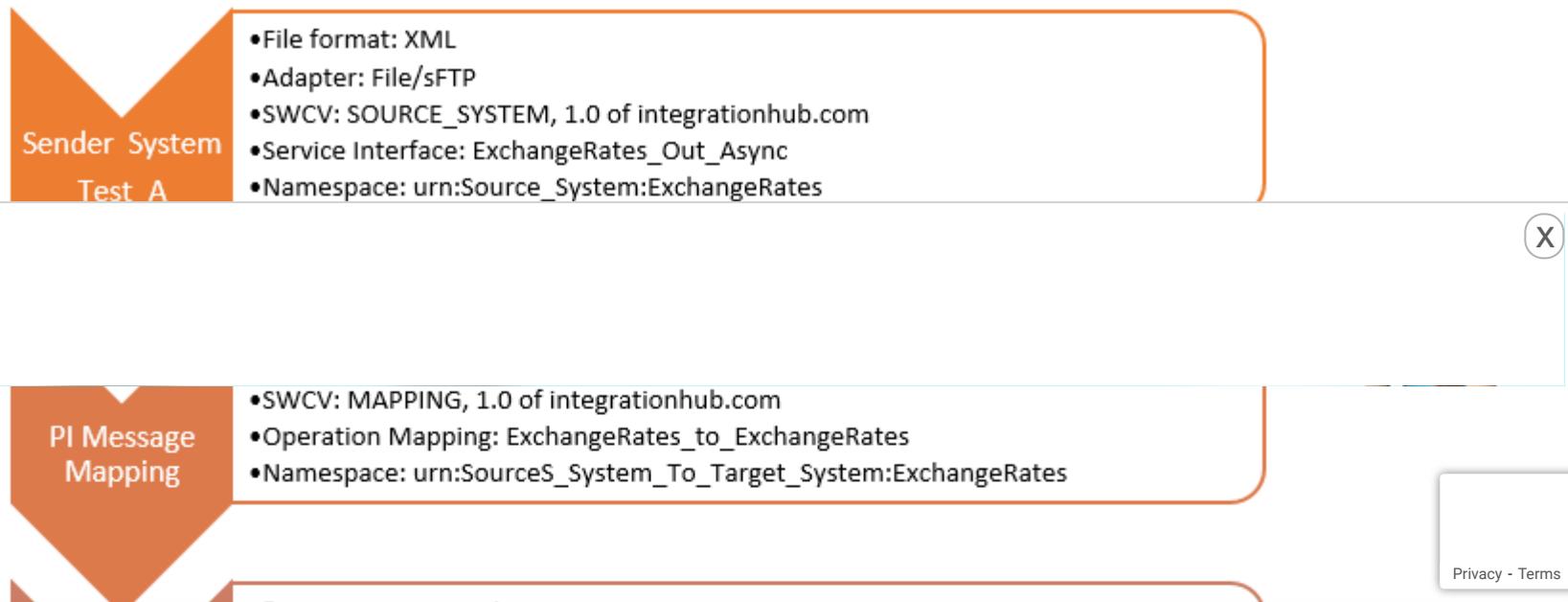
SAP Versions used in the illustration:

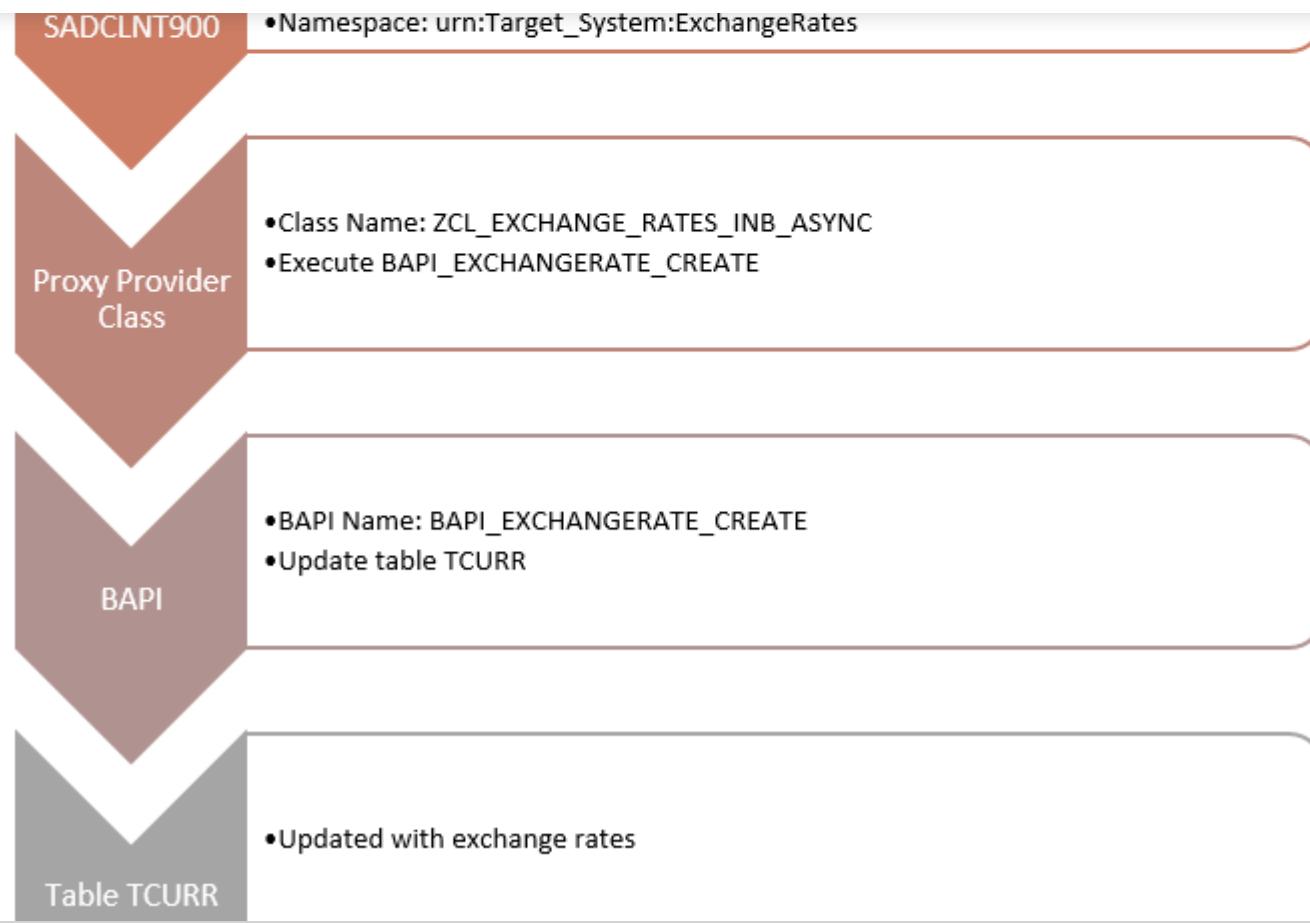
Inbound Proxy Scenario Overview:

Let's assume we have an sFTP file system that sends exchange rates in XML format. Exchange rates in XML file should be extracted, transformed and sent to target SAP system via [SAP PI/PO](#).

Information is sent to SAP system as a Proxy message and exchange rates should be updated in SAP table TCURR (Exchange Rates). However, we will use BAPI 'BAPI_EXCHANGERATE_CREATE' to update the exchange rates table TCURR as it's the preferred best practice.

Interface Flow Overview:





XML file (SFTP) to SAP Proxy Inbound Interface Example Overview.

Sender File (XML) Format:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ns0:ExchangeRates xmlns:ns0="urn:Source_System:ExchangeRates">
```



```

8   </Rates>
9 </ns0:ExchangeRates>
```

Receiver Proxy Format:

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <ns1:ExchangeRates xmlns:ns1="urn:Target_System:ExchangeRates">
3   <EXCH_RATE>
4     <RATE_TYPE>M</RATE_TYPE>
5     <FROM_CURR>USD</FROM_CURR>
6     <TO_CURRENCY>EUR</TO_CURRENCY>
7     <VALID_FROM>2019-01-03</VALID_FROM>
8     <EXCH_RATE>0.1</EXCH_RATE>
9     <FROM_FACTOR>1</FROM_FACTOR>
10    <TO_FACTOR>1</TO_FACTOR>
11    <EXCH_RATE_V>0.0</EXCH_RATE_V>
12    <FROM_FACTOR_V>0</FROM_FACTOR_V>
13    <TO_FACTOR_V>0</TO_FACTOR_V>
14  </EXCH_RATE>
15 </ns1:ExchangeRates>
```

XML File to Proxy Message Mapping

Mapping logic between the XML file sender message and target proxy message is as follows.

Proxy Target Element	Transformation/Mapping Logic	XML Source Element
RATE_TYPE	Constant 'M'	
VALID_FROM	Constant 'MM dd yyyy to yy yy mm dd' → ExchangeRates/Rates/validFromDate	
EXCH_RATE		ExchangeRates/Rates/ <i>ExchangeRate</i>
FROM_FACTOR	Constant '1'	
TO_FACTOR	Constant '1'	



TO_FACTOR_V

Constant '0'

Functionality of the Proxy Class:

ABAP Proxy class update the SAP exchange rates table TCURR using '**BAPI_EXCHANGERATE_CREATE**'. In addition, proxy ABAP logic assigns the exchange rate values from the proxy message to the BAPI input parameter.

BAPI accepts the exchange rates through importing parameter EXCH_RATE. ABAP logic assigns the exchange rates in the proxy message to EXCH_RATE parameter of the **BAPI_EXCHANGERATE_CREATE**.

The screenshot shows the SAP Fiori interface for the BAPI_EXCHANGERATE_CREATE function module. The top navigation bar has tabs for Function module, Import, Export, Changing, Tables, Exceptions, and Source code. The 'Function module' tab is active. Below the tabs is a toolbar with icons for search, refresh, and other functions. A table lists the parameters:

Parameter Name	Type	Associated Type	Default value	Op...	Pa...	Short text
EXCH_RATE	LIKE	BAPI1093_0		<input type="checkbox"/>	<input checked="" type="checkbox"/>	Exchange Rate To Be Written
UPD_ALLOW	LIKE	BAPI1093_2-UPD_ALLOW	SPACE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	'X' = Update of Existing Exchange Rate is Also Permitted
CHG_FIXED	LIKE	BAPI1093_2-CHG_FIXED	SPACE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	'X' = Permit Changes to Fixed Exchange Rate
DEV_ALLOW	LIKE	BAPI1093_2-DEV_ALLOW	'000'	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Permitted Difference to Old Exchange Rate as Percentage
				<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	



Component	Typing...	Component Type	Data Ty...	Length	Deci...	Short Description
RATE_TYPE	Types ▾	KURST_CURR	CHAR	4	0	Exchange rate type
FROM_CURR	Types ▾	FCURR_CURR	CUKY	5	0	From currency
TO_CURRENCY	Types ▾	TCURR_CURR	CUKY	5	0	To-currency
VALID_FROM	Types ▾	GDATU_CUR	DATS	8	0	Date from Which Entry Is Valid
EXCH_RATE	Types ▾	UKURSP	DEC	9	5	Direct Quoted Exchange Rate
FROM_FACTOR	Types ▾	FFACT_CURR	DEC	9	0	Ratio for the "from" currency units
TO_FACTOR	Types ▾	TFACT_CURR	DEC	9	0	Ratio for the "to" currency units
EXCH_RATE_V	Types ▾	UKURSM	DEC	9	5	Indirect Quoted Exchange Rate
FROM_FACTOR_V	Types ▾	FFACT_CURR	DEC	9	0	Ratio for the "from" currency units
TO_FACTOR_V	Types ▾	TFACT_CURR	DEC	9	0	Ratio for the "to" currency units

DDIC data type of Exch_rate parameter of the BAPI

File to Proxy Interface Development Steps:

1. Create Outbound file Service Interface and Inbound Proxy Service Interface.
2. Define Outbound file and Inbound Proxy message mapping.
3. Implement Message Mapping between XML and Proxy message.
4. Generate Proxy class in SAP back-end using transaction SPROXY.
5. Create sFTP sender Communication Channel and Proxy (SOAP) receiver Channel.
6. Create sFTP sender Communication Channel and Proxy (SOAP) receiver Channel.
7. Create sFTP sender Communication Channel and Proxy (SOAP) receiver Channel.



The first step is to set up Proxy connectivity. Follow my complete step-by-step guide on [how to configure proxy connection between ECC and SAP PI/PO](#).

You will find information on SLD setup, RFC destination configuration, HTTP Destination configuration, Integration Engine (SXMB_ADM) settings, and Proxy Communication Channel configuration in my previous post. Therefore, I will not elaborate on the steps in this example.

Step 2 – Create Sender XML Data Type and Message Type:

Next step is to create the exchange rate sender Data Type and Message Type in Enterprise Service Repository (ESR). You can either create the Message Type as an External Definition by importing the XSD or by creating a Data Type which reflects the XML data structure.

In this illustration, I have created a Data Type and Message Type to represent the exchange rate sender XML message format.

Sender Data Type:



Software Component Version SOURCE_SYSTEM, 1.0 of integrationhub.com

Description

Classification Free-Style Data Type Quality Schema None

Type Definition XSD

Search

Name	Category	Type	Occurrence
ExchangeRates	Complex Type		
Rates	Element		1..unbounded
FromCurrency	Element	xsd:string	1
ToCurrency	Element	xsd:string	1
ExchangeRate	Element	xsd:decimal	1
ValidFromDate	Element	xsd:date	1

Source Data Type for XML exchange rate sender message

Sender Message Type:





Software Component Version: SOURCE_SYSTEM, 1.0 of integrationhub.com

Description:

Name *: ExchangeRates

Data Type Used: ExchangeRates

XML Namespace: urn:Source_System:ExchangeRates

Structure XSD

Name	Category	Type	Occurrence
ExchangeRates	Element	ExchangeRates	
Rates	Element		1..unbounded
FromCurrency	Element	xsd:string	1
ToCurrency	Element	xsd:string	1
ExchangeRate	Element	xsd:decimal	1
ValidFromDate	Element	xsd:date	1

Input Message Type for XML exchange rate sender message

Step 3 – Create Receiver Proxy Data Type and Message Type:

The function of the Proxy is to update exchange rates in SAP using the Function Module **BAPI_EXCHANGERATE_CREATE**



To update the exchanges rates we need to assign values from Proxy message to input parameter EXCH_RATE of the BAPI. Create the receiver Message Type similar to the structure of EXCH_RATE Data Type. In other words, Data Type should be similar to

[Privacy](#) · [Terms](#)



Designing the Message Type like this allows us to easily move data from the proxy message to the BAPI using [MOVE-CORRESPONDING](#) ABAP statement.

Receiver Data Type:

Display Data Type

Name	ExchangeRates
Namespace	urn:Target_System:ExchangeRates
Software Component Version	TARGET_SYSTEM, 1.0 of integrationhub.com
Description	

Classification: Free-Style Data Type Qualify Schema: None

Type Definition XSD

	Name	Category	Type	Occurrence
▼	ExchangeRates	Complex Type		
▼	EXCH_RATE	Element		1..unbounded
	RATE_TYPE	Element	xsd:string	1
	FROM_CURR	Element	xsd:string	1
	TO_CURRENCY	Element	xsd:string	1
	VALID_FROM	Element	xsd:date	1
	EXCH_RATE	Element	xsd:decimal	0..1

Receiver Date Type for Proxy message

Receiver Message Type:



Software Component Version **TARGET_SYSTEM, 1.0 of integrationhub.com**

Description

Name *

Data Type Used **ExchangeRates**

XML Namespace **urn:Target_System:ExchangeRates**

Structure **XSD**

Name	Category	Type	Occurrence
▼ ExchangeRates	Element	ExchangeRates	
▼ EXCH_RATE	Element		1..unbounded
RATE_TYPE	Element	xsd:string	1
FROM_CURR	Element	xsd:string	1
TO_CURRENCY	Element	xsd:string	1
VALID_FROM	Element	xsd:date	1
EXCH_RATE	Element	xsd:decimal	1
FROM_FACTOR	Element	xsd:decimal	0..1
TO_FACTOR	Element	xsd:decimal	0..1
EXCH_RATE_V	Element	xsd:decimal	0..1
FROM_FACTOR_V	Element	xsd:decimal	0..1
TO_FACTOR_V	Element	xsd:decimal	0..1



Create an Outbound Service interface for XML file sender and an Inbound Service Interface to represent the Proxy message. As we have only one operation, let's use Interface Pattern Stateless (XI30 Compatible) for both interfaces.

Outbound Asynchronous Service Interface:

[Privacy - Terms](#)

The screenshot shows the SAP Integration Hub interface for creating a new software component. The top navigation bar includes the SAP Integration Hub logo, a search icon, and a menu icon.

Software Component Version: SOURCE_SYSTEM, 1.0 of integrationhub.com

Description: [Empty input field]

Definition tab is selected. Other tabs include WSDL, Matching Service Interfaces, and Classifications.

Attributes section (highlighted with a red box):

- Category:** Outbound
- Interface Pattern:** Stateless (XI30-Compatible) Point-to-Point enabled
- Security Profile:** Basic

Event interface Sensitive Data

Operations section:

- Operation:** ExchangeRates_Out_Async (highlighted)

Operation ExchangeRates_Out_Async details:

- Description:** [Empty input field]
- Release State:** Not Released
- Attributes**:
 - Operation Pattern:** Normal Operation
 - Mode:** Asynchronous
- Messages**:
 - Role** **Type** **Name** **Namespace**

Inbound Asynchronous Proxy Interface:

The screenshot shows the SAP Integration Hub interface for creating a new software component. The top navigation bar includes the SAP Integration Hub logo, a menu icon, and search/filter icons.

Software Component Version: TARGET_SYSTEM, 1.0 of integrationhub.com

Description: (empty)

Definition tab is selected. Other tabs include WSDL, Matching Service Interfaces, and Classifications.

Attributes section:

- Category: Inbound
- Interface Pattern: Stateless (XI30-Compatible) Point-to-Point enabled
- Security Profile: Basic

Event interface Sensitive Data

Operations section:

- Operation: ExchangeRates_Inb_Async (selected)

Operation ExchangeRates_Inb_Async details:

- Description: (empty)
- Release State: Not Released

Attributes for the operation:

- Operation Pattern: Normal Operation
- Mode: Asynchronous

Messages section:

Role	Type	Name	Namespace
(empty)	(empty)	(empty)	(empty)

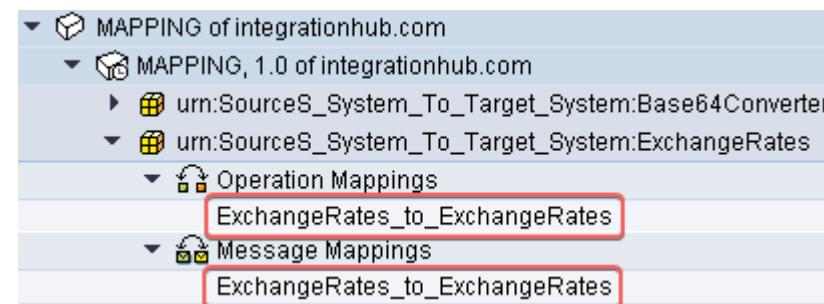
A red box highlights the "Attributes" section and the "Messages" table header.

Step 5 – Implement the Message Mapping between the XML file and the Proxy message:

Next, define the Message Mapping and the Operation mapping.



Operation and Message Mapping:



Message Mapping and Operation Mapping

Message Mapping:

The screenshot shows the SAP Integration Hub interface for a 'File to Inbound Proxy Scenario SAP PI/PO Single Stack - Exchange Rates' scenario. The top navigation bar includes the SAP Integration Hub logo, a search icon, and a help icon. The main area displays two message structures side-by-side:

Message Type: ExchangeRates (Left)

Structure	Occurrences	Type	Description
ns0:ExchangeRates	1..1	p4:ExchangeRates	
Rates	1..unbounded		
FromCurrency	1..1	xsd:string	
ToCurrency	1..1	xsd:string	
ExchangeRate	1..1	xsd:decimal	
ValidFromDate	1..1	xsd:date	

Message Type: ExchangeRates (Right)

Structure	Occurrences	Type	Description
ns1:ExchangeRates	1..1	p5:ExchangeRates	
EXCH_RATE	1..unbounded		
RATE_TYPE	1..1	xsd:string	
FROM_CURR	1..1	xsd:string	
TO_CURRNCY	1..1	xsd:string	
VALID_FROM	1..1	xsd:date	
EXCH_RATE	1..1	xsd:decimal	
FROM_FACTOR	0..1	xsd:decimal	
TO_FACTOR	0..1	xsd:decimal	
EXCH_RATE_V	0..1	xsd:decimal	
FROM_FACTOR_V	0..1	xsd:decimal	
TO_FACTOR_V	0..1	xsd:decimal	

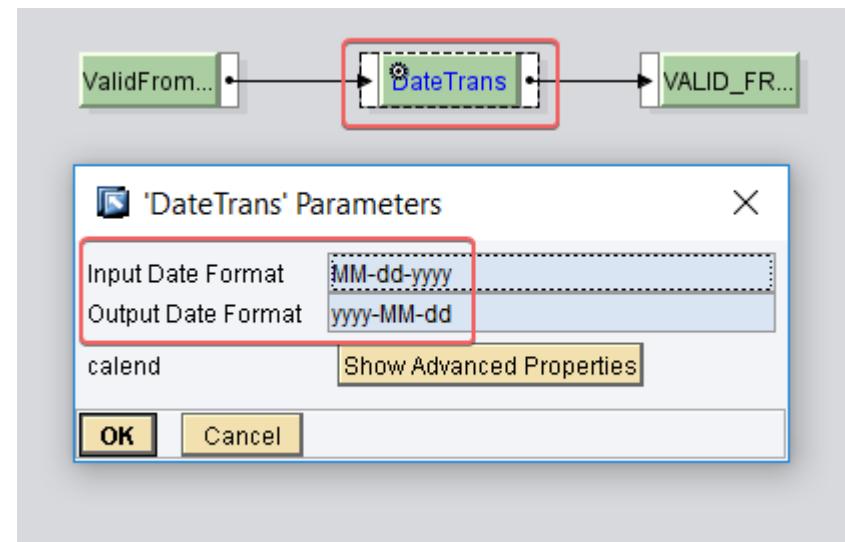
Red arrows indicate the mapping from the left structure's 'Rates' element to the right structure's 'EXCH_RATE' element, and from each of the four child elements ('FromCurrency', 'ToCurrency', 'ExchangeRate', 'ValidFromDate') to their corresponding child elements in the right structure's 'EXCH_RATE' element.

Message Mapping from XML to Proxy message type

Date Transformation using DateTrans Function:



Standard PI mapping function **DateTrans** allows us to convert date formats.



Date transformation using graphical mapping function
DateTrans

Operation Mapping:



Description

Definition Test

Source Operation *				Target Operation *			
Name	Namespace	Software Co...	Occurrence	Name	Namespace	Software Component Version	Occurrence
ExchangeRates_Out_Async	urn:Source_System:ExchangeRates	SOURCE_SYSTEM		ExchangeRates_Inb_Async	urn:Target_System:ExchangeRates	TARGET_SYSTEM, 1.0 of integrationhub.c1	

Read Operations Parameters... Use SAP XML Toolkit Do Not Resolve XOP Includes Read Attachments

Source Message		Mapping Program *		Target Message	
Type	Name	Type	Name	Type	Name
ExchangeRates		Message Mapping	ExchangeRates_to_ExchangeRates		ExchangeRates

Operation Mapping 'ExchangeRates_to_ExchangeRates'.

Step 6 – Generate Proxy Class in SAP Back-end Using Transaction SPROXY:

Now we need to generate Proxy Consumer class in SAP back-end system.

Go to transaction SPROXY in target SAP system (SADCLNT900) and find the Proxy Service Interface from the ESR navigation tree.

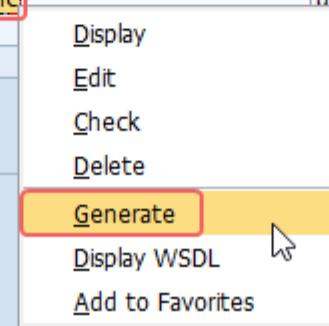




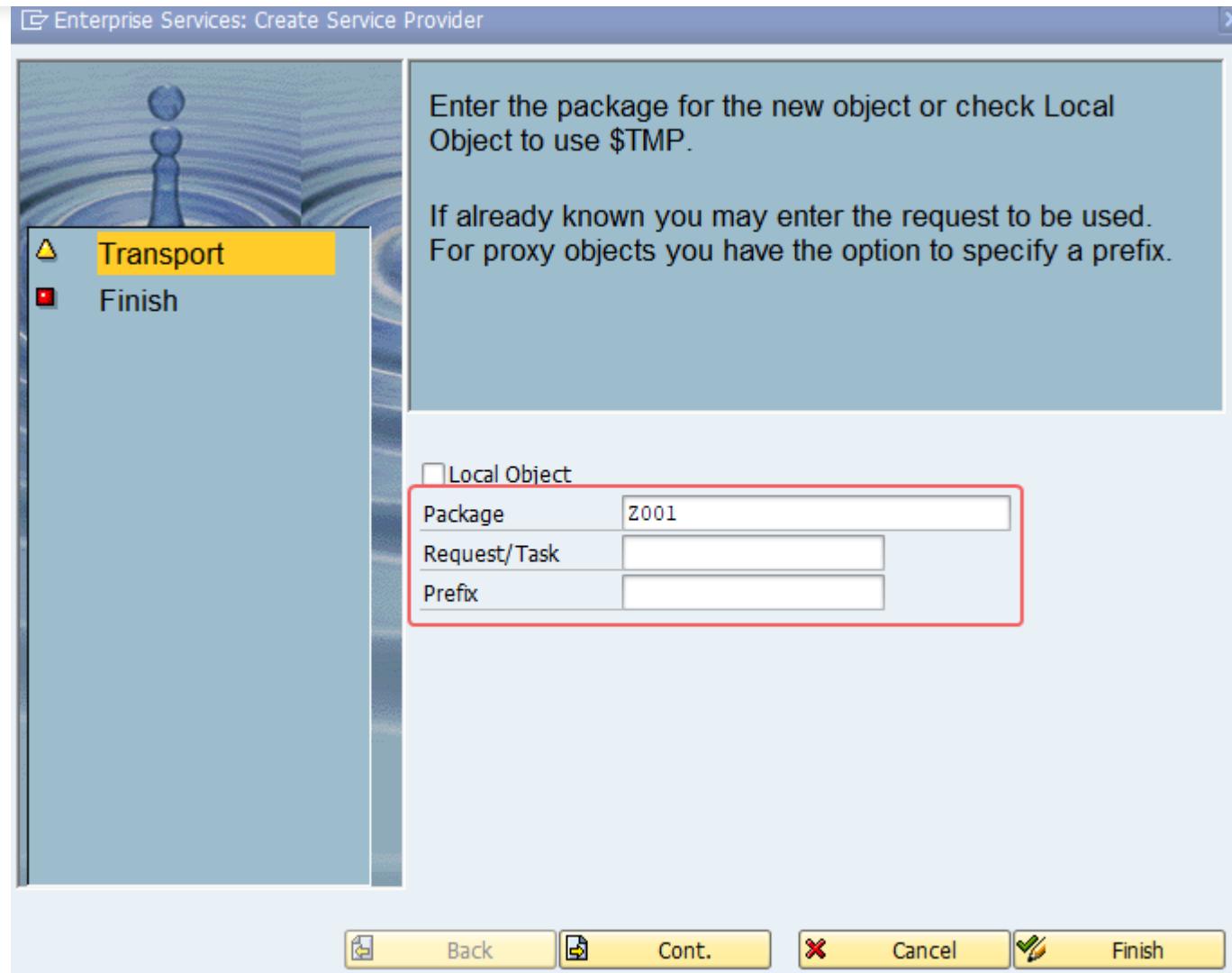
Next, right-click on the Service Interface and click **Generate**. Proxy generation wizard will guide you through the proxy class creation.



Source	
ESR	Enterprise Service Repository
SWCs	
INTERFACEDEMO INTERFACEDEMO, 1.0 of pocasts.com	INTERFACEDEMO, 1.0 of pocasts.com
MAPPING MAPPING, 1.0 of integrationhub.com	MAPPING, 1.0 of integrationhub.com
SENDERSYSTEM SENDERSYSTEM, 1.0 of integrationhub.com	SENDERSYSTEM, 1.0 of integrationhub.com
SOURCE_SYSTEM SOURCE_SYSTEM, 1.0 of integrationhub.com	SOURCE_SYSTEM, 1.0 of integrationhub.com
SWCV_APTOS SWCV_APTOS 1.0 of trg.com	SWCV_APTOS 1.0 of trg.com
SWCV_AVERY SWCV_AVERY 1.0 of averydennison.com	SWCV_AVERY 1.0 of averydennison.com
SWCV_BILLTRUST SWCV_BILLTRUST 1.0 of billtrust.com	SWCV_BILLTRUST 1.0 of billtrust.com
SWCV_MAPPING SWCV_MAPPING, 1.0 of mapping	SWCV_MAPPING, 1.0 of mapping
SWCV_THIRDPARTY_1 SWCV_THIRDPARTY_1, 1.0 of thirdparty-vendor-1	SWCV_THIRDPARTY_1, 1.0 of thirdparty-vendor-1
SWCV_THIRDPARTY_2 SWCV_THIRDPARTY_2, 1.0 of thirdparty-vendor-2	SWCV_THIRDPARTY_2, 1.0 of thirdparty-vendor-2
TARGET_SYSTEM TARGET_SYSTEM, 1.0 of integrationhub.com	TARGET_SYSTEM, 1.0 of integrationhub.com
Namespaces	
urn:Target_System:StudentDetail	
urn:Target_System:ProductList	
urn:TragetSystem:PurchaseOrder	
urn:Target_System:ProductDetail	
urn:Target_System:DecodedString	
urn:Target_System:Base64EncodedString	
urn:Target_System:Base64OutputMessage	
urn:Target_System:ExchangeRates	
Object Types	
Data Types	
Service Providers	
ExchangeRates_Inb_Async	urn:Target_System:ExchangeRates
Message Types	
Objects	
ESR (local)	Enterprise Service Repository (backend view)
External	Created from external WSDL/Schema
Backend	Modelled locally in the backend system
SWCs	
Packages	
Namespaces	
Object Types	
Objects	



Right-click on Service Provider interface and select generate in SPROXY transaction



Proxy Class generation Wizard. Assign the Package, Transport Request and Prefix.

Successfully generated and activated Proxy Service Interface (Proxy ABAP class) will be visible in SPROXY transaction with a green mark as below.



urn:TargetSystem:PurchaseOrder	
urn:Target_System:ProductDetail	
urn:Target_System:DecodedString	
urn:Target_System:Base64EncodedString	
urn:Target_System:Base64OutputMessage	
urn:Target_System:ExchangeRates	
Object Types	
Data Types	
Service Providers	
ExchangeRates_Inb_Async	urn:Target_System:ExchangeRates
Message Types	
Objects	

Service Provider proxy object in SPROXY transaction

Using Proxy Services external view tab, we can check the Proxy Inbound data type. Notice, this correlates to the Receiver Message Type we created in Step 3. Message Types and Data Types we created in ESR are automatically generated as DDIC structures in SAP when the Proxy class is created using SPROXY transaction.



Operation	
Name	ExchangeRates_Inb_Async
ABAP Object	METH Method
ABAP Name	EXCHANGE_RATES_INB_ASYNC
Description	
Pattern	Normal

The screenshot shows the SAP Integration Hub interface. On the left, the ABAP Name tree is displayed under 'ZII_EXCHANGE_RATES_INB_ASYNC'. A red box highlights the 'EXCH_RATE' node under 'INPUT'. On the right, an 'Operation' configuration panel is shown with the following details:

- Name: ExchangeRates_Inb_Async
- ABAP Object: METH Method
- ABAP Name: EXCHANGE_RATES_INB_ASYNC
- Description: (empty)
- Pattern: Normal

In this example, the Proxy Implementing Class name is ZCL_EXCHANGE_RATES_INB_ASYNC. It has one method, which is ZII_EXCHANGE_RATES_INB_ASYNC~EXCHANGE_RATES_INB_ASYNC. ABAP logic to execute the exchange rate creation BAPI should be implemented here.



SAP Integration Hub

ExchangeRates_Inb_Async Active

Properties External View Internal View Objects Configuration WSDL Classifications

Service Provider

Name	ExchangeRates_Inb_Async
Namespace	urn:Target_System:ExchangeRates
ABAP Object	INTF Interface
ABAP Name	ZII_EXCHANGE_RATES_INB_ASYNC
Prefix	
Source	Enterprise Services Repository
Description	Proxy Interface (generated)
Implementing Class	ZCL_EXCHANGE_RATES_INB_ASYNC
WebService Definition	ZExchangeRates_Inb_Async

General Data

Package	2001	Release Status	Not Released
Original Language	EN English		
Created by	ISURUF	on	28.04.2019 16:07:43
Changed by	ISURUF	on	28.04.2019 16:08:13

Double Click

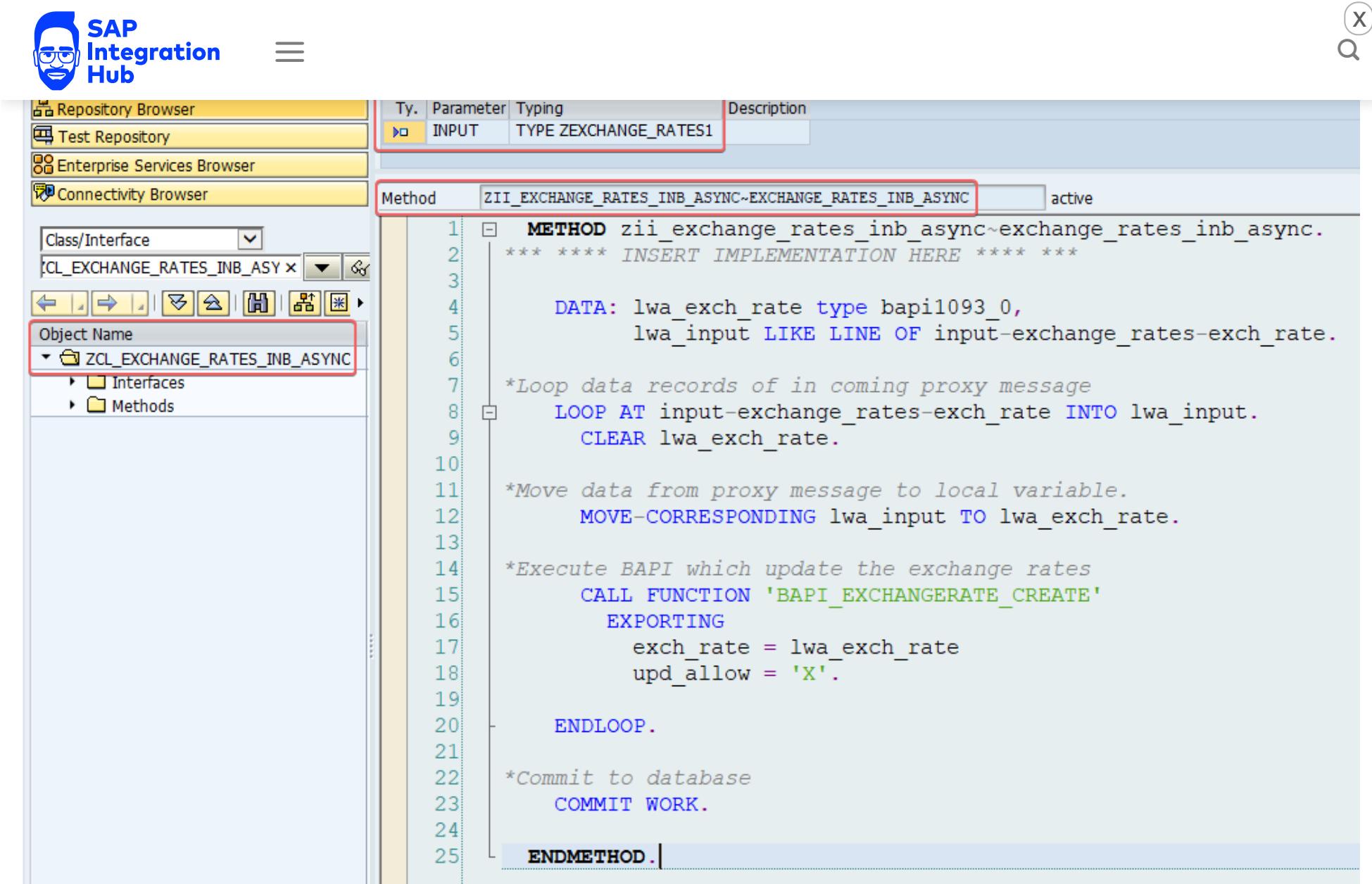


Finally, implement ABAP logic in Proxy class to update the exchange rates in SAP. ABAP logic reads the incoming proxy message (parameter INPUT) and assigns the values to import parameter EXCH_RATE of FM **BAPI_EXCHANGERATE_CREATE**.

Double-click on the Implementing Proxy class in transaction SPROXY and it will take you to the method EXCHANGE_RATES_INB_ASYNC in which you need to implement ABAP code. Or you can implement ABAP code via transaction se24.

Next, implement ABAP program logic in Proxy class which updates exchange rates in SAP.

```
1 METHOD zii_exchange_rates_inb_async~exchange_rates_inb_async.
2 *** **** INSERT IMPLEMENTATION HERE **** ***
3
4 DATA: lwa_exch_rate type bapi1093_0,
5       lwa_input LIKE LINE OF input-exchange_rates-exch_rate.
6
7 *Loop data records of in coming proxy message
8 LOOP AT input-exchange_rates-exch_rate INTO lwa_input.
9   CLEAR lwa_exch_rate.
10
11 *Move data from proxy message to local variable.
12 MOVE-CORRESPONDING lwa_input TO lwa_exch_rate.
13
14 *Execute BAPI which update the exchange rates
15 CALL FUNCTION 'BAPI_EXCHANGERATE_CREATE'
16   EXPORTING
17     exch_rate = lwa_exch_rate
18     upd_allow = 'X'.
19
20 ENDLOOP.
21
22 *Commit to database
23 COMMIT WORK.
24
25 ENDMETHOD.
```



The screenshot shows the SAP Integration Hub interface. On the left, there's a navigation bar with links to Repository Browser, Test Repository, Enterprise Services Browser, and Connectivity Browser. The main area is a code editor with the following details:

- Object Name:** ZCL_EXCHANGE_RATES_INB_ASYNC
- Method:** ZII_EXCHANGE_RATES_INB_ASYNC~EXCHANGE_RATES_INB_ASYNC
- Description:** active
- Type:** INPUT
- Typing:** TYPE ZEXCHANGE_RATES1

The code itself is an ABAP method implementation:

```

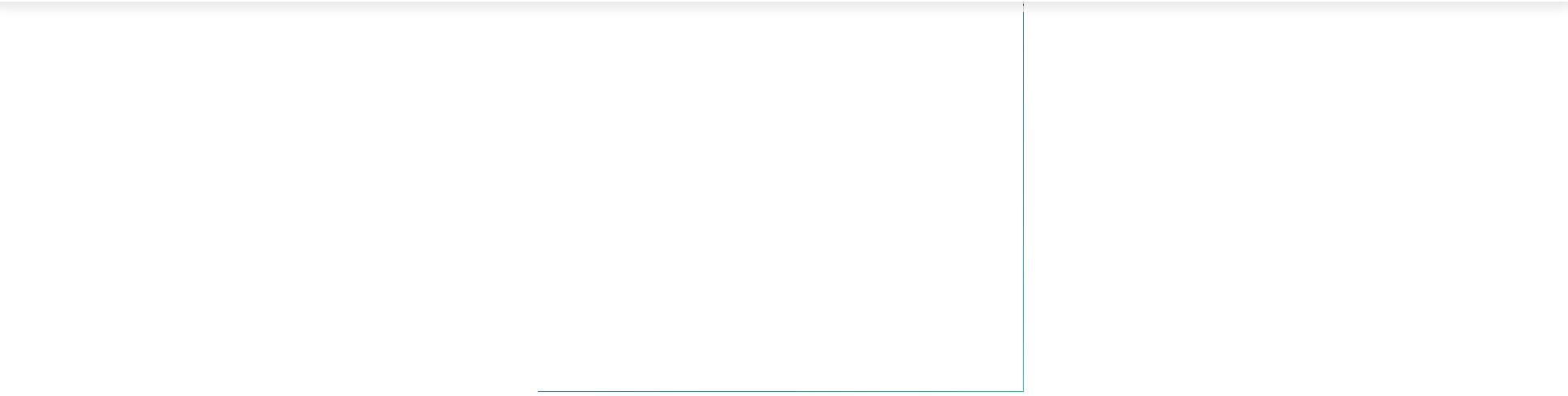
1 METHOD zii_exchange_rates_inb_async~exchange_rates_inb_async.
2   *** ***** INSERT IMPLEMENTATION HERE **** * ***
3
4   DATA: lwa_exch_rate type bapi1093_0,
5         lwa_input LIKE LINE OF input-exchange_rates-exch_rate.
6
7   *Loop data records of in coming proxy message
8   LOOP AT input-exchange_rates-exch_rate INTO lwa_input.
9     CLEAR lwa_exch_rate.
10
11  *Move data from proxy message to local variable.
12  MOVE-CORRESPONDING lwa_input TO lwa_exch_rate.
13
14  *Execute BAPI which update the exchange rates
15  CALL FUNCTION 'BAPI_EXCHANGERATE_CREATE'
16    EXPORTING
17      exch_rate = lwa_exch_rate
18      upd_allow = 'X'.
19
20  ENDLOOP.
21
22  *Commit to database
23  COMMIT WORK.
24
25 ENDMETHOD.

```

Proxy Class Signature and ABAP Code to update Exchange Rates in SAP

Step 7 – Create sFTP Sender Communication Channel and Proxy (SOAP) Receiver Channel:

Let's create sender and receiver Communication Channels.



Proxy Receiver Communication Channel:

Similarly, we need a Communication Channel for Proxy Receiver. Adapter type of Proxy receiver Channel is SOAP and Transport Protocol is HTTP with Message Protocol XI 3.0.

SOAP Adapter General Configuration:



Enter a name and description for the channel

Party:

Communication Component: **SADCLNT900**

Channel Name: **SOAP_Receiver**

Direction: **Receiver**

Adapter Engine: **Central Adapter Engine (af.po1.po1-att-aws)**

Description:

Choose an adapter type and its protocols

Adapter Type: **SOAP** **http://sap.com/x** **SAP BASIS 7.50** **Browse...**

Transport Protocol: **HTTP**

Message Protocol: **XI 3.0**

SOAP Receiver General Configuration

Receiver Proxy Channel should be created under [SAP back-end system Business System](#).

SOAP Adapter Specific Configuration:

HTTP destination pointed at SAP back-end system is created in Netweaver Development Administration (NWA). The destination should be assigned to the Communication Channel.



General Advanced

Security Parameters

Select security profile

Connection Parameters

Addressing mode:* HTTP destination

HTTP destination:* SADCLNT900_HTTP

XI Packaging

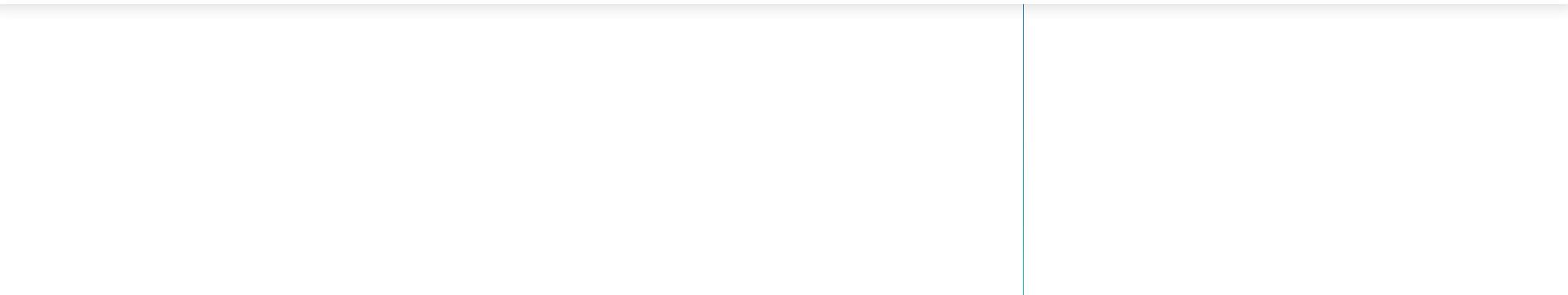
Use Packaging

Pass message hops

SOAP Receiver Communication Channel

Step 8 – Generate the iFlow Using NWDS:

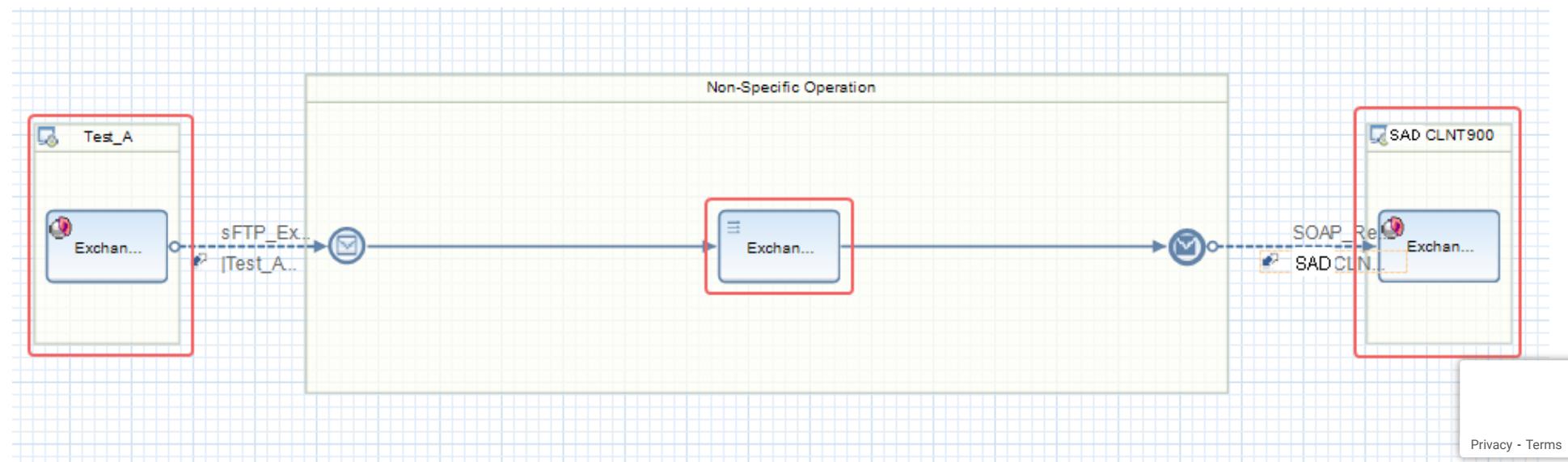
Last step of the interface development configuration is to setup the iFlow.



Exchange rate sender system is Test A and receiver system is SADCLNT900. The exchange rates mapping program we created in Step 5 is ExchangeRates_to_ExchangeRates.

Receiver Proxy SOAP Communication Channel is SOAP_Reliever. We created the Channels in Step 7.

Using these objects, [define the iFlow using Eclipse NWDS](#). You can also complete the Integration Directory configuration using the Integrated Configuration Object (ICO).





TEST THE INBOUND PROXY INTERFACE.

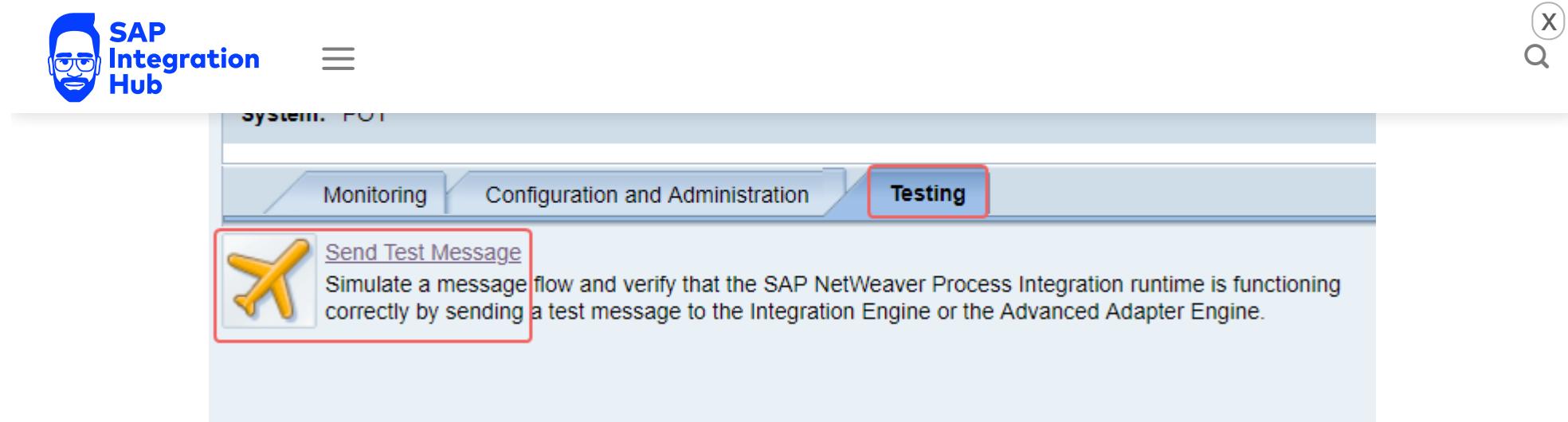
Here we will test the complete interface end to end.

Test Interface using PI Test Tool:

First, let's test the interface using SAP PI AFW Test Tool. This will help us to find out if there are any issues in the Message Mapping program or Proxy ABAP logic.

Send a Test Message using PI Test Tool:

Go to Test Tool from the Monitoring home.



The screenshot shows the SAP Integration Hub interface. At the top left is the SAP Integration Hub logo. To its right are three horizontal bars. On the far right are a search icon (magnifying glass) and a close/cancel icon (X). Below the header is a navigation bar with tabs: 'Monitoring', 'Configuration and Administration', and 'Testing'. The 'Testing' tab is highlighted with a red border. Below the tabs is a section titled 'Send Test Message' featuring a yellow airplane icon. A callout box with a red border contains the text: 'Simulate a message flow and verify that the SAP NetWeaver Process Integration runtime is functioning correctly by sending a test message to the Integration Engine or the Advanced Adapter Engine.' The entire screenshot is framed by a thick black border.

Select the Integrated Configuration Object (ICO) and choose the Quality of Service. Next, paste the XML payload content and click 'Send' button to process the message.

The screenshot shows the SAP Integration Hub interface for configuring an inbound proxy scenario. The top navigation bar includes the SAP Integration Hub logo, a search bar, and standard navigation icons.

Header Information:

- Sender Party: Test_A
- Sender Component: Test_A
- Receiver Party: (empty)
- Receiver Component: (empty)
- Interface: ExchangeRates_Out_Async
- Interface Namespace: urn:Source_System:ExchangeRates
- Quality of Service: Exactly Once

Payload:

Enter Payload

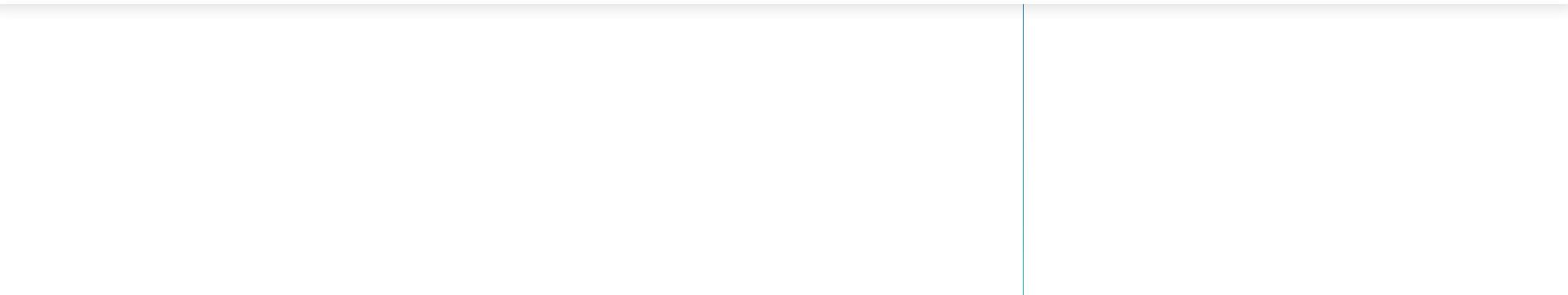
```
<?xml version="1.0" encoding="UTF-8"?>
<ns0:ExchangeRates xmlns:ns0="urn:Source_System:ExchangeRates">
  <Rates>
    <FromCurrency>USD</FromCurrency>
    <ToCurrency>EUR</ToCurrency>
    <ExchangeRate>0.89</ExchangeRate>
    <ValidFromDate>05-01-2019</ValidFromDate>
  </Rates>
  <Rates>
    <FromCurrency>EUR</FromCurrency>
```

Upload File

Key UI elements highlighted with red boxes and green numbered callouts:

- 1: A red box surrounds the "Prefill Fields Based On:" section, which contains the Sender Agreement or Integrated Configuration (ICo) field: "Test_A|ExchangeRates_Out_Async|urn:Source_System:ExchangeRates".
- 2: A red box surrounds the Quality of Service dropdown set to "Exactly Once".
- 3: A red box surrounds the XML payload code.
- 4: A green circle with the number 4 is located above the "Send" button in the toolbar.

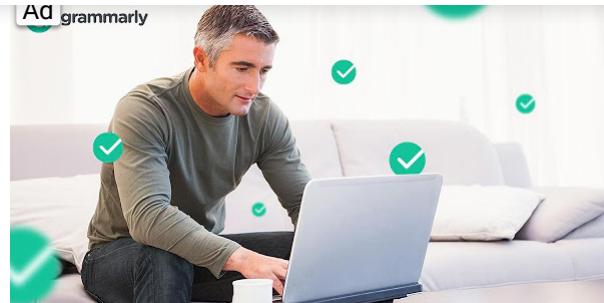
Note down the message ID. It's easy to find the proxy message in SAP using message ID.



Monitor the Interface from PI to SAP:

Monitor the message via PI message monitor.

A screenshot of the SAP PI message monitor interface. The top part is a table with columns: Status, Status Details, Start Time, End Time, Integration Scenario, Sender Party, and Sender Component. A single row is highlighted in yellow, showing 'Delivered' status, start time '5/1/2019 07:17:52.230 AM', end time '5/1/2019 07:17:52.583 AM', scenario 'ExchangeRates', sender party 'Test_A', and component 'Test_A'. Below the table is a section titled 'Message Details' with tabs for 'Message Log', 'Further Links', and 'Contact Information'. The 'Message Log' tab is selected, displaying a table of log entries with columns: Time, Status, and Description. The log entries show the process of sending a message via connection AFW, including putting it into the queue and retrieving it successfully.



Correct All Grammar Errors

Trusted by millions of students, faculty, and professionals worldwide. Try now

 Grammarly

[LEARN MORE](#)





Status

Standard Selection Criteria Advanced Selection Criteria User-Defined Selection Criteria

Message ID

Message Type	<input type="text"/>
From Date/Time Sent	<input type="text"/> / <input type="text" value="00:00:00"/>
To Date/Time Sent	<input type="text"/> / <input type="text" value="00:00:00"/>
From Execution Date/Time	<input type="text"/> / <input type="text" value="00:00:00"/>
To Execution Date/Time	<input type="text"/> / <input type="text" value="00:00:00"/> <input type="button" value="..."/>
Logical Pipeline ID	<input type="text"/>
Error ID	<input type="text"/>
Status Details	<input type="text"/>
Client	<input type="text"/>
Technical Inbound Channel	<input type="text"/>
Outbound Status	<input type="text" value="Initial"/>
Queue ID	<input type="text"/>
With Acknowledgment Messages	<input type="checkbox"/>

Proxy XML message content can be monitored in SXMB_MONI message viewer.

The screenshot shows the SAP Integration Hub interface. On the left, there's a sidebar with a user icon, the text "SAP Integration Hub", and a menu icon. Below that is a tree view with "Payloads" expanded, showing "MainDocument (application/xml)". Under "Response", there's a file icon. At the top right are search and X icons.

The main area displays two XML documents side-by-side. The first document is a SOAP header:

```
xmlns:SAP="http://sap.com/xi/XI/Message/30">
- <SOAP:Header>
  - <SAP:Main versionMajor="003" versionMinor="001" SOAP:mustUnderstand="1"
    wsu:Id="wsuid-main-92ABE13F5C59AB7FE10000000A1551F7">
```

The second document is an "ExchangeRates" message:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <ns1:ExchangeRates xmlns:ns1="urn:Target_System:ExchangeRates">
  - <EXCH_RATE>
    <RATE_TYPE>M</RATE_TYPE>
    <FROM_CURR>USD</FROM_CURR>
    <TO_CURRENCY>EUR</TO_CURRENCY>
    <VALID_FROM>2019-05-01</VALID_FROM>
    <EXCH_RATE>0.89</EXCH_RATE>
    <FROM_FACTOR>1</FROM_FACTOR>
    <TO_FACTOR>1</TO_FACTOR>
    <EXCH_RATE_V>0.0</EXCH_RATE_V>
    <FROM_FACTOR_V>0</FROM_FACTOR_V>
    <TO_FACTOR_V>0</TO_FACTOR_V>
  </EXCH_RATE>
  - <EXCH_RATE>
    <RATE_TYPE>M</RATE_TYPE>
    <FROM_CURR>EUR</FROM_CURR>
    <TO_CURRENCY>AUD</TO_CURRENCY>
    <VALID_FROM>2019-05-01</VALID_FROM>
    <EXCH_RATE>1.59</EXCH_RATE>
    <FROM_FACTOR>1</FROM_FACTOR>
    <TO_FACTOR>1</TO_FACTOR>
    <EXCH_RATE_V>0.0</EXCH_RATE_V>
    <FROM_FACTOR_V>0</FROM_FACTOR_V>
    <TO_FACTOR_V>0</TO_FACTOR_V>
  </EXCH_RATE>
</ns1:ExchangeRates>
```

Go to transaction se16n to check if exchange rates are updated as expected.

Privacy - Terms



Search in Table Exchange Rates

Number of hits

Runtime Maximum no. of hits

Details

ExRt	From	To	Valid from	Exch. Rate	Ratio	Ratio
M	EUR	AUD	01.05.2019	1,59000	0	0
M	USD	EUR	01.05.2019	0,89000	0	0

Use either SPROXY session debugging or [external debugging](#) if the behavior of the Proxy ABAP logic needs to be analyzed at runtime.

If you have any questions on the steps illustrated or Proxy Inbound interfaces, please leave a comment below!



This entry was posted in [Proxy](#) and tagged [ABAP](#), [Asynchronous](#), [BAPI_EXCHANGERATE_CREATE](#), [Data Type](#), [DateTrans](#), [File Adapter](#), [File to Proxy](#), [HTTP](#), [iFlow](#), [Message Type](#), [NWDS](#), [Proxy](#), [Service Interface](#), [SFTP](#), [SFTP Adapter](#), [SOAP](#), [SOAP Adapter](#), [sproxy](#), [TCURR](#), [Test Message](#), [Test Tool](#), [XML](#).



tech-related. I started my career as an ABAP developer and soon found my love for system integration when I learned SAP XI 3.0 in 2008. Playing a variety of roles from an offshore technical consultant (ABAP, PI/PO, BW, BOBJ) at the beginning of my career to a technical lead managing a team of consultants in different countries, I have gained immense experience in SAP project implementation life cycle. Having the opportunity to work on SAP implementation projects in USA, EU, and Asia, I learned valuable ins and outs of global business processors in Sales and Distribution (SD), Material Management, Retail, Customer Relationship Management (CRM), and Finance and Controlling (FICO). Through this blog, I want to share my expertise in SAP technical areas such as SAP ABAP, PI/PO, AIF, and Basis. I also want to provide a platform for others with similar ambitions who would like to share their SAP technical expertise with the world!

[Base64 Encoding Java Mapping Example for SAP PI/PO](#)

[Use of UseOneAsMany Node Function with Examples](#)

SIGN UP TODAY!

Sign up to receive our monthly newsletter and special deals!

Your Name (Required)

Your Email (Required)



I accept Newsletters & Blog updates.

SIGN UP



[Privacy - Terms](#)



BECOME AN AUTHOR

Sign up as a contributing author to write your own articles!

[REGISTER](#)

TECH GADGETS I USE EVERY DAY

These are some of the tech gadgets I use every day. If you make a purchase through these links I will earn a small comission at absolutely no extra cost to you.



Bose
QuietComfort...
\$299.00 Prime

[Shop now](#)



2019 Apple
MacBook Pro...
\$2,285.00 Prime

[Shop now](#)



(Renewed)
Apple iPhone
Pro
\$630.00

[Shop now](#)





Nikon D3500
24.2MP DSL

\$799.99

Shop now



Bose
SoundSport.

\$239.97

Shop now

2 THOUGHTS ON “FILE TO INBOUND PROXY SCENARIO SAP PI/PO SINGLE STACK”

Chaouki says:

Bonjour,

I have a question about the Proxy Implementing Class ZCL_EXCHANGE_RATES_INB_ASYNC.

For example, if the BAPI returns an error message BAPI_EXCHANGERATE_CREATE, then how do catch it and save it ? So we can see it later in the SXMB_MONI transaction for example

BR,

Chaouki

NOVEMBER 3, 2019 AT 8:38 AM

REPLY

Isuru Fernando says:

Privacy - Terms



in my new article about fault messages in SAP PI/PO. Additionally, you can also save BAPI return messages to [SAP application log](#).

Cheers!

Isuru

NOVEMBER 3, 2019 AT 3:31 PM

[REPLY](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

[Privacy](#) - [Terms](#)



POST COMMENT

 ezoic

report this ad

REGRESSION TEST AUTOMATION





SAP
Integration
Hub

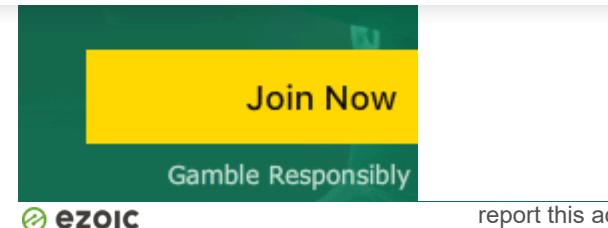
migration
testing

TRY FOR FREE

Use promo code
"SAPIntegrationHub"
to extend the free trial period
of Int4 IFTT by one week

TAG CLOUD





RECENT COMMENTS

Jakku on [Change iDoc Status by Standard Program](#)

pavani on [XML Transformation Example with XSLT_TOOL](#)

Jamaic on [iDoc Collection and Package Size – S4 HANA and PI/PO Configuration](#)

Nila on [Outbound IDoc Configuration with Output Determination in SAP – Techno-functional Guide](#)

Rituparna Roy on [Proxy Outbound Interface Example SAP to PI File Receiver](#)

LATEST POSTS

09 Jul [Integrating VIES VAT Validation with SAP S4 HANA](#)

06 Jun [Electronic Bank Statement Integration \(SAP S4 HANA\)](#)

20 May [SAP S4 HANA Application Log \(User Manual\)](#)

[Privacy](#) - [Terms](#)

06
Mar

Ultimate Guide to SAP PI/PO to CPI Migration

CATEGORIES

[ABAP](#)[AIF](#)[BASIS](#)[CPI](#)[How To](#)[Integration Architecture](#)[Integration Scenarios](#)[PI/PO](#)



SITE LINKS

[Home](#)

[Blog](#)

[Write for Us](#)

[Disclaimer](#)

[Privacy Policy](#)

[Contributing Author Registration](#)

SIGN UP TODAY!

Sign up to receive our monthly newsletter and special deals!

Your Name (Required)

Your Email (Required)



I accept Newsletters & Blog updates.

[SIGN UP](#)



[Privacy - Terms](#)



Copyrights 2020 © SAP Integration Hub