Developing Applications for Integration between PI and SAP ERP in Different Network Domains or Landscapes

Applies to:

SAP NetWeaver Process Integration 7.1+
SAP NetWeaver 7.02 (ABAP)

Summary

This document describes the application development process for integration between Process Integration and SAP ERP systems when the two environments are physically located in different geographic areas without any network connections except through the standard public internet.

Author: William Li
Company: SAP Labs LLC
Created on: February 2012

Author Bio

William Li is a member of the SAP NetWeaver Solution Management team focusing on Process Orchestration and Integration at SAPLabs. He has been with SAP since 1998 - originally as a SAP America consultant then as a developer and NW RIG member at SAPLabs. He joined SAP Solution Management in April of 2010.
Table of Contents

Introduction ................................................................................................................................. 3
Restriction............................................................................................................................... 3
Architecture Overview ............................................................................................................ 3
Integration Development ......................................................................................................... 5
  1. Obtaining the WSDL from the Service Interface ................................................................. 5
  2. Generate the ABAP Proxy from the WSDL ......................................................................... 5
  3. Proxy Outbound Configuration .......................................................................................... 11
  4. ERP Proxy Configuration .................................................................................................. 14
  5. PI Integration Directory Configuration ............................................................................ 16
  6. Firewall Configuration on the PI Side ................................................................................ 16
Copyright .................................................................................................................................. 17
Introduction

Developing integrations between Process Integration (PI) and SAP ERP (ABAP) systems within the same landscape or network environment can be achieved easily through native tools and intranet network connectivity. The service interfaces can be designed on the PI system using the Enterprise Service Repository (ESR). With PI’s ESR connected to the SAP ERP system within the same landscape, ABAP proxy of the service interface can easily be generated in the ERP system using transaction SPROXY. The generated ABAP proxy can be used by any programs in the ERP system.

However, what can we do when the SAP ERP system is not in the same network environment within the same landscape? How do we generate the ABAP proxies? Can the ERP system send a message to the PI system using the public internet?

In this document, we will go through the steps necessary to:

1. Generate the ABAP proxies in the ERP system, which is designed on the PI system at another network location without access to the ESR.
2. Configuration requirement to send messages from ERP system to PI through the internet.

Restriction

As of NW 7.02, the ERP system will have to have access to a System Landscape Directory (SLD). During runtime, the SLD is used to resolve the sender system name, which must be included in the message. The sender name is necessary for PI to determine the interface to use and the mapping program to execute during the integration process.

With NW 7.03, which will be released in the future, the system name can be configured in the ERP system, without using the SLD.

Architecture Overview

Below is an overview of the two environments:
The architecture contains the following characteristics:

- Messaging between PI and the ERP system is using the XI protocol, which is HTTP-based. XI protocol provides reliable messaging with guaranteed delivery. It can also be used for asynchronous messaging requiring the quality of service of EOIO (exactly-once-in-order).
- The two environments are separated by firewalls.
- HTTP/S or SSL is used for network security reasons. It is also possible to use client digital certificates between the client and company to further identify the sender to the company in order to further secure the integrations.
- From the client location, an HTTP port may need to be opened to allow message to go out. Outgoing messages will not compromise network security from external sources.
- At the company location, a DMZ should be established to protect network messages coming into the company environment. DMZ is a common setup for companies who are doing B2B integrations with their trading partners. It is very likely that the company already has a DMZ.
- Within the DMZ, a reverse-proxy should be installed to “hide” the internal network access information from the outside. Reverse-proxy can also be used to protect companies from external network attacks. Reverse-proxy can be either software or hardware. SAP provides a software reverse-proxy, SAP WebDispatcher. It is very likely the company already has a reverse-proxy in place.
- The PI system in the company has an ESR and an SLD. Both of them are included with the PI license. PI also comes with a set of technical adapters, e.g. JDBC (as seen in the diagram), SOAP, File, Mail, etc.
- The ERP system on the client side has access to an SLD. SLD is part of NW Java server. SLD is available on various SAP systems, e.g. Solution Manager.

NOTE: The using of SSL, firewall, DMZ and reverse-proxy are standard network security practices. They are unrelated to PI or any SAP software components. They are necessary to protect company’s network environments.
Integration Development

1. Obtaining the WSDL from the Service Interface

Normally, ABAP proxies are generated from ESR on the ERP system using transaction SPROXY. However, in our case, the ERP system does not have access to the ESR. Therefore, we must have an alternative to generate the ABAP proxy from the service interface.

ABAP proxies can be generated from the ESR, or it can be generated from the WSDL of the service interface. In our case, we will use the WSDL. After the service interface has been designed, we can obtain the WSDL as follows:

1. Click on the tab “WSDL”
2. Click on the icon “Export WSDL to File”. Assign a name and directory location to the file.
3. The WSDL file can be emailed to the ABAP developer at the client location.

2. Generate the ABAP Proxy from the WSDL

Since the ABAP proxy cannot be generated using the transaction SPROXY, the transaction SE80 can be used to generate the ABAP proxy.

a. In SE80, provide a package where the development object will be saved. In this case, I am using a temporary location.
b. Create an “Enterprise Service”:

![Object Navigator](image1)

- **Create**
  - Change
  - Display
  - Copy...
  - Delete
  - Check
- **Deploy**
  - Object Directory Entry
  - Transport Entry
- **Enterprise Service**

![Service Consumer](image2)

c. Select “Service Consumer” and “Continue”:
d. Select option: external WSDL, and “Continue”:

![Diagram showing external WSDL selection]

```
Choose a generation source for the proxy.
```

- Backend
- Enterprise Service Repository
- external WSDL

```
Back  Continue  Cancel
```

e. Select option: Local File, and “Continue”:

![Diagram showing local file selection]

```
Choose a generation source for the proxy.
```

- Local File
- URL
- HTTP Destination
- UDDI Registry
- Service Registry

```
Back  Continue  Cancel
```
f. Enter WSDL file name and “Continue”:
Enter a package name to save the proxy and a prefix for the names of the objects which will be created during the proxy generation process:

g. Click on “Complete”:
h. The following object will be created. Save and activate the object.

This object is the ABAP proxy which is the same as if it was generated using the transaction SPROXY.

i. Below is a sample ABAP report which uses the ABAP proxy generated above:

```abap
REPORT ZTST_PROXY_CONSUMER.
parameters: _KEY_ TYPE ZTST_MT_REQUEST-MT_REQUEST-
KEY LOWER CASE default 'key1'.
DATA: lv_output TYPE ZTST_MT_REQUEST,
  lv_input TYPE ZTST_MT_RESPONSE,
  lv_system_error TYPE REF TO cx_ai_system_fault,
  GetInfo TYPE REF to ZTST_CO_SELECT_SYNC_OUT,
  lv_record TYPE ZTST_DT_RESPONSE_RECORD_TAB with header line.

  * provide key value
  lv_output-MT_REQUEST-KEY = KEY.

  * Output key value
  write: /, 'KEY: ', lv_output-MT_REQUEST-KEY.
  write: /, '___________________________________________'.
  TRY.
    create object GetInfo.
    call method GetInfo->SELECT_SYNC_OUT
      exporting
```
input = lv_output
importing
output = lv_input.
CATCH cx_ai_system_fault INTO lv_system_error.
   write: / "system fault: ', 'code:', lv_system_error->code,
   'errortext:', lv_system_error->errortext.
ENDTRY.
loop at lv_input-MT_RESPONSE-RECORD into lv_record.
   write: /, lv_record-KEY.
   write: /, lv_record-VALUE_1.
   write: /, lv_record-VALUE_2.
endloop.

3. Proxy Outbound Configuration

To send message through the internet to the PI system on another network, we will need to create an RFC connection. Then, assign this RFC connection to SXMB_ADM.

a. Create RFC connection of type “H”:

![Configuration of RFC Connections](image)
b. Use the IP address and port number of the PI system. The IP address and port will probably be different than the actual address, due to reverse-proxy configuration to hide the actual address and port number.

c. Provide authentication information:
d. Enter RFC destination in transaction SXMB_ADM. Click on “Integration Engine Configuration”:

![Integration Engine: Administration](image)

e. Navigate to menu: **Edit → Change Global Configuration Data**

![Change Global Configuration Data](image)

f. Enter the RFC destination created above:

![Integration Engine Configuration Data](image)

This designates the PI system to receive the ABAP proxy message.
4. ERP Proxy Configuration

To send messages from the ERP ABAP system to the internet, it is possible that we will need to configure the proxy settings for the ERP system. This is very similar as to setting the proxy for internet browsers, e.g. Microsoft Internet Explorer.

To configure proxy settings, use transaction SICF:

a. In SICF, click “Execute”: 

![Image of Maintain Services screen]

b. Select menu: Client → Proxy Settings:

![Image of Maintain Service screen]
c. Select the appropriate options and enter values in the tab “Global Setting”:

![Proxy Configuration for HTTP Client](image1.png)

- Proxy Setting is Active
- No Proxy Setting for Local Server

Authorization: [Enter Value]

No Proxy for the Following Addresses

Filter Entries

- sap.corp
- sap-ag.de

Global Proxy Setting:

![Proxy Configuration for HTTP Client](image2.png)

- Host Name: proxy
- Port: 0000

Logon Data

- User Name: [Enter Value]
- Password: ********

Still Initial

d. Enter the appropriate values in the tab “HTTP Log” or “HTTPS Log”:

![Proxy Configuration for HTTP Client](image3.png)
5. PI Integration Directory Configuration

On the PI side, there is no change to the configuration. It is done just as though the integration is all in the same network environment. The Sender Communication Component of the ERP system name is configured in the SLD. This name must be the same on the ERP side as on the PI side. Below is an example.

<table>
<thead>
<tr>
<th>Display/Receiver Determination</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sender</td>
<td>Active</td>
</tr>
<tr>
<td>Communication Party</td>
<td>BS_ECC_B60</td>
</tr>
<tr>
<td>Communication Component</td>
<td>select_Sync_Out</td>
</tr>
<tr>
<td>Interface</td>
<td>select_Sync_Out</td>
</tr>
<tr>
<td>Namespace</td>
<td><a href="http://value.com">http://value.com</a></td>
</tr>
</tbody>
</table>

6. Firewall Configuration on the PI Side

This a simplified example where the Windows server is used.

On the PI side, the firewall must be setup to allow the TCP port used by the PI system through. In our case, the port used is 50500.

This configuration and setup may be completely different than the actual network configuration of the customer.