

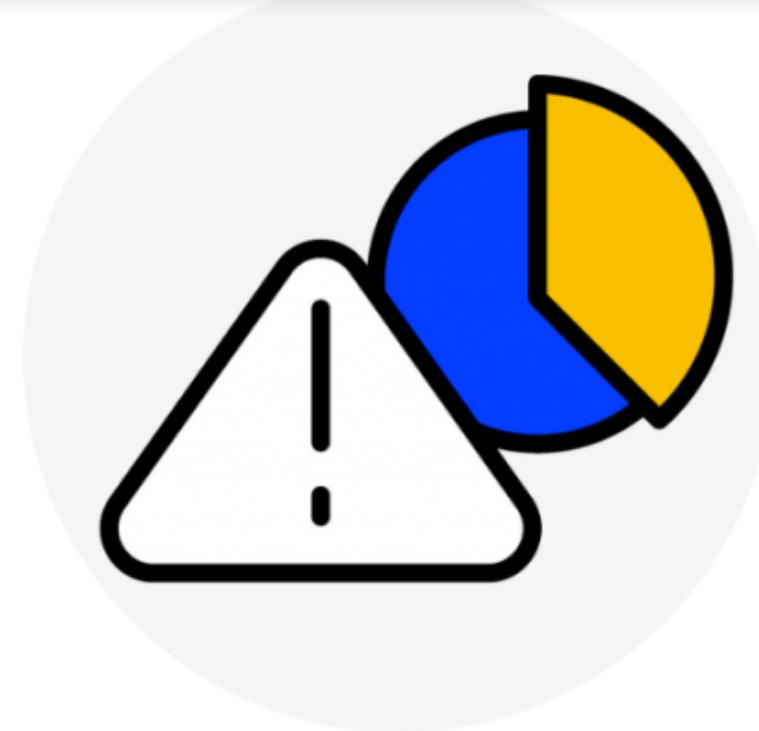


FAULT MESSAGES

Handling Fault Messages in Asynchronous ABAP Proxy

POSTED ON JUNE 21, 2019 BY ISURU FERNANDO





What is a Fault Message? Fault Messages provide information about application-specific errors that occur in the inbound system. Think about Fault Messages as an acknowledgment from the message receiver to the message sender or the message monitor about the application processing status.

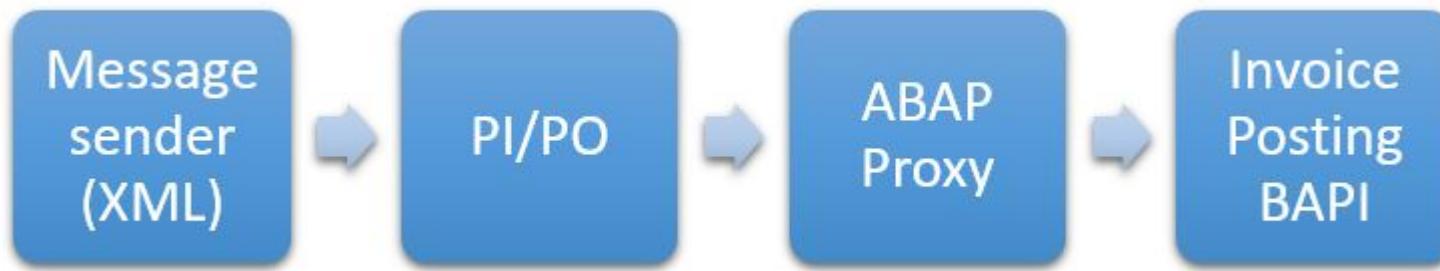


SAP VERSIONS USED IN THE ILLUSTRATION.

- SAP S4 HANA Fashion 1709

Use Cases of Fault Messages in ABAP Proxy

Use case 1: Capturing BAPI/FM Return



Let's assume a sender system sends a message with invoice information to target system SAP. The invoice is posted from the inbound ABAP proxy class using a BAPI.

What will happen if all information needed to post an invoice in SAP is not sent in the message? Obviously, invoice won't be posted in the target SAP system. But how does a user identify the issue? What are the return messages from BAPI? If we do not capture the BAPI return messages and pass it back to the message monitor, messages will always show as successful in SXMB_MONI as technically message was processed successfully. (!!!message message message success successful)

processing to message monitor. Then, the user will be able to identify application processing (Invoice posting) errors and take necessary actions to correct the issue and reprocess the messages.



Use Case 2: Message Validations

Let's assume you need to validate the inbound message content based on business rules before processing the message in SAP. How do we capture the status of the validation in message monitor?

If validation failures are not passed to SXMB_MONI as an exception, the message will always show as successfully processed. In this case, business users won't be able to identify which messages actually posted successfully in SAP and which messages did not pass the validation.

Therefore, you need to capture error messages of the validation and raise them as an exception.

For example, take the [exchange rates interface](#) we built previously. This interface updates exchange rates in SAP via a BAPI implemented in the proxy class. Now we have a requirement to validate the message content before calling the BAPI. “Valid from date” in the exchange rates input message should not be in the past.

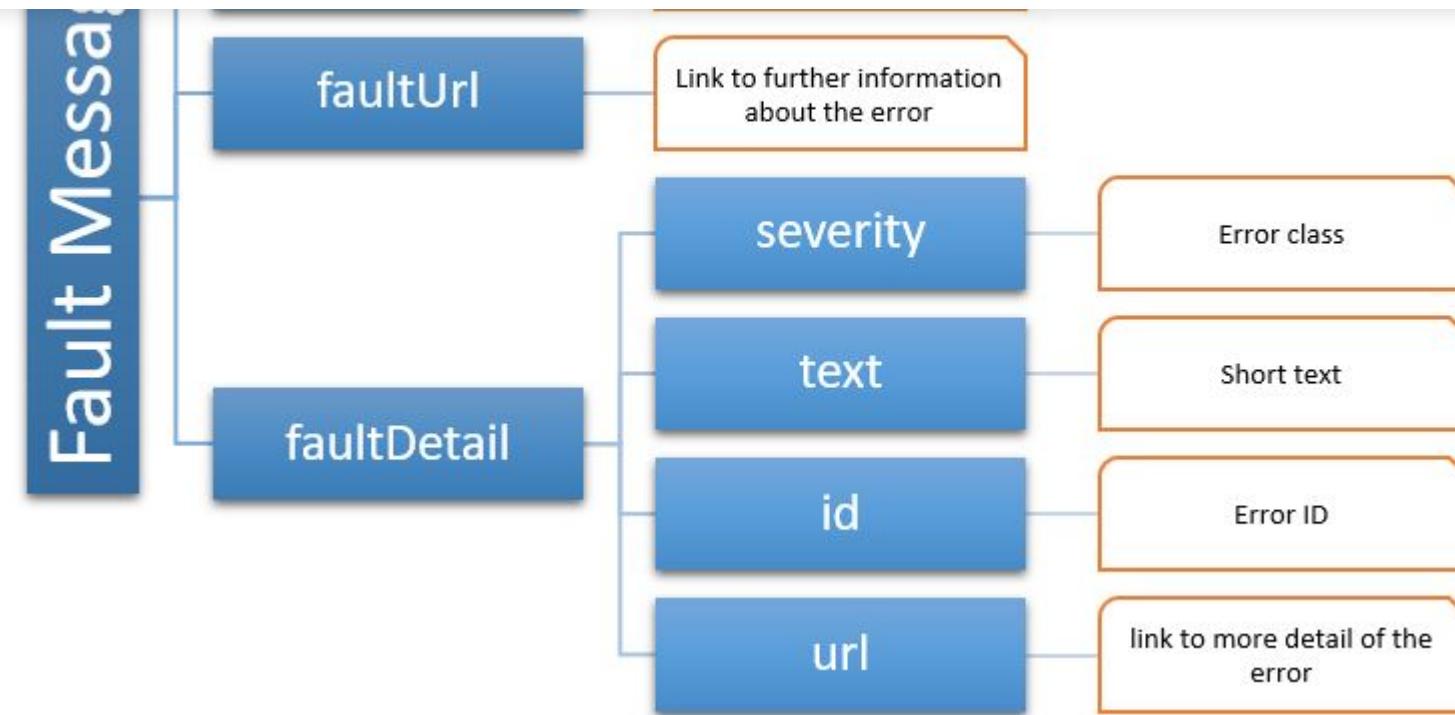
Structure or Format of Fault Messages in SAP PI and ABAP Proxy

Standard Fault Messages in SAP PI/PO have a predefined format. But of course you can enhance this format to add additional fields and segments if necessary.

Standard Structure of Fault Messages in SAP PI

Standard Fault Message Type has 3 main segments. They are two fields named **faultText** and **faultUrl** and a table named **faultDetail**.





Standard Fault Message Structure

faultText:

This parameter provides a short description of the application error. It provides an overall status of message processing. For example, for use case 1, we can define messages as "Invoice not posted".

In use case 2, we can define them as "Exchange rates not updated in SAP"



faultDetail:

This is a table parameter which can hold multiple error/warning/information messages from the application processing. For example, BAPI can return multiple messages about invoice processing or application processing. We can include all those messages to the proxy run-time monitor so the user can find all the details of the message processing.

Structure of the faultDetail includes four fields:

- severity
- text



severity



text

Message text.

url

Url to the long text of the message in '**text**'.

id

You can assign the message ID here. For example, message class and message number of the error in SAP. Assuming our message class is SD and message ID is 002, you can set the ID as SD(002).

This correlates to how the message class and message numbers are defined in transaction se91.

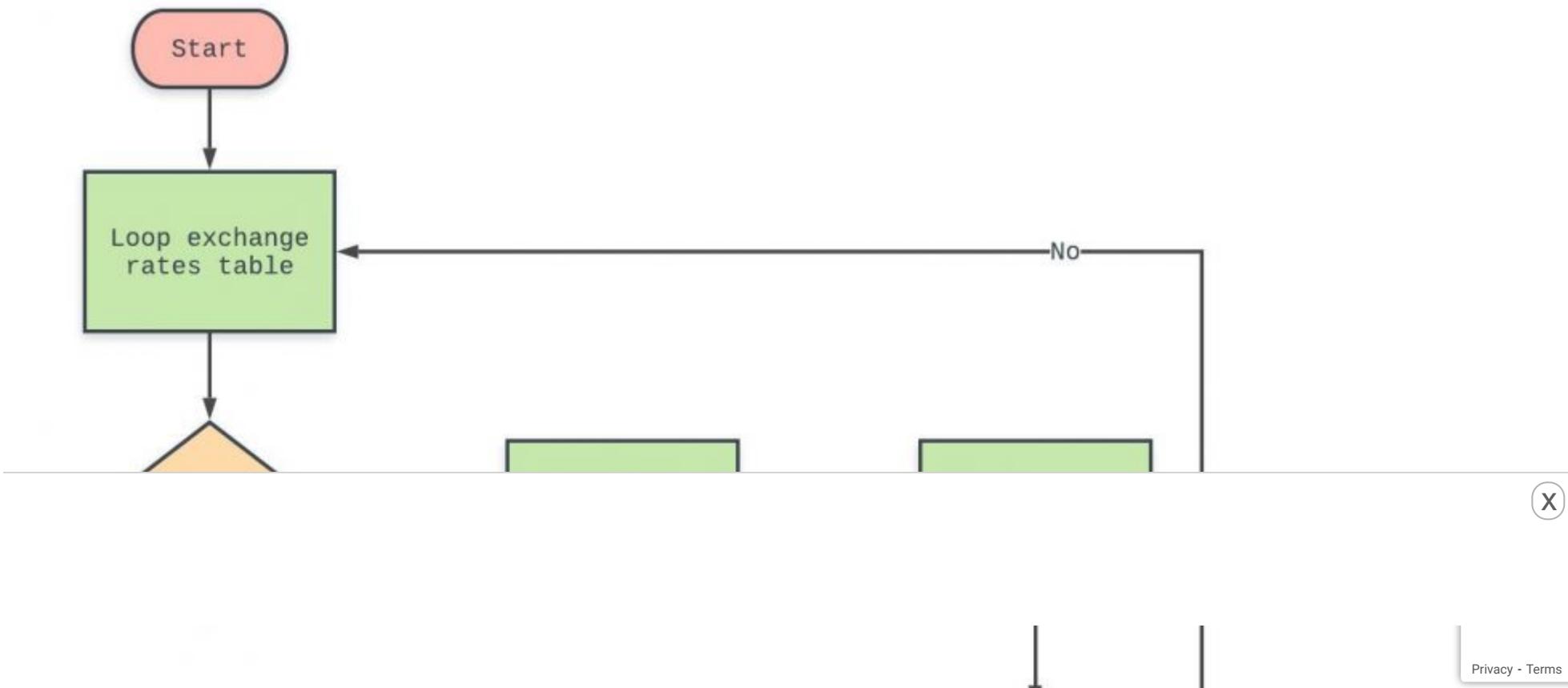


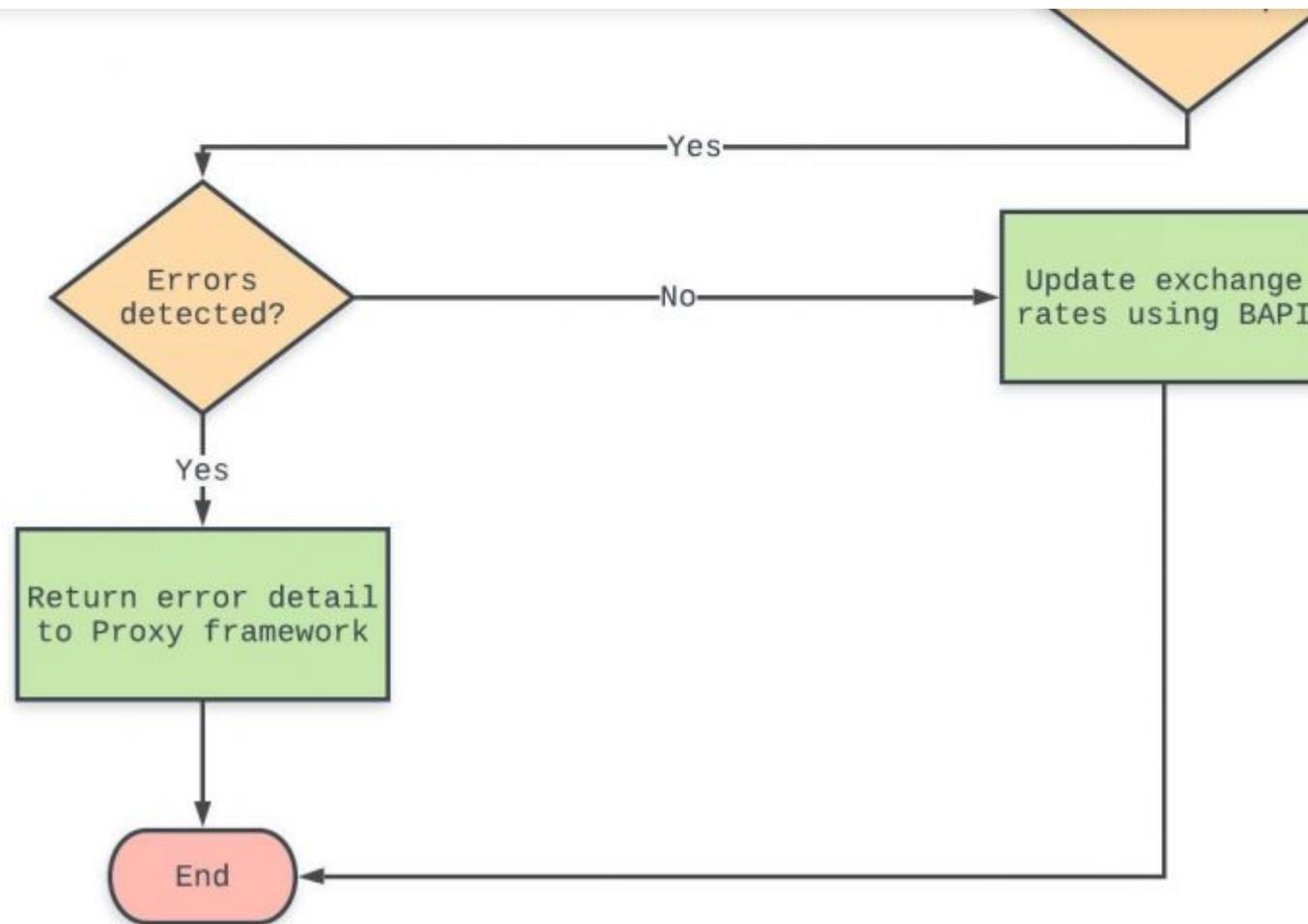
| *Have a look at the implementation of the exchange rates interface before proceeding.*

We will modify the exchange rates interface to return application processing errors to SXMB_MONI. Let's assume if the "Valid from" date (VALID_FROM) of the exchange rates record in the file is in the past, we do not want to update exchange rates in SAP. Records which do not satisfy the "Valid from" date validation should be visible in SXMB_MONI as error messages.

Pseudo Code of the Validation Logic in Proxy Class

"Valid from date" validation should be implemented in the Proxy class. Here's how the processing logic looks like.





Proxy class ABAP logic pseudo code



Now we will look at all the steps of Fault messages implementation in our second interface. The differences in steps between

[Asynchronous Proxy interface](#) we previously implemented without Fault Message and this interface with Fault Messages are,

Privacy - Terms



- Creation of a new Operation Mapping.

Full list of Fault Message implementation steps are as follows,

1. Configure [Proxy connectivity](#) between SAP PI/PO and SAP back-end system.
2. Create standard Fault Message's Data Type and Message Type.
3. Create sender Data Type and Message Type.
4. Create receiver Proxy Data Type and Message Type.
5. Define the Inbound Service Interface with Fault Message.
6. Implement Message Mapping and Operation Mapping.
7. Generate Proxy Class in SAP back-end system.
8. Implement ABAP logic to return application errors to SXMB_MONI.
9. Configure the [iFlow](#).

As all the steps, except for step 2, step 5, and step 8, are the same for Fault Message implementation as the original [exchange rates interface](#), I will demonstrate only the detail of steps 2, 5 and 8 in this article. You can refer to the previous post to check more information about other steps.

Step 2: Create Fault Message

First, create two data types for FaultData and Log Data. You can copy any standard fault message data types to your custom SXMB_MONI.





Software Component Version TARGET_SYSTEM, 1.0 of integrationhub.com

Description

Classification Free-Style Data Type Quality Schema None

Type Definition XSD

Name	Category	Type	Occurrence
▼ ExchangeLogData	Complex Type		
severity	Element	xsd:string	0..1
text	Element	xsd:string	1
url	Element	xsd:string	0..1
id	Element	xsd:string	0..1

ExchangeLogData Data Type

Data Type ExchangeFaultData





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
ExchangeFaultData	Complex Type		
faultText	Element	xsd:string	1
faultUrl	Element	xsd:string	0..1
faultDetail	Element	ExchangeLogData	0..unbounded
severity	Element	xsd:string	0..1
text	Element	xsd:string	1
url	Element	xsd:string	0..1
id	Element	xsd:string	0..1

ExchangeFaultData Data Type

Create Fault Message Type

Create a Fault Message Type by assigning the Fault Data Type **ExchangeFaultData** created previously.

To create the Fault Message Type, first, right-click on the namespace and select "Fault Message Type" from the list of objects.





A screenshot of the SAP Integration Hub interface. On the left, there is a sidebar with various options: Service Interface, Message Type, Fault Message Type (which is highlighted with a red box), Data Type, Data Type Enhancement, External Definition, Context Object, Mapping Objects, Adapter Objects, Version Creation, and Work Areas. The main area has two fields: 'Namespace' and 'Software Component Version *'. A large red box highlights the 'Fault Message Type' option in the sidebar.

Create Fault Message Type in ESR

Then assign the Data Type ExchangeFaultData we created in step 2 to the Fault Message Type.





Software Component Version TARGET_SYSTEM, 1.0 of integrationhub.com

Description

Data Type Used

Name *	Namespace *
Standard Data ExchangeFaultData	urn:Target_System:ExchangeRates

Additional Data

XML Namespace urn:Target_System:ExchangeRates

Structure XSD

Search Go

Name	Category	Type	Occurrence	Default
FaultMessage	Element			
standard	Element	ExchangeFaultData	1	
faultText	Element	xsd:string	1	
faultUrl	Element	xsd:string	0..1	
faultDetail	Element	ExchangeLogData	0..unbounded	
severity	Element	xsd:string	0..1	
text	Element	xsd:string	1	
url	Element	xsd:string	0..1	
id	Element	xsd:string	0..1	

Fault Message structure in ESR

Step 5: Define the Inbound Service Interface with Fault Message



The screenshot shows the SAP Integration Hub interface for defining a service interface. 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 (selected), WSDL, Matching Service Interfaces, Classifications

Attributes

- Category: Inbound
- Interface Pattern: Stateless (XI30-Compatible) (highlighted with a red box)
- Point-to-Point enabled
- Event interface
- Sensitive Data

Operations

Operation ExchangeRateswithFaultMessage_Inb_Async

Description: (empty)

Release State: Not Released

Attributes

- Operation Pattern: Normal Operation
- Mode: Asynchronous

Messages

Context Objects: (empty)

Role	Type	Name	Namespace
Request *	Message Type	ExchangeRates	urn:Target_System:ExchangeRates
Fault	Fault Message Type	FaultMessage	urn:Target_System:ExchangeRates

Inbound Service Interface with Fault Message assigned



<ul style="list-style-type: none">▶ urn:TragetSystem:PurchaseOrder▶ urn:Target_System:ProductDetail▶ urn:Target_System:DecodedString▶ urn:Target_System:Base64EncodedString▶ urn:Target_System:Base64OutputMessage▼ urn:Target_System:ExchangeRates<ul style="list-style-type: none">▶ Object Types▼ Objects<ul style="list-style-type: none">• ExchangeFaultData• ExchangeFaultData• ExchangeLogData• ExchangeRates• ExchangeRates• ExchangeRates_Inb_Async• ExchangeRateswithFaultMessage_Inb_Async• FaultMessage	
--	--

Inbound Service Interface and Proxy class in SPROXY Transaction

Go to Proxy implementation class and you will be able to view the exception class **ZCX_FAULT_MESSAGE** in the method signature.





Method ZII_EXCHANGE_RATESWITHFAULT_M~EXCHANGE_RATESWITHFAULT_MESSA active

```

1  METHOD zii_exchange_rateswith_fault_m~exchange_
2    **** * INSERT IMPLEMENTATION HERE **** *
3
4      DATA: lwa_exch_rate  LIKE LINE OF input-excha
5          lwa_bapi_input  TYPE bapi1093_0,
6          lwa_input        LIKE input.
7
8      DATA: lt_return           TYPE TABLE OF bapiret
9          lwa_standard_data  TYPE zexchange_fault
10         lwa_error_detail   TYPE zexchange_log_da
11
12     DATA: lv_error_flag  TYPE c VALUE IS INITIAL.
13

```

Proxy class with zcx_fault_messages exception class

Interface ZII_EXCHANGE_RATESWITHFAULT_M Implemented / Active

Properties Interfaces Attributes Methods Events Types Aliases

Exceptions of Method EXCHANGE_RATESWITHFAULT_MESSA

Methods	Parameters	Properties
Exception	Resumable	Description
ZCX_FAULT_MESSAGE	<input type="checkbox"/>	
	<input type="checkbox"/>	

Proxy class with zcx_fault_messages exception class



Properties						
Attribute	Level	Visibility	R...	Typing	Associated Type	
IF_AI_APPLICATION_FAULT~MT_FAULT_R...	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	TTFAULT_REASON_TYPE	
IF_AI_APPLICATION_FAULT~MT_SUBCODE	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	TT_SUBCODE	
CX_ROOT	Constant	Public	<input type="checkbox"/>	Type	SOTR_CONC	
TEXTID	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	SOTR_CONC	
PREVIOUS	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type Ref To	CX_ROOT	
KERNEL_ERRID	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	S380ERRID	
IS_RESUMABLE	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	ABAP_BOOL	
CX_AI_APPLICATION_FAULT	Constant	Public	<input type="checkbox"/>	Type	SOTR_CONC	
AUTOMATIC_RETRY	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	PRX_AUTOMATIC_RETRY	
CONTROLLER	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	PRXCTRLTAB	
NO_RETRY	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	PRX_NO_RETRY	
STANDARD	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	ZEXCHANGE_FAULT_DATA	
WF_TRIGGERED	Instance Attribute	Public	<input checked="" type="checkbox"/>	Type	PRX_WORKFLOW_TRIGGERED	
			<input type="checkbox"/>	Type		

Fault Message DDIC structure in SAP back-end system under zcx_fault_message class

DDIC structure ZEXCHANGE_FAULT_DATA correlates to FaultMessage Message Type in ESR.



Built-In Type							4
Component	Typing Method	Component Type	Data Type	Length	Deci...	Short Description	
CONTROLLER	Types	PRXCTRLTAB	Table	0	0	Control Flags for Fields of a Structure	
FAULT_TEXT	Types		STRING	0	0		
FAULT_URL	Types		STRING	0	0		
FAULT_DETAIL	Types	ZEXCHANGE_LOG_DATA_TAB	Table	0	0	Proxy Table Type (generated)	

Exchange Fault Message DDIC structure in SAP back-end

Also, DDIC structure ZEXCHANGE_LOG_DATA corresponds to ExchangeLogData Data Type.

Built-In Type							5
Component	Typing Method	Component Type	Data Type	Length	Deci...	Short Description	
CONTROLLER	Types	PRXCTRLTAB	Table	0	0	Control Flags for Fields of a Structure	
SEVERITY	Types		STRING	0	0		
TEXT	Types		STRING	0	0		
URL	Types		STRING	0	0		
ID	Types		STRING	0	0		

FaultLogData Message Type as a DDIC structure in SAP back-end system

Proxy Class ABAP Code

```
1 METHOD zii_exchange_rateswith_fault_m~exchange_rateswith_fault_messa.
```



```

7
8     DATA: lt_return          TYPE TABLE OF bapiret2,
9     lwa_standard_data      TYPE zexchange_fault_data,
10    lwa_error_detail       TYPE zexchange_log_data.
11
12    DATA: lv_error_flag    TYPE c VALUE IS INITIAL.
13
14 *Assign proxy input message to a local variable
15    lwa_input = input.
16
17
18 *Validate if data is in the past
19
20 *Loop data records of in coming proxy message
21    LOOP AT lwa_input-exchange_rates-exch_rate INTO lwa_exch_rate.
22
23 *Check if from date is in the past
24    IF lwa_exch_rate-valid_from < sy-datum.
25        lv_error_flag = abap_true.
26
27 *Append errors to exception table
28
29    lwa_standard_data-fault_text   = 'Error Occured, exchanges rates were not posted in SAP'.
30    lwa_standard_data-fault_url   = 'https://sapintegrationhub.com'.
31
32    CLEAR lwa_error_detail.
33
34    lwa_error_detail-severity     = 'High'.
35    CONCATENATE 'Date' lwa_exch_rate-valid_from 'is in the past' INTO lwa_error_detail-text SEPARATED BY space.
36    lwa_error_detail-id = sy-tabix. "Record number of the input message
37    lwa_error_detail-url         = 'https://sapintegrationhub.com'.
38
39    APPEND lwa_error_detail TO lwa_standard_data-fault_detail.
40
41    ENDIF.
42
43 ENDOLOOP.
44
45 *if no errors found update the exchanges rates
46 if lv_error_flag is INITIAL.
47    LOOP AT lwa_input-exchange_rates-exch_rate INTO lwa_exch_rate.
48
49    MOVE-CORRESPONDING lwa_exch_rate TO lwa_bapi_input.
50

```



```

56   ENDOLOOP.
57
58 *Else send acknowledgment
59 ELSEIF lv_error_flag = abap_true.
60   RAISE EXCEPTION TYPE zcx_fault_message
61   EXPORTING
62   *      textid      =
63   *      previous    =
64   *      automatic_retry =
65   *      controller   =
66   *      no_retry     =
67   standard = lwa_standard_data.
68
69 ENDIF.
70
71 ENDMETHOD.
```

Test Fault Message using PI/PO Test Tool

Test Case

We will trigger a message with valid date in the past. In the test message <VALID_FROM> date is 2019-05-01 which is in the past.

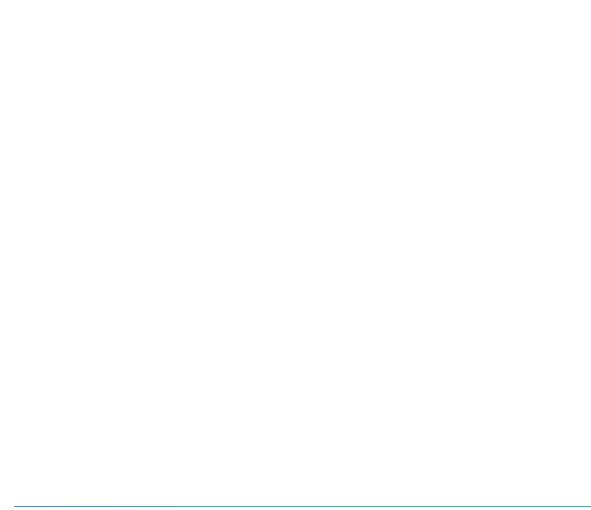
Test Message

```

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>2019-05-01</VALID_FROM>
8      <EXCH_RATE>0.89</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>
```

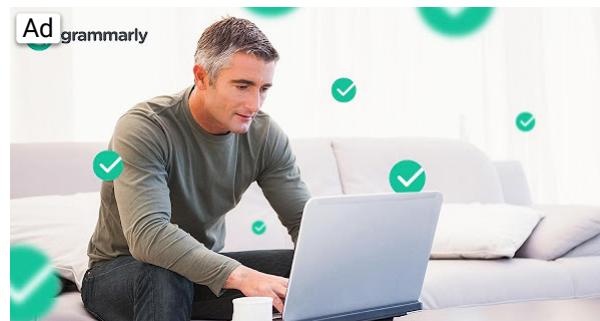


Go to transaction SXMB_MONI to monitor the application processing status of the message. You will notice it has been flagged as an error message.



You will be able to see the message in status “Application Error – Manual Restart Possible”.

If you need to analyze the behavior of the Proxy ABAP logic at runtime, set an [external debugging](#) break-point and trigger the interface from PI.



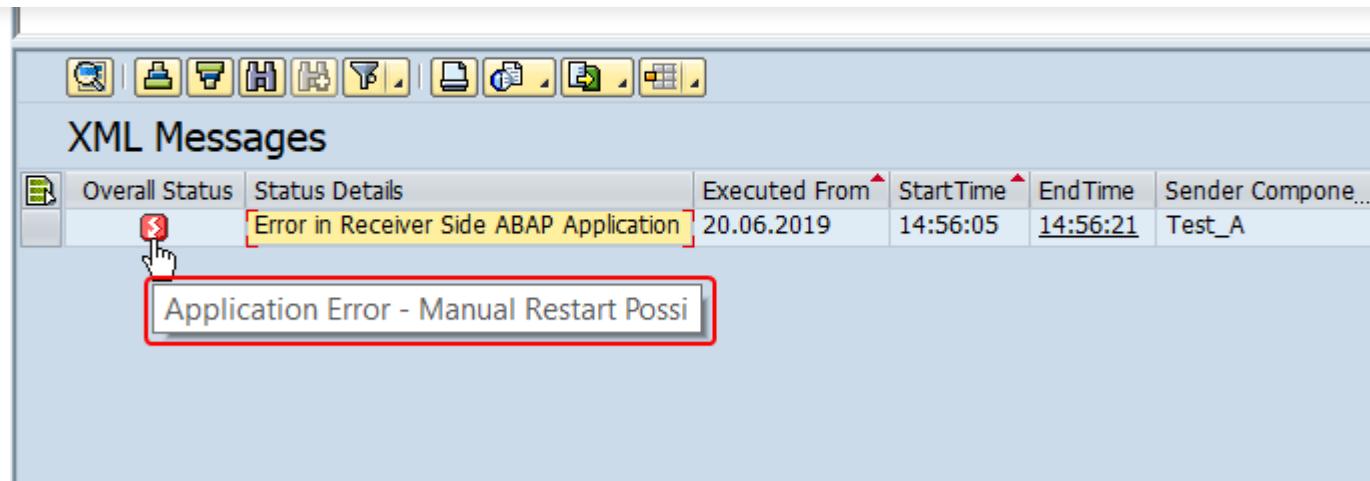
Grammar and Spelling Checker

Easily fix typos, grammatical mistakes, and other common issues before you hit Send

 Grammarly

[LEARN MORE](#)





The screenshot shows a table titled "XML Messages" with columns: Overall Status, Status Details, Executed From, StartTime, EndTime, and Sender Component. A single row is selected, highlighted with a yellow background. The "Status Details" column contains the text "Error in Receiver Side ABAP Application". A tooltip "Application Error - Manual Restart Possi" is displayed below the table, enclosed in a red box. A cursor arrow points to the "Overall Status" column header.

Overall Status	Status Details	Executed From	StartTime	EndTime	Sender Component
	Error in Receiver Side ABAP Application	20.06.2019	14:56:05	14:56:21	Test_A

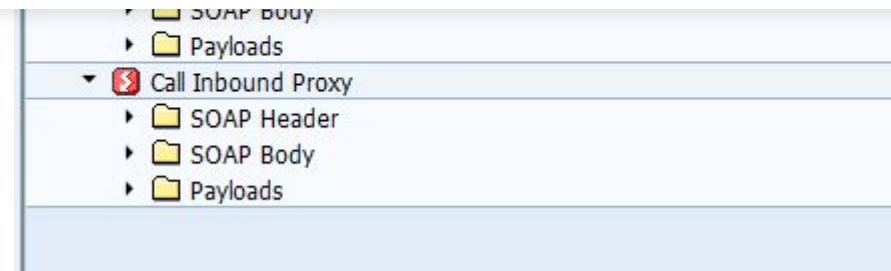
Message overall status in SXMB_MONI

Business users will be able to easily identify messages which were not posted in SAP due to ABAP validation not being satisfied. While messages which processed successfully will be in a successful status while messages failed due to validation will be flagged in an error status.

How to Monitor Fault Message Details in SXMB_MONI?

To view the message processing detail and payload, double click on the message in SXMB_MONI to go to the detail view.

You can view the **Inbound Message** and **Call Inbound Proxy** messages in the navigation tree.



SXMB_MONI XML message navigation tree

Go to **Main Document** section of **Call Inbound Proxy** message to view the error detail. This detail you can view here, we appended from the ABAP proxy logic.

The screenshot shows the SAP Integration Hub interface. On the left, a tree view under 'Call Inbound Proxy' shows 'Payloads' and 'MainDocument (application/xml)' highlighted with a red box. A red arrow points from this box to the right-hand XML editor area. The XML code in the editor is as follows:

```

<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/">
  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"
      xmlns:SAP="http://sap.com/xi/XI/Message/30"
      xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/"
      xmlns:wsu="http://www.docs.oasis-open.org/wss/2004/01/oasis-200401-wssecurity-utility-1.0.xsd">
        <SAP:MessageClass>ApplicationMessage</SAP:MessageClass>
        <SAP:ProcessingMode>asynchronous</SAP:ProcessingMode>
        <SAP:MessageId>C4F9B507-9359-11E9-96F5-0000008CEA6A</SAP:MessageId>
        <SAP:TimeSent>2019-06-20T12:48:57Z</SAP:TimeSent>
    </SAP:Main>
    <SAP:Sender>
      <SAP:Service>Test_A</SAP:Service>
    </SAP:Sender>
  </SOAP:Header>
  <?xml version="1.0" encoding="UTF-8" ?>
  <n0:FaultMessage xmlns:n0="urn:Target_System:ExchangeRates">
    xmlns:prx="urn:sap:com:proxy:SSX:/1SAI/TAS528DFF059A91C484745F:753">
    <standard>
      <faultText>Error Ooccured, exchanges rates were not posted in SAP</faultText>
      <faultUrl>https://sapintegrationhub.com</faultUrl>
    </standard>
    <faultDetail>
      <severity>High</severity>
      <text>Date 20190501 is in the past</text>
      <url>https://sapintegrationhub.com</url>
      <id>1</id>
    </faultDetail>
  </n0:FaultMessage>

```

Fault Message data displayed in SXMB_MONI under main document section



This entry was posted in [Fault Messages](#) and tagged [ABAP](#), [ABAP Proxy](#), [Asynchronous](#), [Asynchronous ABAP Proxy interface](#), [BAPI_EXCHANGERATE_CREATE](#), [Call Inbound Proxy](#), [ExchangeFaultData](#), [ExchangeLogData](#), [Fault Messages](#), [faultDetail](#), [faultText](#), [faultUrl](#), [File to Proxy](#), [Inbound Proxy](#), [sproxy](#), [SXMB_MONI](#).



ISURU FERNANDO

Hi, I am Isuru Fernando, Senior SAP Integration Consultant with 10 years of SAP full-cycle implementation and support project experience. From the early days, I had a passion for coding, software development, and even tech-related. I started my carrier as an ABAP developer and soon found my love for system integration when

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



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!

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

[What is SAP PI \(PO\) – Ultimate Guide](#)

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
QuietComfo
\$299.00

[Shop now](#)



2019 Apple
MacBook Pr
\$2,285.00

[Shop now](#)



(Renewed)
Apple iPhone
Pro
\$630.00

[Shop now](#)



[Privacy Inform](#)



Nikon D3500
24.2MP DSLR
\$799.99

[Shop now](#)



Bose
SoundSport.
\$239.97

[Shop now](#)

7 THOUGHTS ON “HANDLING FAULT MESSAGES IN ASYNCHRONOUS ABAP PROXY”



Septic says:

Hi Isuru,

Good post! Currently stuck in a situation. The flow is 3rd Party(REST) -> PO -> ECC. My requirement is to return a fault message when ECC is down. I understand this is possible through fault messages, but I don't see how. I have done the PI

[Privacy](#) · [Terms](#)



AUGUST 16, 2019 AT 1:02 PM

REPLY

*Divakar* says:

Hi Isuru,

Good post! Currently stuck in a situation. The flow is 3rd Party(REST) -> PO -> ECC. My requirement is to return a code when ECC is down. I understand this is possible through fault messages, but I don't see how. I have done the PI part, but how can the ABAP part of fault message be configured to return something when the whole ECC is down? Any advice is much appreciated 😊

AUGUST 16, 2019 AT 1:02 PM

REPLY

*Gurpreet Singh* says:

Hi Isuru,

Great post!!

Any idea if I need the async fault message back to SAP PI.

Regards,

SEPTEMBER 23, 2019 AT 7:20 AM

REPLY

*Gurpreet Singh* says:

Hi Isuru,

Great post!!

Any idea if I need the async fault message back to SAP PI.

Regards,

SEPTEMBER 23, 2019 AT 7:20 AM

Privacy - Terms



Try using the channel parameter 'XMBWS.NoSOAPIgnoreStatusCode'. Check if this [thread](#) would help.

Cheers!

Isuru

SEPTEMBER 23, 2019 AT 2:09 PM

REPLY

Leave a Reply

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

Comment

Name *

Email *

Website

Privacy - Terms



The ezoic logo, which is a green circle with a white 'e' symbol followed by the word "ezoic" in a bold, sans-serif font.

report this ad

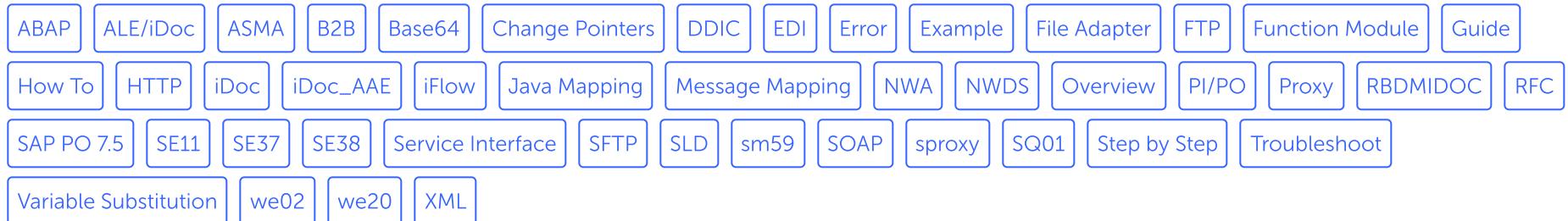
REGRESSION TEST AUTOMATION

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

[TRY FOR FREE](#)

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

TAG CLOUD





The ezoic logo, which consists of a green circle with a white 'e' and the word "ezoic" in a bold, sans-serif font.

report this ad

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 \(User Manual\)](#)

21
Mar [POS Integration with SAP S4 HANA and CAR](#)
[2 Comments](#)

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



CATEGORIES

[ABAP](#)



[AIF](#)

[BASIS](#)



[CPI](#)

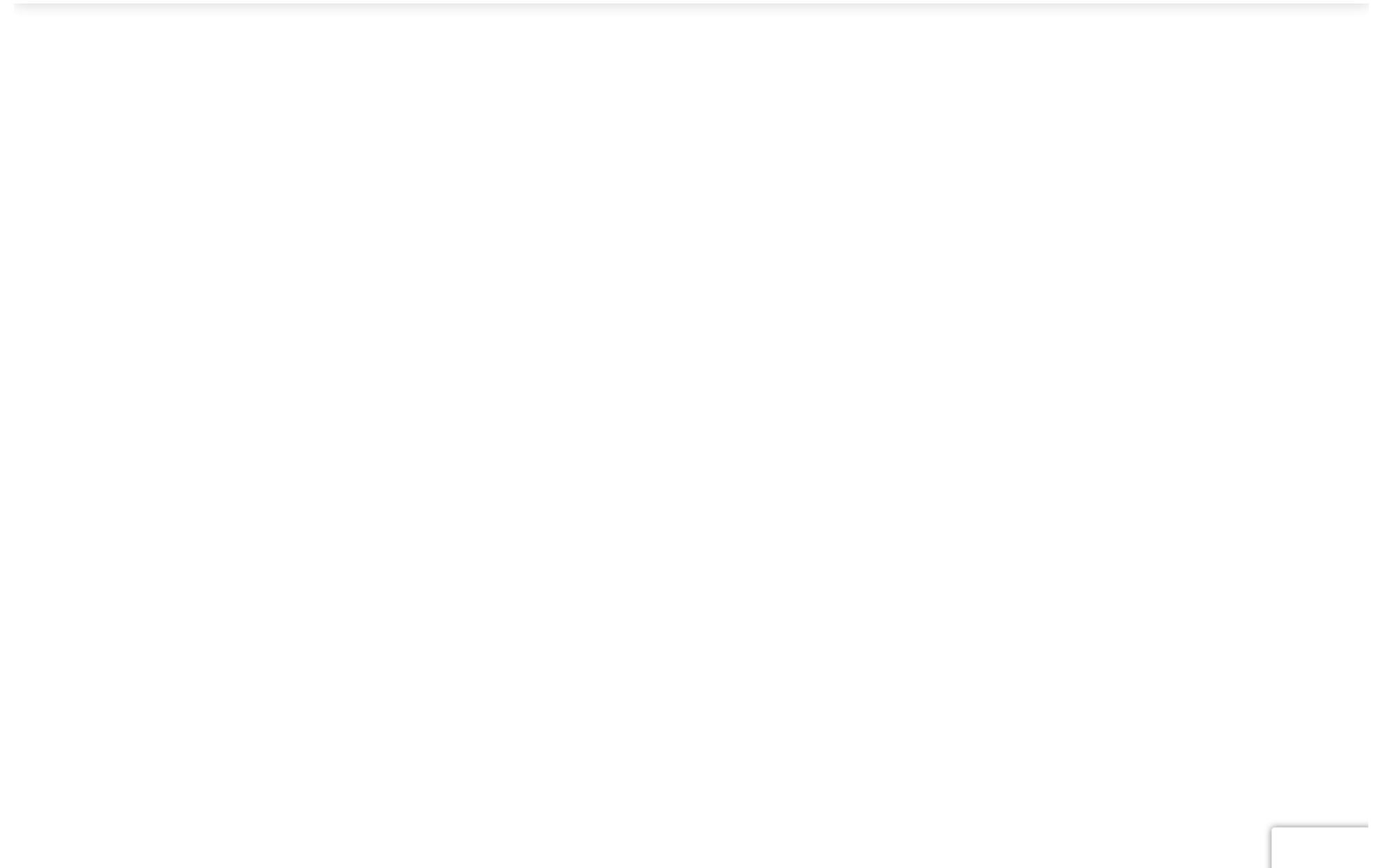
[How To](#)

[Integration Architecture](#)

[Integration Scenarios](#)

[PI/PO](#)







SITE LINKS

[Home](#)

[Blog](#)

[Write for Us](#)

[Disclaimer](#)

[Privacy Policy](#)

[Privacy - Terms](#)



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



Copyrights 2020 © SAP Integration Hub

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