SAP NetWeaver
How-To Guide

How To… Develop, Monitor and Debug WS Consumer and Provider

Applicable Releases:

SAP NetWeaver 7.0 SP14

IT Practice:

Service SOA & Design

IT Scenario:

Enabling Enterprise Services

Version 1.0

January 2010

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.

These materials are provided “as is” without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

SAP NetWeaver “How-to” Guides are intended to simplify the product implementation. While specific product features and procedures typically are explained in a practical business context, it is not implied that those features and procedures are the only approach in solving a specific business problem using SAP NetWeaver. Should you wish to receive additional information, clarification or support, please refer to SAP Consulting.

Any software coding and/or code lines / strings (“Code”) included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressively prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP’s Support Services and may not be modified or altered in any way.

Document History

Document Version	Description
1.00	First official release of this guide

Typographic Conventions

Type Style	Description
Example Text	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation
Example text	Emphasized words or phrases in body text, graphic titles, and table titles
`Example text`	File and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
`Example text`	User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
`EXAMPLE TEXT`	Keys on the keyboard, for example, `F2` or `ENTER`.

Icons

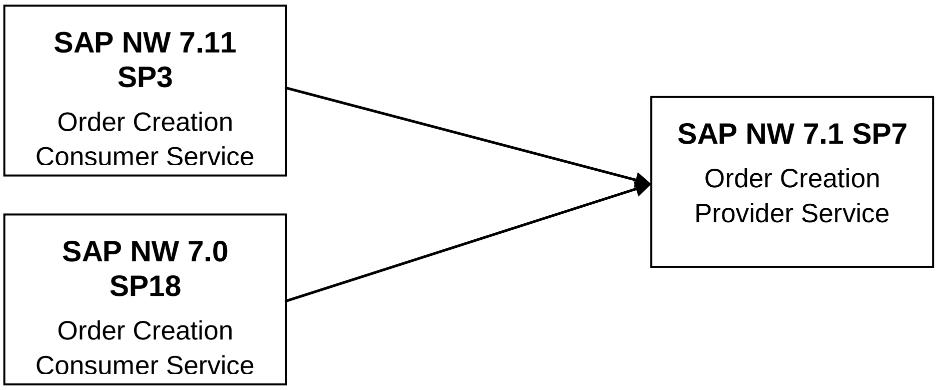
Icon	Description
⚠	Caution
💡	Note or Important
❖	Example
⬆	Recommendation or Tip

Table of Contents

1. Business Scenario

Since SAP NetWeaver 7.0 SP14, Web Services Reliable Messaging (WS-RM) was introduced. WS-RM provides reliable delivery of asynchronous messages using the SOAP protocol. WS-RM is included as part of the ABAP stack.

In this how-to guide, we will take a look at the development of consumer and provider services using WS-RM. In our business scenario, we will have a simple order creation process on SAP NW 7.1 SP7 system. To simplify the application for demo purposes, the service simply inserts the order information into a database table. Then, we will have 2 consumer services on SAP NW 7.0 SP18 and SAP NW 7.11 SP3 systems. This is to show interoperability between the different releases.


SAP NW 7.11 SP3
Order Creation Consumer Service
SAP NW 7.0 SP18
Order Creation Consumer Service
SAP NW 7.1 SP7
Order Creation Provider Service

In addition, we will take a look at the monitoring of the messages sent from the consumer to the provider system.

Note

WS-RM protocol supports only asynchronous messaging. When interfaces are configured as synchronous using WS-RM, it will still be treated as a standard SOAP request.

2. Background Information

The standard SOAP protocol does not guarantee the delivery of messages. With the inclusion of the WS-RM protocol, message delivery is guaranteed. When a sequence of messages is sent from the consumer systems, these messages are tracked by the provider system. If any of the messages did not arrive, then the provider will request the missing message be re-sent. Consequently, a WS-RM runtime engine needs to be available at both the consumer and provider systems.

This guaranteed message delivery has been available with SAP NW PI/XI systems. Now the same quality of service can be handled by WS-RM, without using SAP NW PI/XI.

3. Prerequisites

3.1 Check and Verify WS-RM Configurations

Before we can use WS-RM, we must verify that WS-RM has been configured on the consumer and provider systems.

1. In transaction SICF, activate the following nodes, if not already activated:
   a. /sap/bc/srt (including sub nodes)

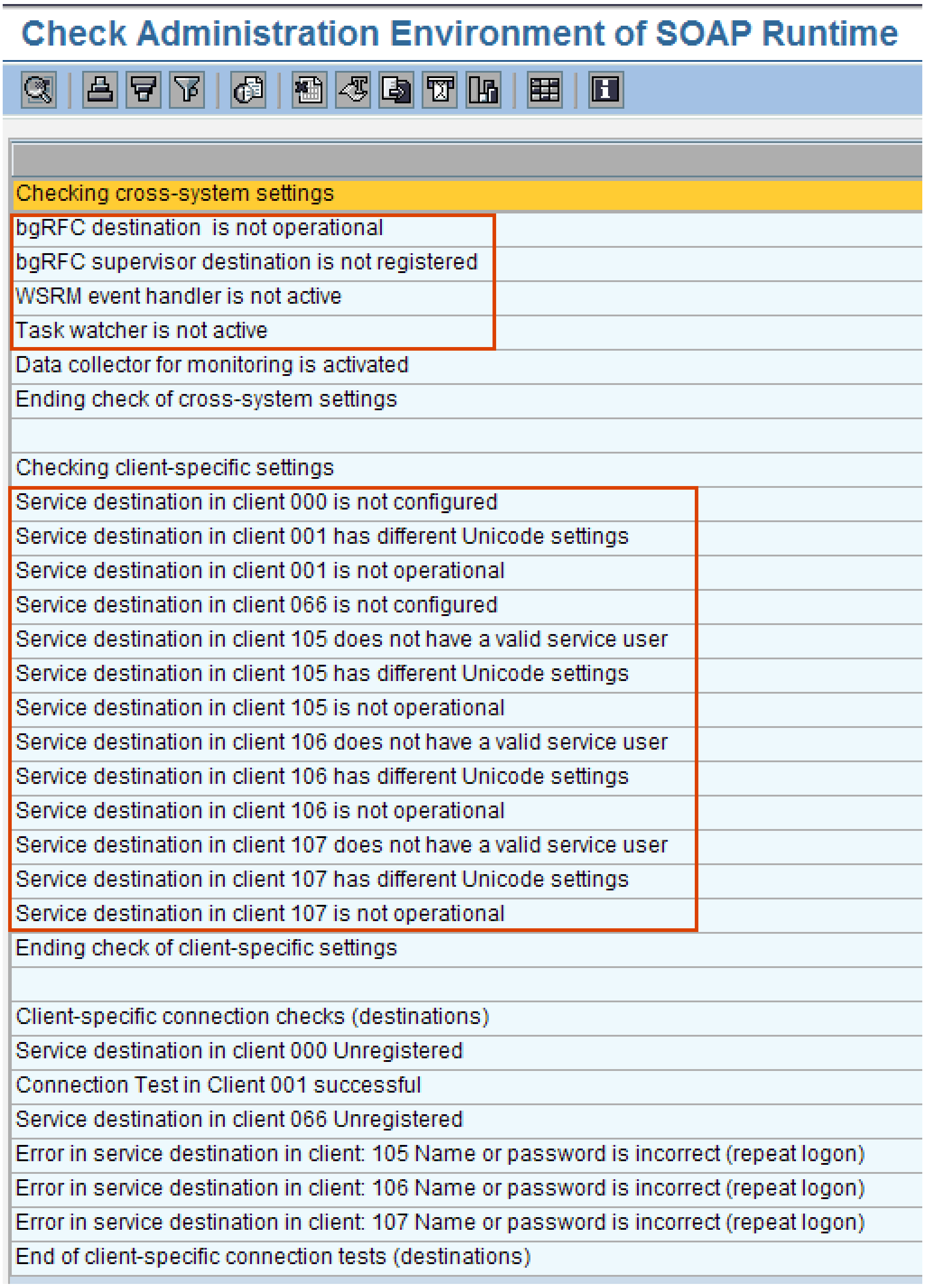
b. /sap/bc/webdynpro/sap/appl_soap_management

Tip

Check Note 1124553 for details (http://service.sap.com/sap/support/notes/1124553)

2. For each system, execute the report using SE38: SRT_ADMIN_CHECK

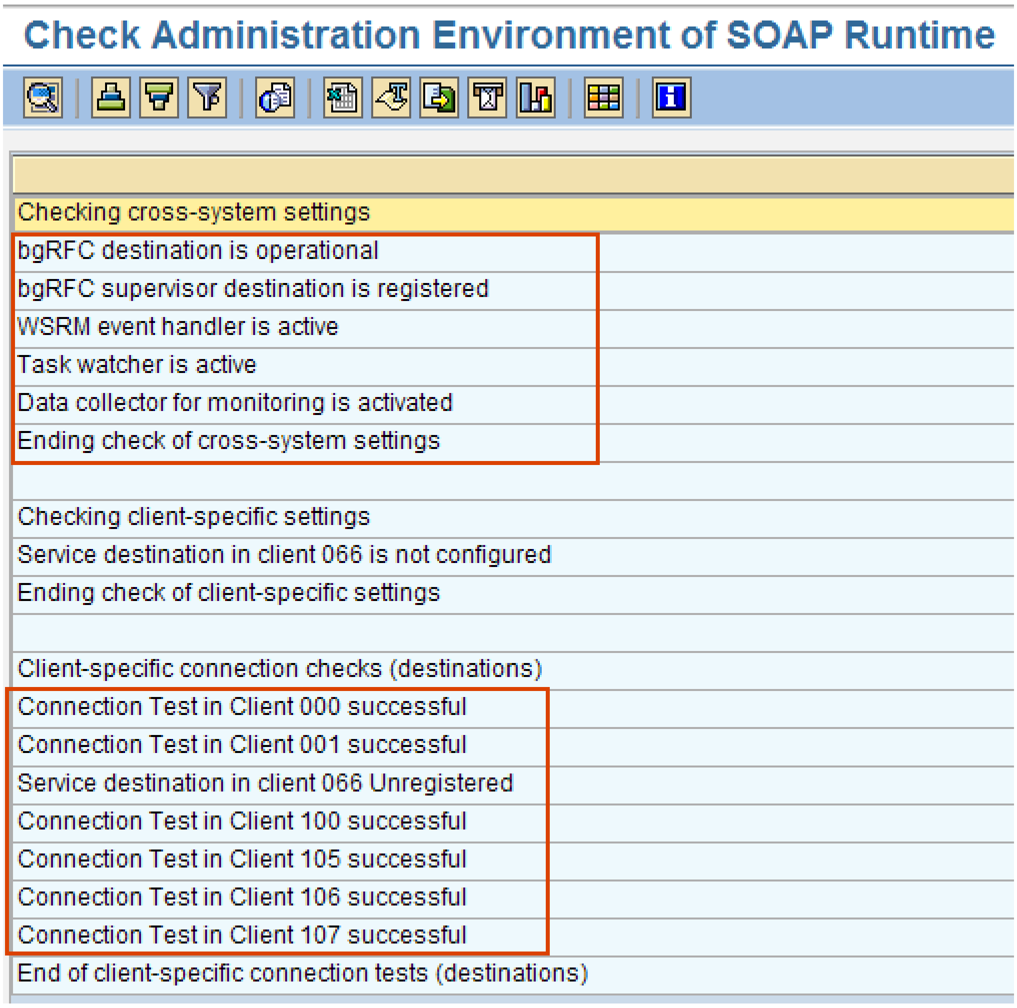
Example output of a not configured WS-RM system:



Check Administration Environment of SOAP Runtime

```
Checking cross-system settings
bgRFC destination  is not operational
bgRFC supervisor destination is not registered
WSRM event handler is not active
Task watcher is not active
Data collector for monitoring is activated
Ending check of cross-system settings

Checking client-specific settings
Service destination in client 000 is not configured
Service destination in client 001 has different Unicode settings
Service destination in client 001 is not operational
Service destination in client 066 is not configured
Service destination in client 105 does not have a valid service user
Service destination in client 105 has different Unicode settings
Service destination in client 105 is not operational
Service destination in client 106 does not have a valid service user
Service destination in client 106 has different Unicode settings
Service destination in client 106 is not operational
Service destination in client 107 does not have a valid service user
Service destination in client 107 has different Unicode settings
Service destination in client 107 is not operational
Ending check of client-specific settings

Client-specific connection checks (destinations)
Service destination in client 000 Unregistered
Connection Test in Client 001 successful
Service destination in client 066 Unregistered
Error in service destination in client: 105 Name or password is incorrect (repeat logon)
Error in service destination in client: 106 Name or password is incorrect (repeat logon)
Error in service destination in client: 107 Name or password is incorrect (repeat logon)
End of client-specific connection tests (destinations)
```

Example output of a configured WS-RM system:



Check Administration Environment of SOAP Runtime

```
Checking cross-system settings
bgRFC destination is operational
bgRFC supervisor destination is registered
WSRM event handler is active
Task watcher is active
Data collector for monitoring is activated
Ending check of cross-system settings

Checking client-specific settings
Service destination in client 066 is not configured
Ending check of client-specific settings

Client-specific connection checks (destinations)
Connection Test in Client 000 successful
Connection Test in Client 001 successful
Service destination in client 066 Unregistered
Connection Test in Client 100 successful
Connection Test in Client 105 successful
Connection Test in Client 106 successful
Connection Test in Client 107 successful
End of client-specific connection tests (destinations)
```

If the report output does not reflect similarity to a properly configured WS-RM, then follow the steps in Section 3.2, or else, skip the next section, Section 4.

3.2 Configure WS-RM for Consumer and Provider System.

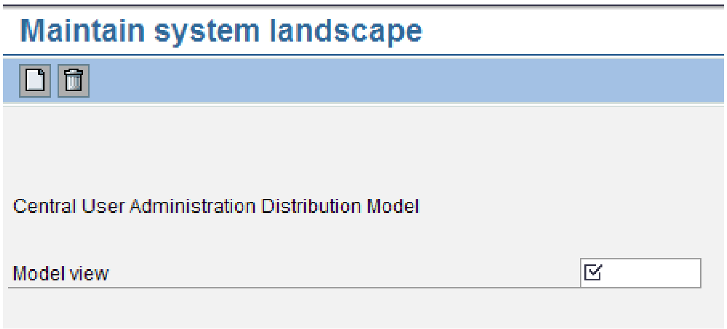
Important

Follow the steps in this section only if WS-RM has not been configured based on the instructions above.

Note

SAP Library on "Configuring the Web Service Runtime" can be found on: http://help.sap.com/saphelp_nwpi71/helpdata/en/46/abbc05ba0c2a7fe10000000a1553f6/frameset.htm

1. On **all** the clients, check if CUA is active by using transaction SCUA. If not active, the following screen should be displayed.



Maintain system landscape

Central User Administration Distribution Model

Model view ☑

- If not active, use function module (SE37) SRT_TECHNICAL_SETUP

  Note

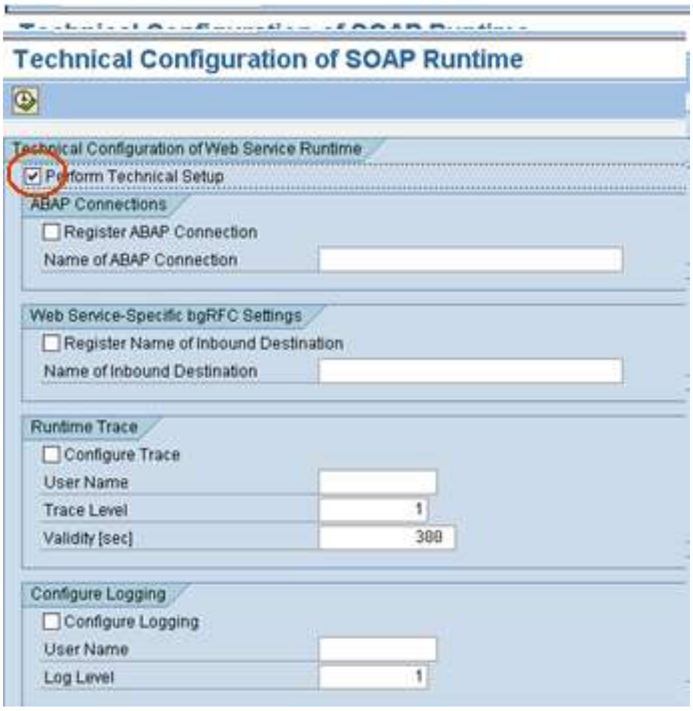
  SAP Note 1110741 Web service configuration can only be executed using SE37

- If active, use report (SE38) SRT_ADMIN

  Note

  SAP Note 1043195 Configuration of the Web service runtime

2. On all the clients, make sure the userid used to perform the configurations has SAP_ALL previleges, or DDIC.
3. On Client 000:
   a. Execute report (SE38) SRT_ADMIN
   b. Select option “Perform Technical Setup”



Technical Configuration of SOAP Runtime

Technical Configuration of Web Service Runtime

☑ Perform Technical Setup

ABAP Connections

□ Register ABAP Connection

Name of ABAP Connection

Web Service-Specific bgRFC Settings

□ Register Name of Inbound Destination

Name of Inbound Destination

Runtime Trace

□ Configure Trace

User Name

Trace Level 1

Validity [sec] 300

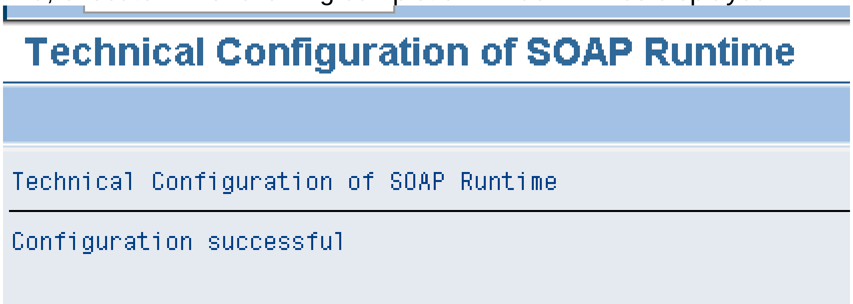
Configure Logging

□ Configure Logging

User Name

Log Level 1

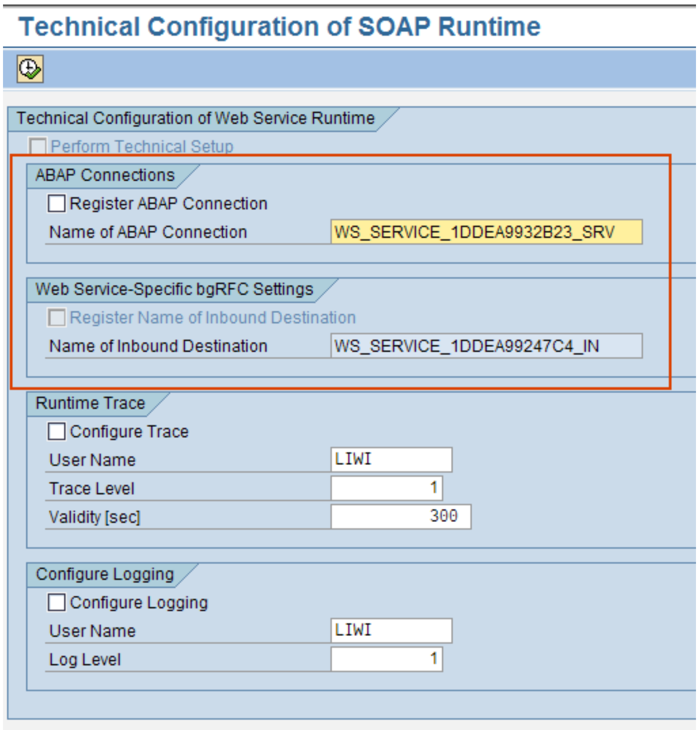
And, execute. The following completion window will be displayed.



Technical Configuration of SOAP Runtime

```
Technical Configuration of SOAP Runtime

Configuration successful
```

c. Re-execute report SRT_ADMIN, you should see ABAP Connections and Web Service-Specific bgRFC Setting are populated:



Technical Configuration of SOAP Runtime

Technical Configuration of Web Service Runtime

□ Perform Technical Setup

ABAP Connections

□ Register ABAP Connection

Name of ABAP Connection: WS_SERVICE_1DDEA9932B23_SRV

Web Service-Specific bgRFC Settings

□ Register Name of Inbound Destination

Name of Inbound Destination: WS_SERVICE_1DDEA99247C4_IN

Runtime Trace

□ Configure Trace

User Name: LIWI

Trace Level: 1

Validity [sec]: 300

Configure Logging

□ Configure Logging

User Name: LIWI

Log Level: 1

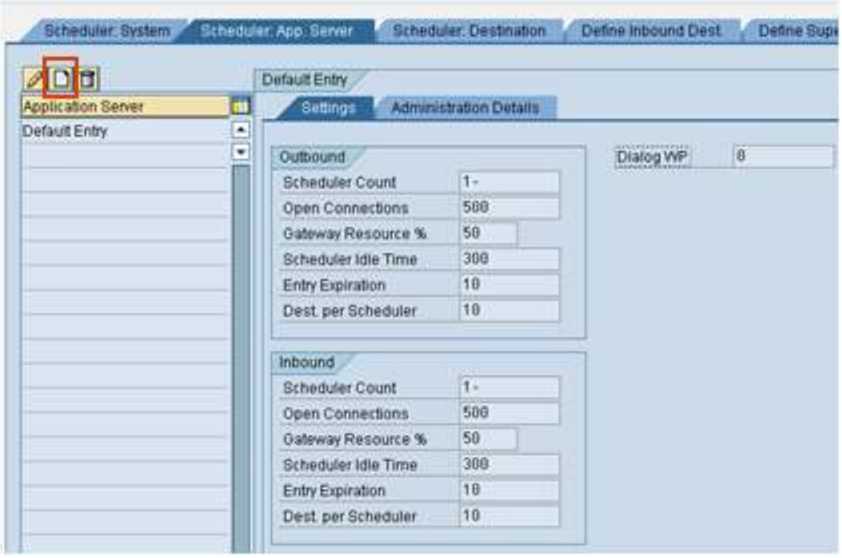
Tip

During the configuration process, we can use report SRT_ADMIN_CHECK to verify pending and completed tasks.

4. On Client 000: Configure bgRFC

   a. Execute transaction: SBGRFCCONF

   b. Click on “Create” in the tab “Scheduler App Server”:



Dialog WP	0

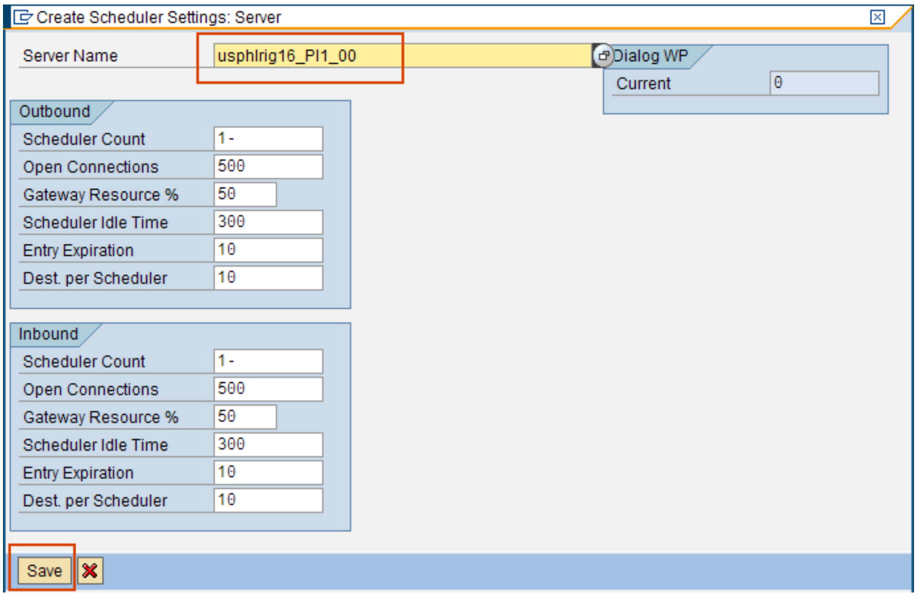
Outbound

Scheduler Count	1 -
Open Connections	500
Gateway Resource %	50
Scheduler Idle Time	300
Entry Expiration	10
Dest. per Scheduler	10

Inbound

Scheduler Count	1 -
Open Connections	500
Gateway Resource %	50
Scheduler Idle Time	300
Entry Expiration	10
Dest. per Scheduler	10

c. Provide a server name:



Server Name	usphlrig16_PI1_00

Dialog WP

Current	0

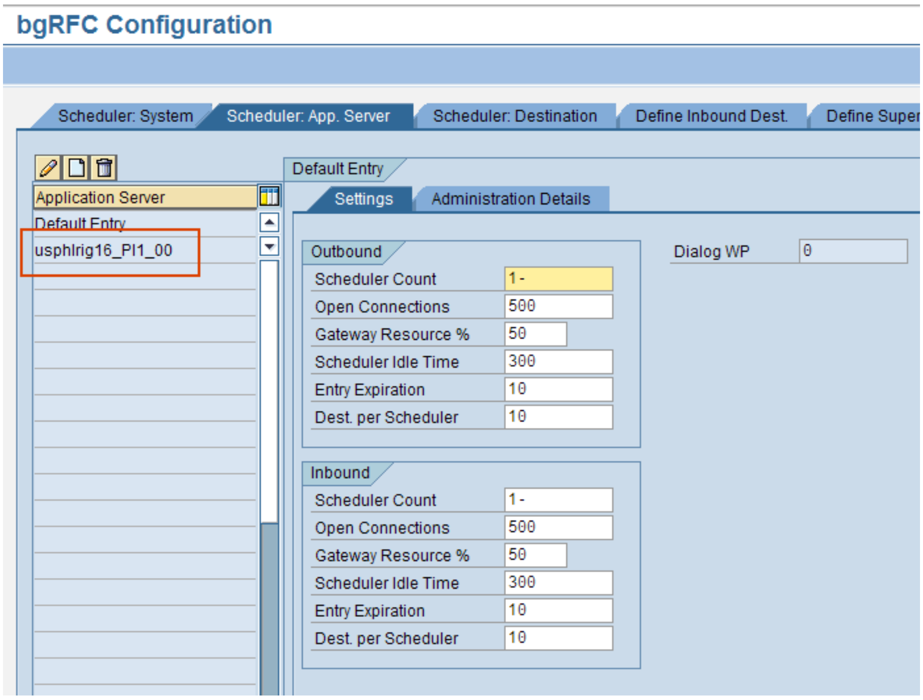
Outbound

Scheduler Count	1 -
Open Connections	500
Gateway Resource %	50
Scheduler Idle Time	300
Entry Expiration	10
Dest. per Scheduler	10

Inbound

Scheduler Count	1 -
Open Connections	500
Gateway Resource %	50
Scheduler Idle Time	300
Entry Expiration	10
Dest. per Scheduler	10

d. The will result:



bgRFC Configuration

Application Server

Default Entry

usphlrig16_PI1_00

Default Entry

Outbound

Scheduler Count	1-
Open Connections	500
Gateway Resource %	50
Scheduler Idle Time	300
Entry Expiration	10
Dest. per Scheduler	10

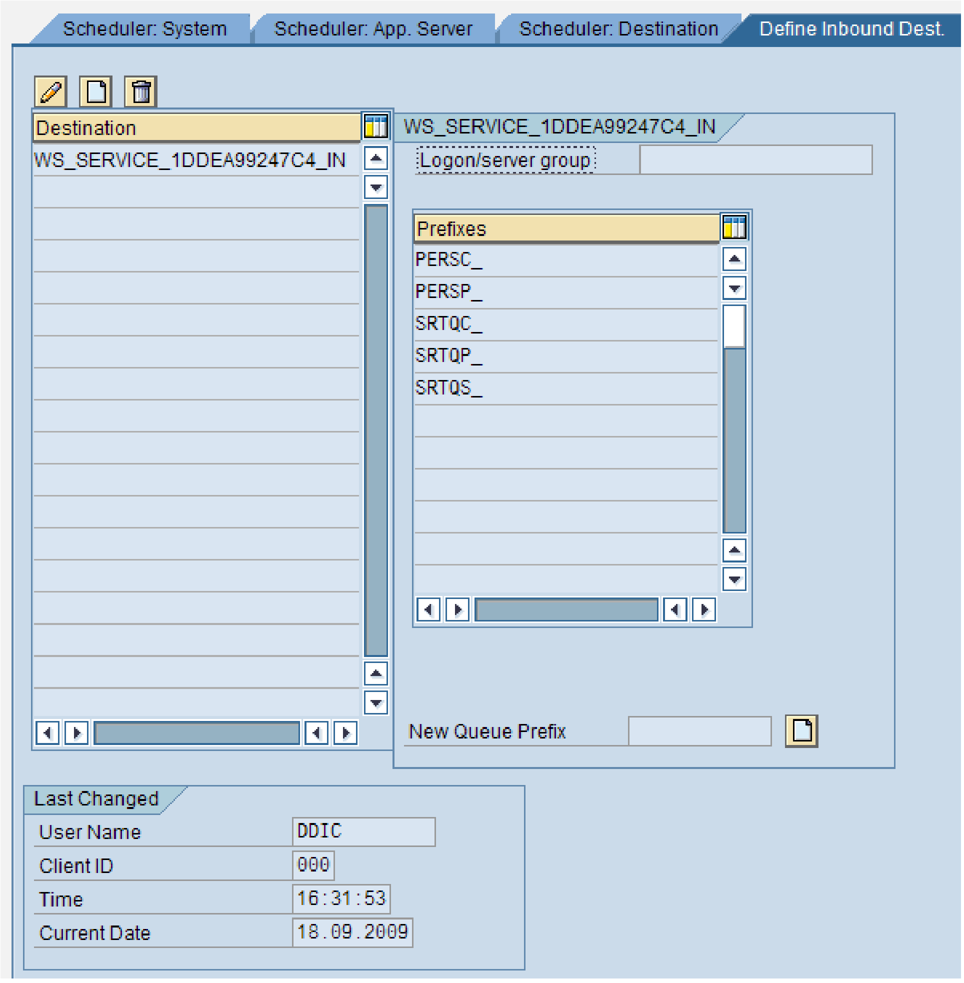
Dialog WP 0

Inbound

Scheduler Count	1-
Open Connections	500
Gateway Resource %	50
Scheduler Idle Time	300
Entry Expiration	10
Dest. per Scheduler	10

No other entries will be necessary.

e. Check the queue prefixes have been registered. If the setup was successful, no entries are necessary on this tab.



Destination

WS_SERVICE_1DDEA99247C4_IN

WS_SERVICE_1DDEA99247C4_IN

Logon/server group

Prefixes

PERSC_
PERSP_
SRTQC_
SRTQP_
SRTQS_

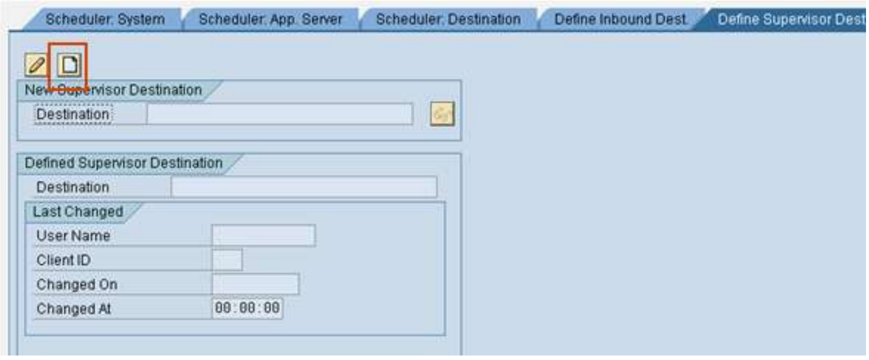
New Queue Prefix

Last Changed

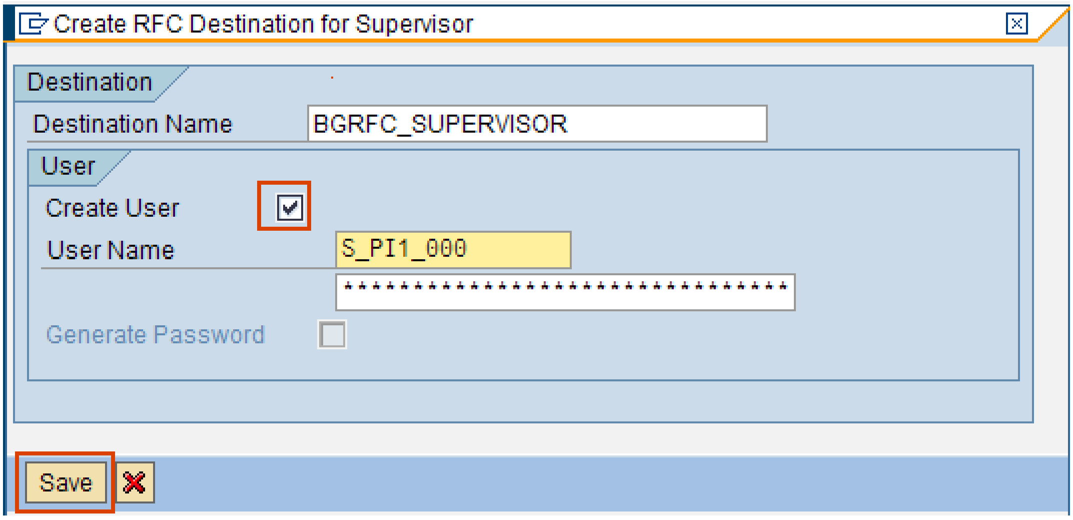
User Name	DDIC
Client ID	000
Time	16:31:53
Current Date	18.09.2009

f. Define RFC Destination by clicking on “Create” in the tab “Define Supervisor Destination” and enter the necessary information:

i.



ii.


Create RFC Destination for Supervisor

Destination

Destination Name: BGRFC_SUPERVISOR

User

Create User ☑

User Name: S_PI1_000

****

Generate Password □

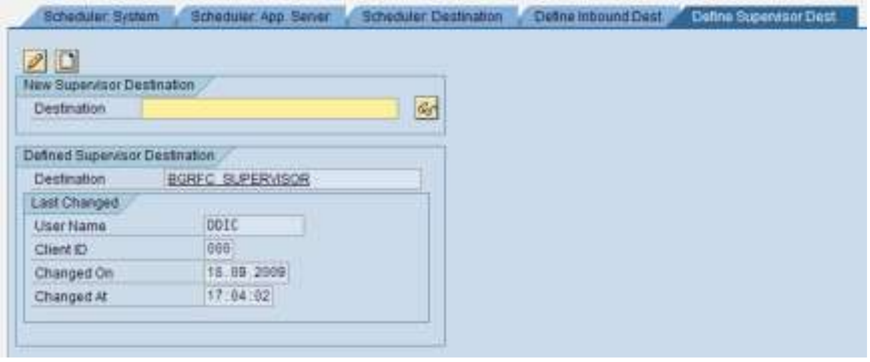
Provide anything for password. After “Save”, the userid S_PI1_000 should be created also.

You may want to verify that the userid is a “Service User” and has the role: SAP_BC_BGRFC_SUPERVISOR (add if not present)

The RFC Destination (type 3) should also have been created: BGRFC_SUPERVISOR

The RFC destination should be using the logon user: S_PI1_000

iii.



Defined Supervisor Destination

Destination: BGRFC_SUPERVISOR

Last Changed

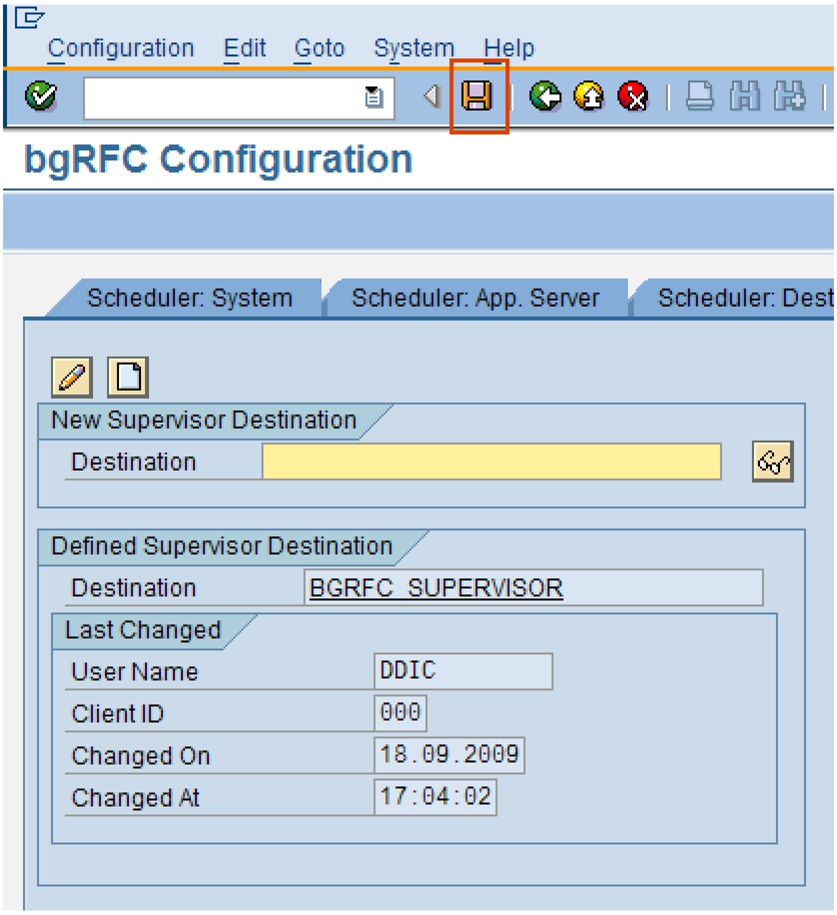
User Name: DDIC

Client ID: 000

Changed On: 16.09.2009

Changed At: 17:04:02

iv. Save the entries:



New Supervisor Destination

Destination

Defined Supervisor Destination

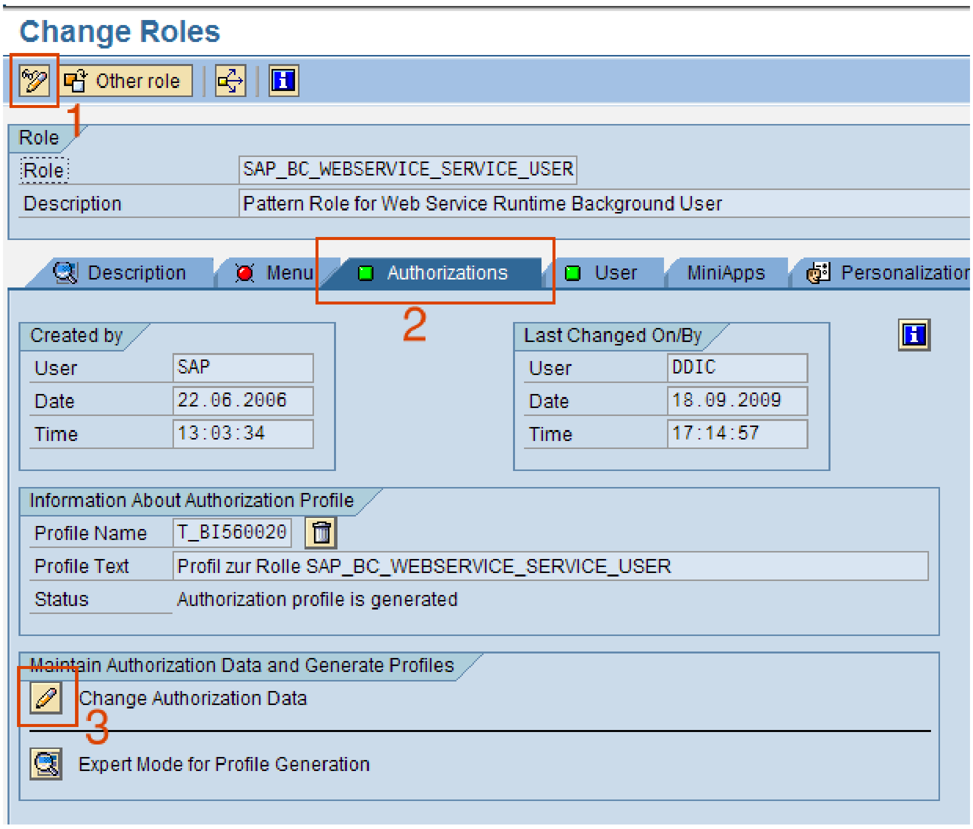
Destination BGRFC_SUPERVISOR

Last Changed

User Name	DDIC
Client ID	000
Changed On	18.09.2009
Changed At	17:04:02

g. Verify the configuration by executing the report SRT_ADMIN_CHECK. If the error, “No RFC authorization for function module RFCPING“, is present, then follow the steps below, using the transaction PFCG and add the role SAP_BC_WEBSERVICE_SERVICE_USER with authority for RFC function group SYST.

i. Go to the “Change” mode in PFCG, tab “Authorization” and click on “Change Authorization Data”:



Change Roles

Role

Role	SAP_BC_WEBSERVICE_SERVICE_USER
Description	Pattern Role for Web Service Runtime Background User

Created by

User	SAP
Date	22.06.2006
Time	13:03:34

Last Changed On/By

User	DDIC
Date	18.09.2009
Time	17:14:57

Information About Authorization Profile

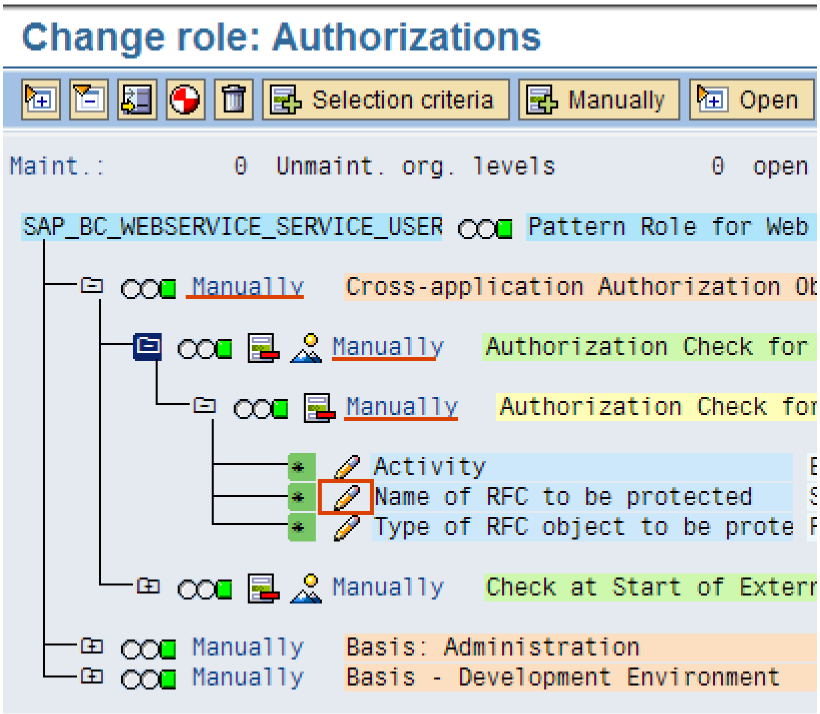
Profile Name	T_BI560020
Profile Text	Profil zur Rolle SAP_BC_WEBSERVICE_SERVICE_USER
Status	Authorization profile is generated

Maintain Authorization Data and Generate Profiles

Change Authorization Data

Expert Mode for Profile Generation

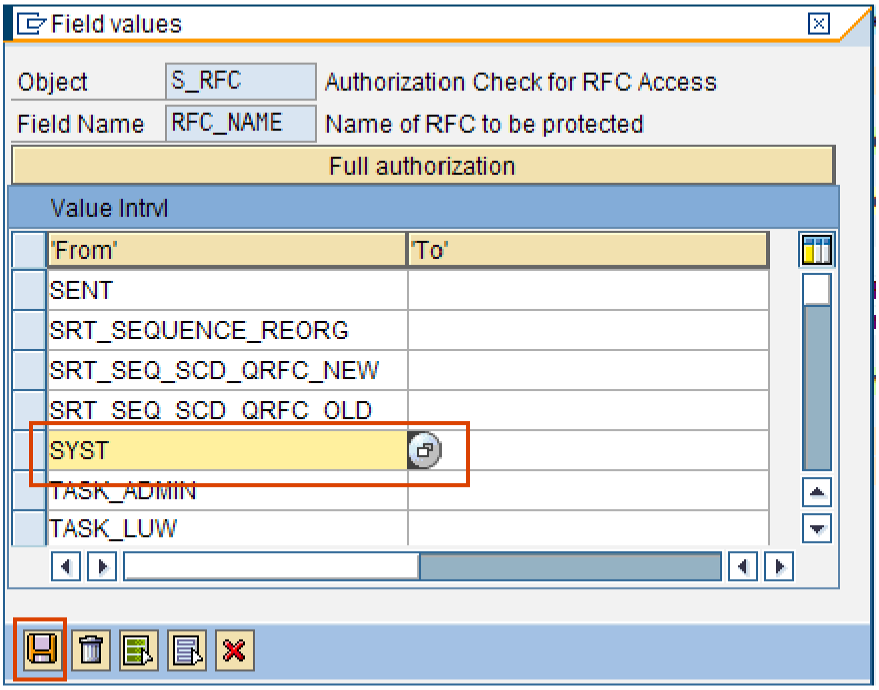
ii. Expand the folder per the screenshot below and click on “change” for “Name of RFC to be protected”:



Change role: Authorizations

```
Maint.:          0  Unmaint. org. levels        0  open

SAP_BC_WEBSERVICE_SERVICE_USER  Pattern Role for Web
   Manually  Cross-application Authorization Ob
      Manually  Authorization Check for
         Manually  Authorization Check for
            Activity                    E
            Name of RFC to be protected S
            Type of RFC object to be prote F
      Manually  Check at Start of Extern
   Manually  Basis: Administration
   Manually  Basis - Development Environment
```

iii. Enter “SYST” and “Save”:



Field values

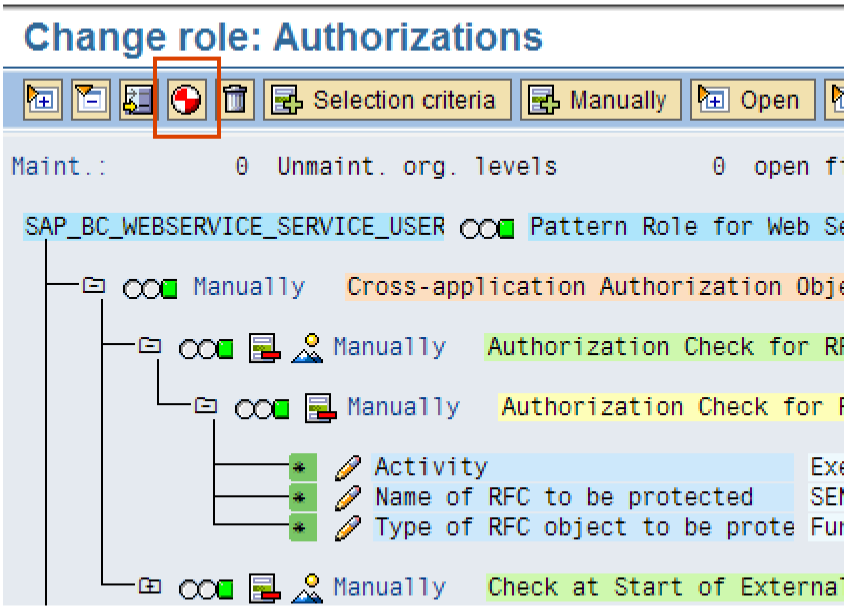
Object	S_RFC	Authorization Check for RFC Access
Field Name	RFC_NAME	Name of RFC to be protected

Full authorization

Value Intrvl

'From'	'To'
SENT	
SRT_SEQUENCE_REORG	
SRT_SEQ_SCD_QRFC_NEW	
SRT_SEQ_SCD_QRFC_OLD	
SYST	
TASK_ADMIN	
TASK_LUW	

iv. Generate the change:

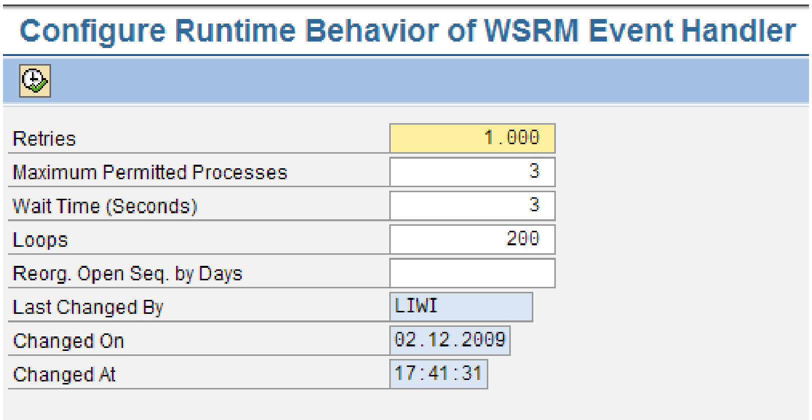


Change role: Authorizations

```
Maint.:          0  Unmaint. org. levels          0  open f
SAP_BC_WEBSERVICE_SERVICE_USER    Pattern Role for Web Se
   Manually   Cross-application Authorization Obje
      Manually   Authorization Check for RF
         Manually   Authorization Check for F
            Activity                        Exe
            Name of RFC to be protected     SEN
            Type of RFC object to be prote  Fur
      Manually   Check at Start of Externa
```

v. Execute report SRT_ADMIN_CHECK to verify that there are no more errors.

h. Configure WSRM runtime event handler by running report RSEHCONFIG:



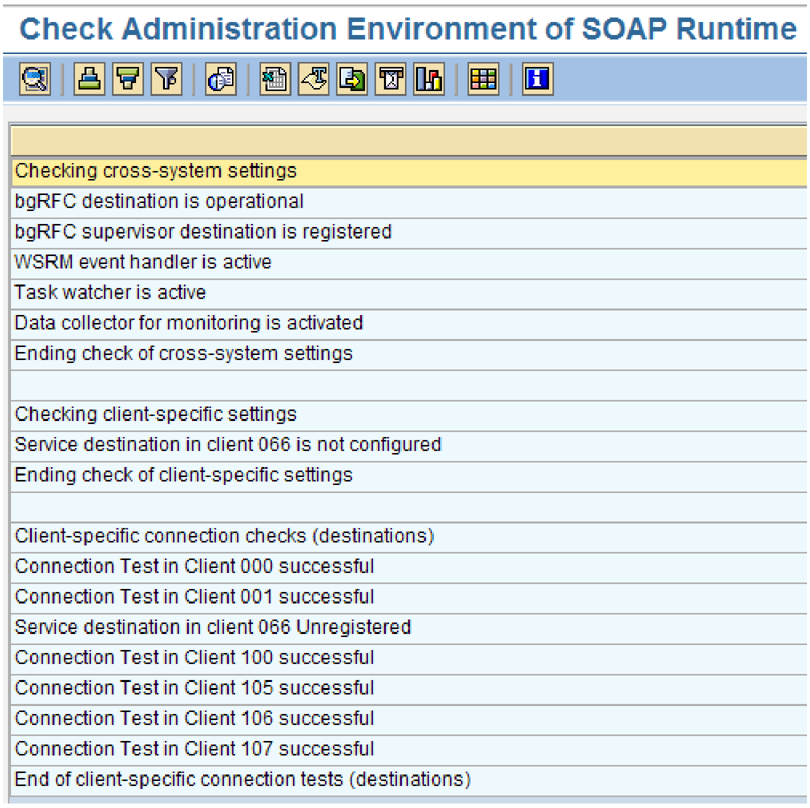
Configure Runtime Behavior of WSRM Event Handler

Retries	1.000
Maximum Permitted Processes	3
Wait Time (Seconds)	3
Loops	200
Reorg. Open Seq. by Days	
Last Changed By	LIWI
Changed On	02.12.2009
Changed At	17:41:31

i. Start background job BC_SAP_SOAP_RUNTIME_MANAGEMENT.

If not scheduled in transaction SM37, call transaction SM36, and choose *Standard Jobs - Standard Scheduling*. You must schedule the job hourly.

5. On all of the other clients, execute report SRT_ADMIN. Execute transaction PFCG to enhance role SAP_BC_WEBSERVICE_SERVICE_USER with authority for RFC function group SYST using the instructions outlined previously.
6. Execute report SRT_ADMIN_CHECK to verify the configurations:



Check Administration Environment of SOAP Runtime

```
Checking cross-system settings
bgRFC destination is operational
bgRFC supervisor destination is registered
WSRM event handler is active
Task watcher is active
Data collector for monitoring is activated
Ending check of cross-system settings

Checking client-specific settings
Service destination in client 066 is not configured
Ending check of client-specific settings

Client-specific connection checks (destinations)
Connection Test in Client 000 successful
Connection Test in Client 001 successful
Service destination in client 066 Unregistered
Connection Test in Client 100 successful
Connection Test in Client 105 successful
Connection Test in Client 106 successful
Connection Test in Client 107 successful
End of client-specific connection tests (destinations)
```

4. Step-by-Step Procedure

In this section, we will implement WS-RM provider and consumer applications on SAP NW.

The provider application is an ABAP proxy. The ABAP proxy is generated from an inbound service interface defined in the ESR. Once the proxy has been generated, SOAMANAGER will be used to create the WS-RM service.

On the consumer system, the URL from the provider web service is used to create the consumer proxy. Then, SOAMANAGER is used to create the consumer service. A program can use the consumer service to send a message to the provider service.

In summary, the steps are:

On Provider System:

1. Create service interface in ESR.
2. Generate ABAP proxy from the service interface.
3. Provide code to the ABAP proxy. In this case, we will create a simple program to insert data into a table.
4. Use SOAMANAGER to create a web service from the ABAP proxy.
5. Test the web service using WSNavigator.

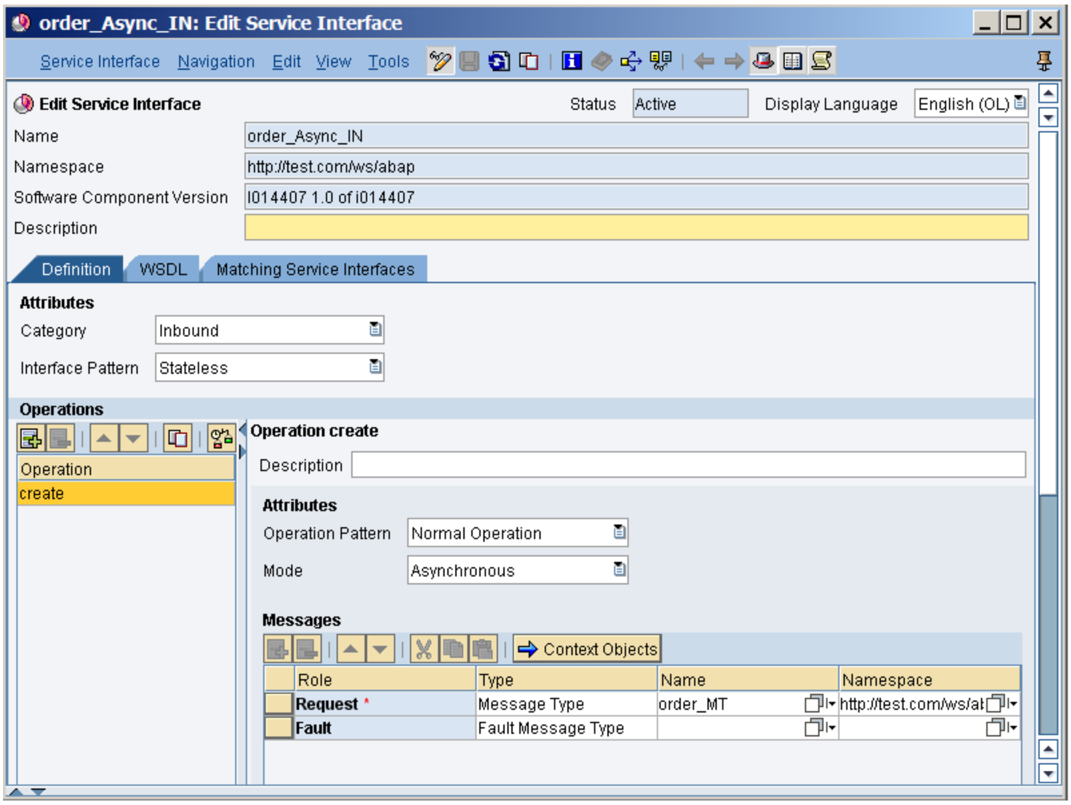
On Consumer System:

1. Create consumer proxy using the URL of the provider web service.
2. Use SOAMANAGER to create the consumer service from the proxy.
3. Create a program to call the consumer service.

4.1 Create the Provider Application

1. Create service interface in the ESR of the provider system.

   Create and activate the following service interface:



Edit Service Interface

Status: Active | Display Language: English (OL)

Name	order_Async_IN
Namespace	http://test.com/ws/abap
Software Component Version	I014407 1.0 of i014407
Description	

Attributes

Category	Inbound
Interface Pattern	Stateless

Operations

Operation
create

Operation create

Description

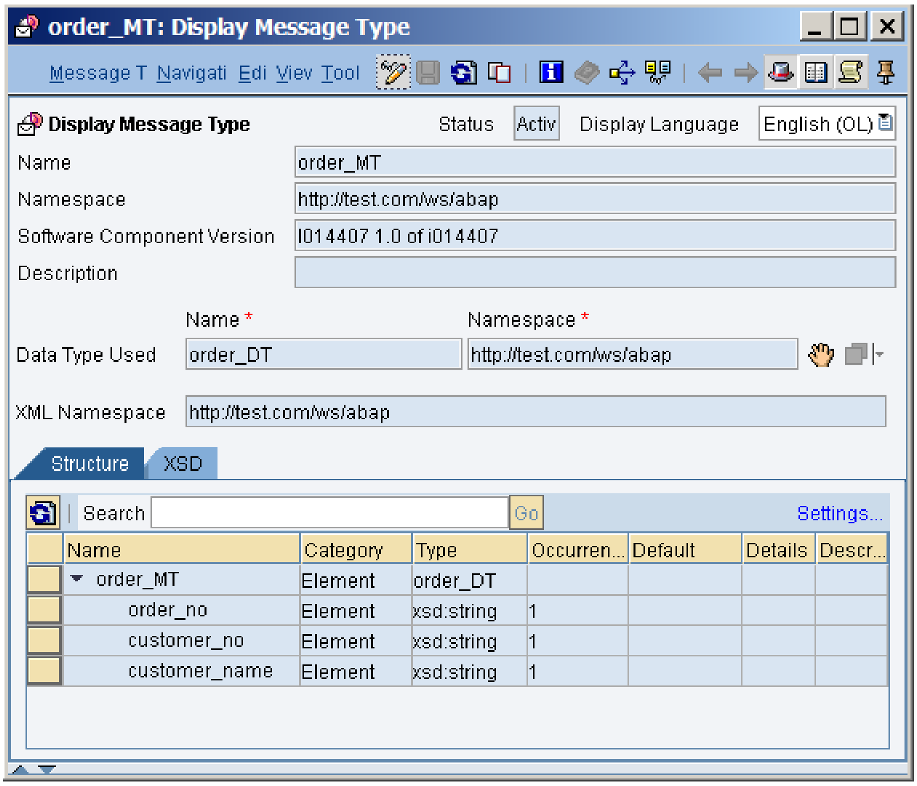
Attributes

Operation Pattern	Normal Operation
Mode	Asynchronous

Messages

Role	Type	Name	Namespace
Request *	Message Type	order_MT	http://test.com/ws/ab
Fault	Fault Message Type		

This interface is based on the message type:



Display Message Type

Status: Activ | Display Language: English (OL)

Name	order_MT
Namespace	http://test.com/ws/abap
Software Component Version	I014407 1.0 of i014407
Description	

	Name *	Namespace *
Data Type Used	order_DT	http://test.com/ws/abap

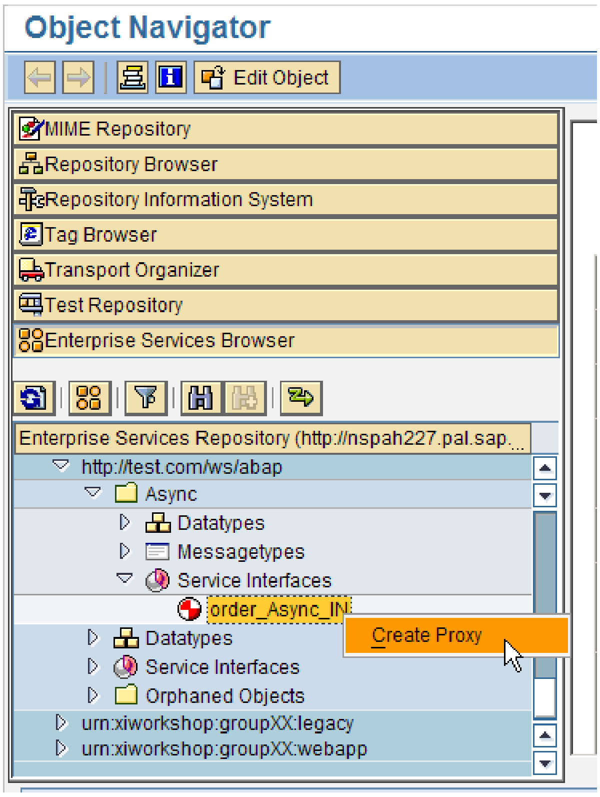
XML Namespace: http://test.com/ws/abap

Search

Name	Category	Type	Occurren...	Default	Details	Descr...
order_MT	Element	order_DT				
order_no	Element	xsd:string	1			
customer_no	Element	xsd:string	1			
customer_name	Element	xsd:string	1			

2. Generate ABAP proxy from the service interface.

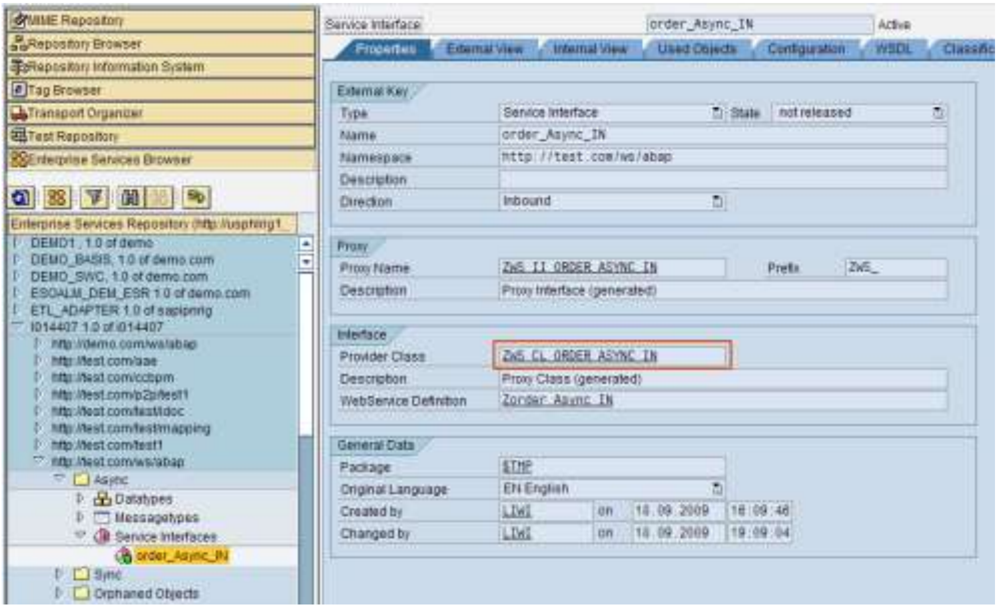
   In transaction SPROXY, create the proxy from the service interface:



Save and activate after creation.

3. Provide code to the ABAP proxy. In this case, we will create a simple program to insert data into a table.

   a. Open the proxy and double-click on the Provider Class:



Service Interface: order_Async_IN Active

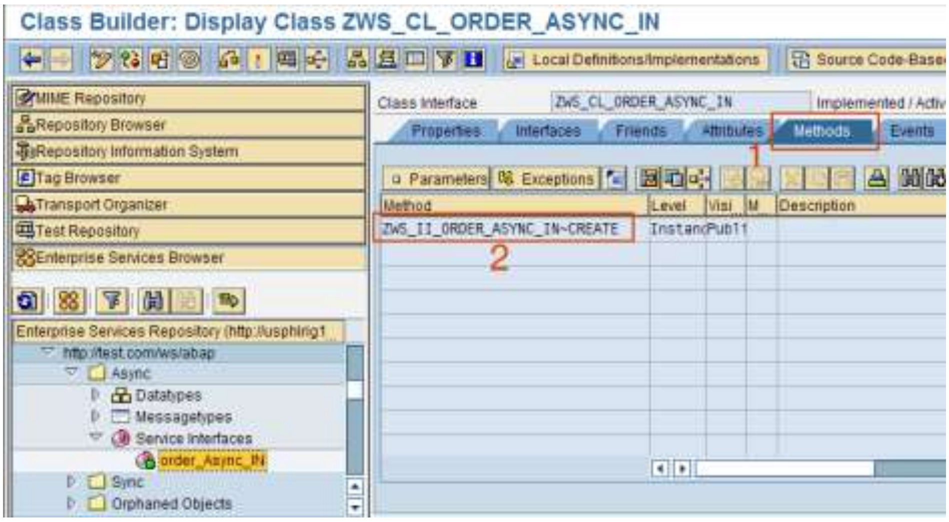
External Key
Type: Service Interface State: not released
Name: order_Async_IN
Namespace: http://test.com/ws/abap
Description:
Direction: Inbound

Proxy
Proxy Name: ZWS_II_ORDER_ASYNC_IN Prefix: ZWS_
Description: Proxy Interface (generated)

Interface
Provider Class: ZWS_CL_ORDER_ASYNC_IN
Description: Proxy Class (generated)
WebService Definition: Zorder_Async_IN

General Data
Package: $TMP
Original Language: EN English
Created by: LIWI on 18.09.2009 18:09:48
Changed by: LIWI on 18.09.2009 19:09:04

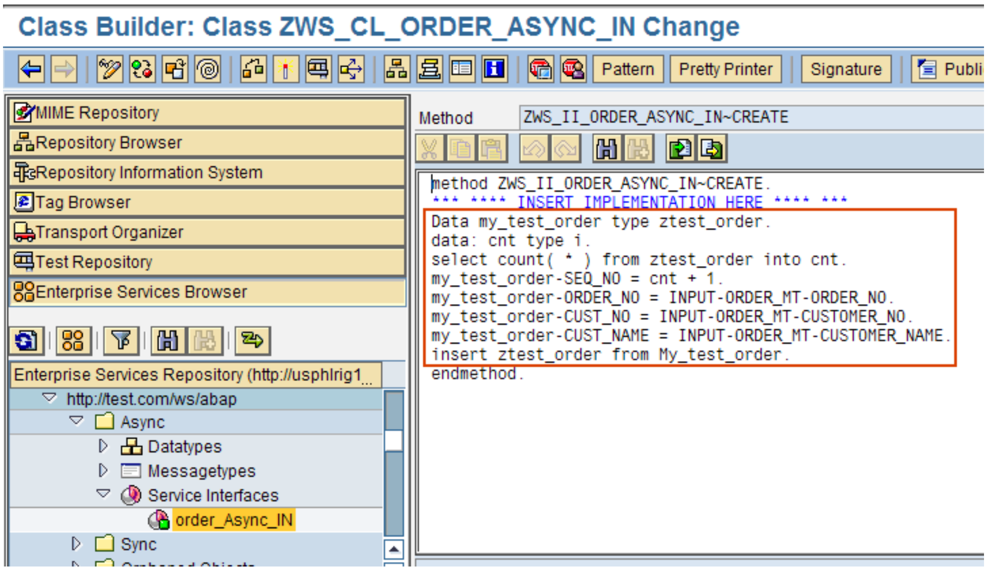
   b. Select the tab “Method” and double-click on the method name:



Class Builder: Display Class ZWS_CL_ORDER_ASYNC_IN

Method	Level	Visi	M	Description
ZWS_II_ORDER_ASYNC_IN~CREATE	Instanc	Pub	11	

c. Insert code into the proxy:



Class Builder: Class ZWS_CL_ORDER_ASYNC_IN Change

Method ZWS_II_ORDER_ASYNC_IN~CREATE

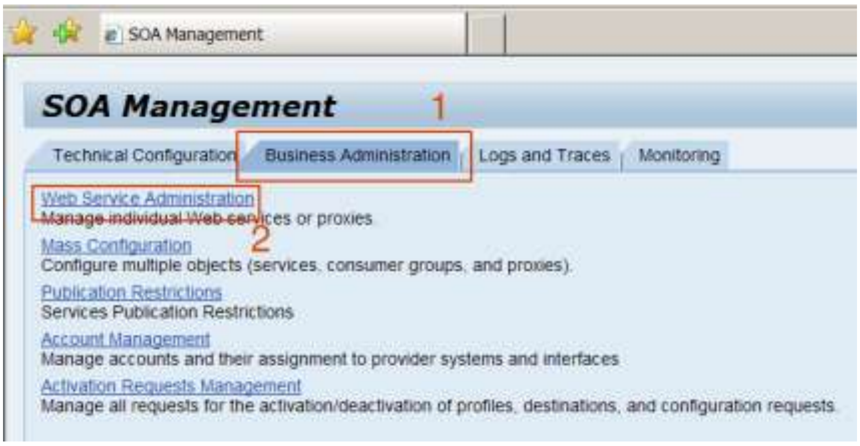
```
method ZWS_II_ORDER_ASYNC_IN~CREATE.
*** **** INSERT IMPLEMENTATION HERE **** ***
Data my_test_order type ztest_order.
data: cnt type i.
select count( * ) from ztest_order into cnt.
my_test_order-SEQ_NO = cnt + 1.
my_test_order-ORDER_NO = INPUT-ORDER_MT-ORDER_NO.
my_test_order-CUST_NO = INPUT-ORDER_MT-CUSTOMER_NO.
my_test_order-CUST_NAME = INPUT-ORDER_MT-CUSTOMER_NAME.
insert ztest_order from My_test_order.
endmethod.
```

d. Save and activate the proxy.

4. Use SOAMANAGER to create a web service from the ABAP proxy.

   a. On the application client of the provider system, enter transaction SOAMANAGER. This will start a webdynpro application on the browser.

   b. In the browser SOA Management application, select the tab “Business Administration” and click on “Web Service Administration”:



SOA Management

Web Service Administration
Manage individual Web services or proxies

Mass Configuration
Configure multiple objects (services, consumer groups, and proxies)

Publication Restrictions
Services Publication Restrictions

Account Management
Manage accounts and their assignment to provider systems and interfaces

Activation Requests Management
Manage all requests for the activation/deactivation of profiles, destinations, and configuration requests

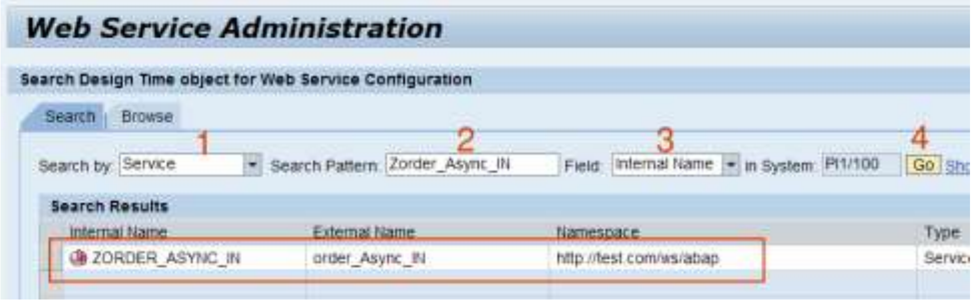
c. Enter the following search criteria:

Search by: Service

Search Pattern: Zorder_Async_IN (this value can be obtained from SPROXY)

Field: Internal Name

The will result in the proxy being displayed:

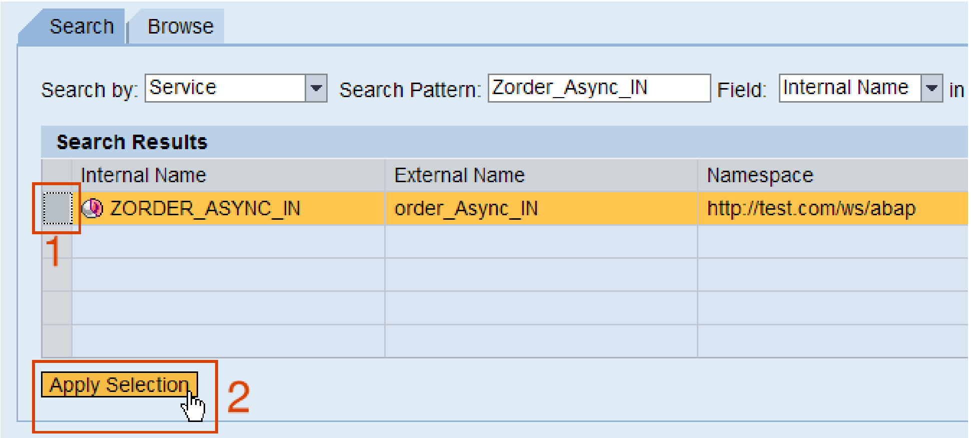

Web Service Administration
Search Design Time object for Web Service Configuration
Search
Browse
1
2
3
4
Go
Sho

Search by: Service   Search Pattern: Zorder_Async_IN   Field: Internal Name   in System: PI1/100

Search Results

Internal Name	External Name	Namespace	Type
ZORDER_ASYNC_IN	order_Async_IN	http://test.com/ws/abap	Servic

d. Select the proxy name and click on “Apply Selection”:

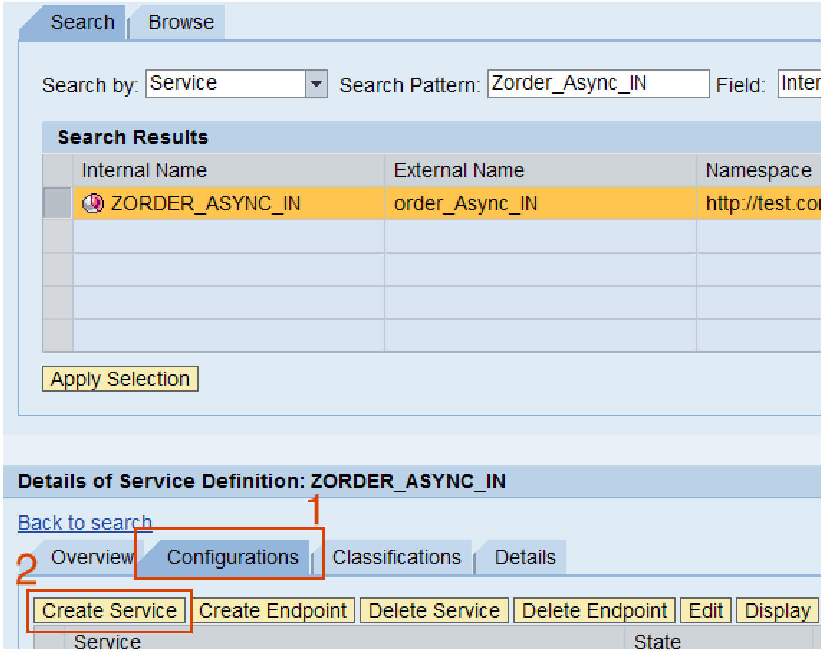


Search by: Service   Search Pattern: Zorder_Async_IN   Field: Internal Name

Search Results

Internal Name	External Name	Namespace
ZORDER_ASYNC_IN	order_Async_IN	http://test.com/ws/abap

e. On the bottom screen, select the tab “Configurations”, and click on “Create Service”:



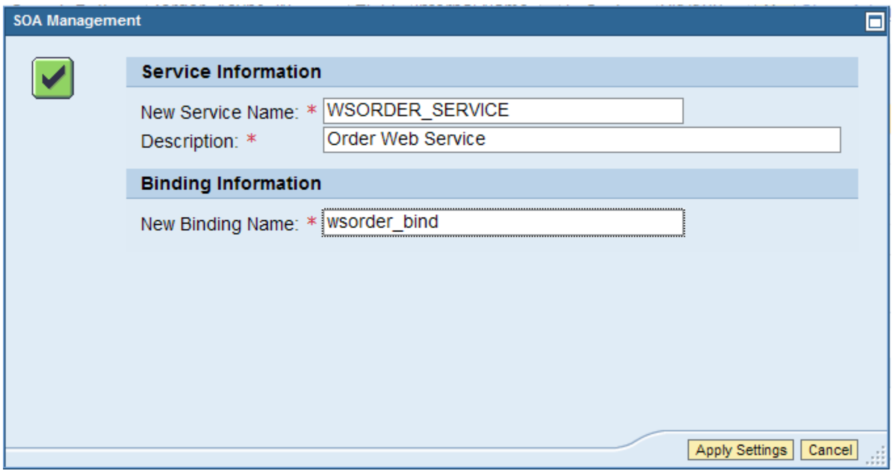
Search by: Service   Search Pattern: Zorder_Async_IN   Field: Inter

Search Results

Internal Name	External Name	Namespace
ZORDER_ASYNC_IN	order_Async_IN	http://test.co

Details of Service Definition: ZORDER_ASYNC_IN

f. Enter service name and binding name:



Service Information

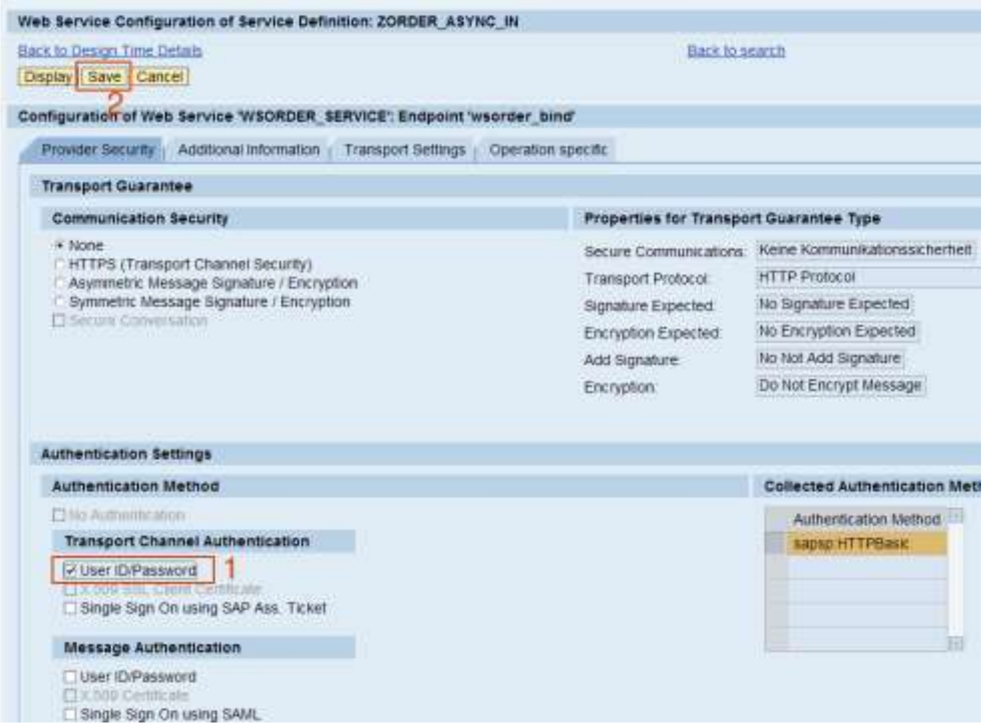
New Service Name: * WSORDER_SERVICE

Description: * Order Web Service

Binding Information

New Binding Name: * wsorder_bind

g. Scroll down to the lower screen and select “User ID/Password”, and “Save”:


Web Service Configuration of Service Definition: ZORDER_ASYNC_IN

Configuration of Web Service 'WSORDER_SERVICE': Endpoint 'wsorder_bind'

Transport Guarantee

Communication Security

● None
○ HTTPS (Transport Channel Security)
○ Asymmetric Message Signature / Encryption
○ Symmetric Message Signature / Encryption
□ Secure Conversation

Properties for Transport Guarantee Type

Secure Communications:	Keine Kommunikationssicherheit
Transport Protocol:	HTTP Protocol
Signature Expected:	No Signature Expected
Encryption Expected:	No Encryption Expected
Add Signature:	No Not Add Signature
Encryption:	Do Not Encrypt Message

Authentication Settings

Authentication Method

□ No Authentication

Transport Channel Authentication

☑ User ID/Password
□ X.509 SSL Client Certificate
□ Single Sign On using SAP Ass. Ticket

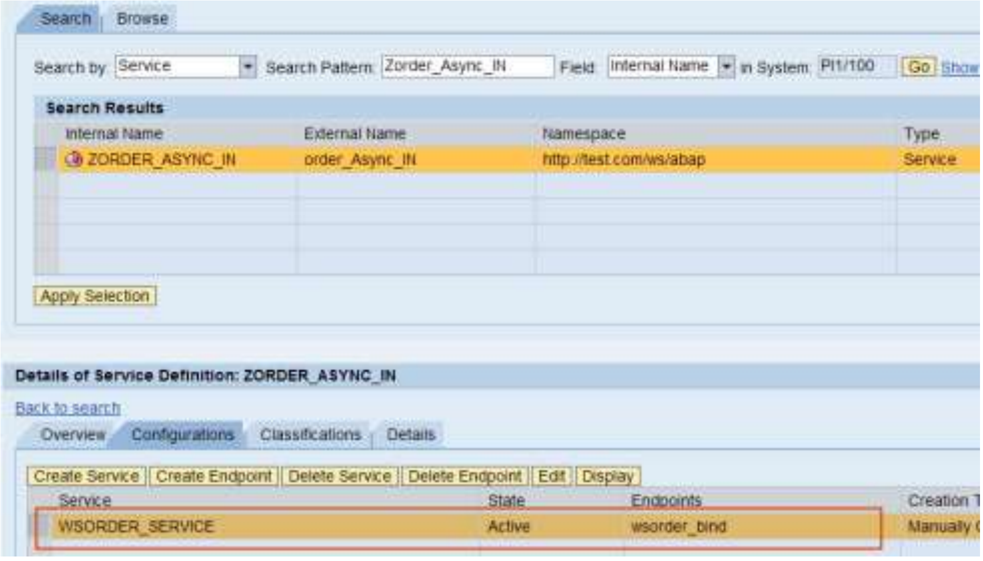
Message Authentication

□ User ID/Password
□ X.509 Certificate
□ Single Sign On using SAML

Collected Authentication Meth

Authentication Method
sapsp:HTTPBasic

h. The new service will now be visible:



Search by: Service Search Pattern: Zorder_Async_IN Field: Internal Name in System: PI1/100

Search Results

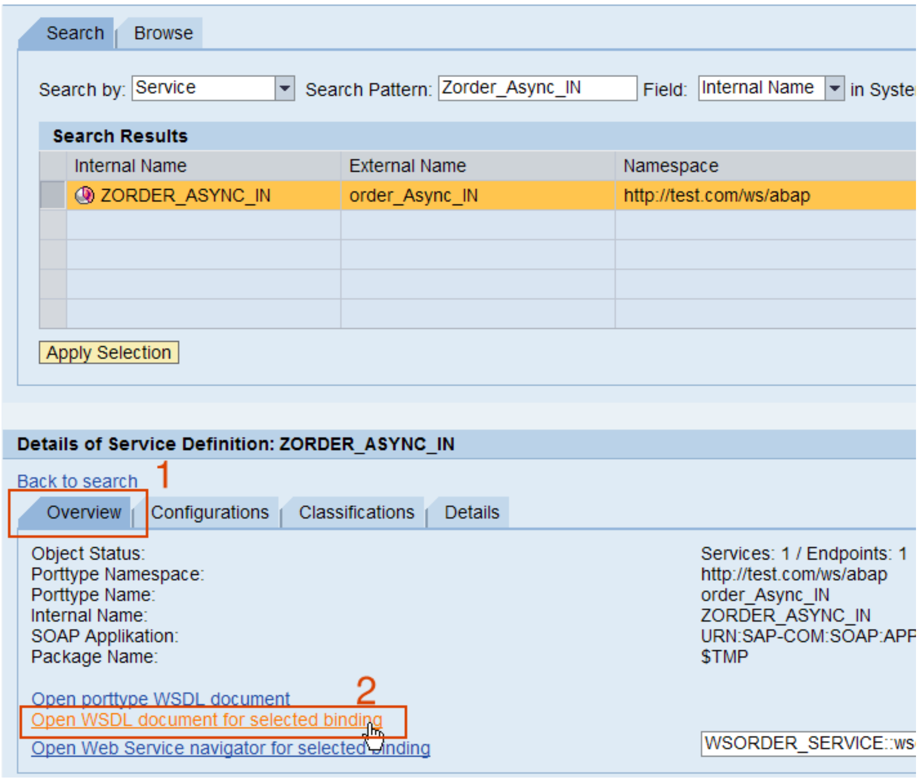
Internal Name	External Name	Namespace	Type
ZORDER_ASYNC_IN	order_Async_IN	http://test.com/ws/abap	Service

Details of Service Definition: ZORDER_ASYNC_IN

Service	State	Endpoints	Creation T
WSORDER_SERVICE	Active	wsorder_bind	Manually (

The web service for the provider proxy is now ready for use.

i. To obtain the URL for the web service, select the tab “Overview” and click on “Open WSDL document…”:


Search
Browse
Apply Selection
Back to search
1
Overview
Configurations
Classifications
Details
2
Open porttype WSDL document
Open WSDL document for selected binding
Open Web Service navigator for selected binding

Search by: Service Search Pattern: Zorder_Async_IN Field: Internal Name in Syste

Search Results

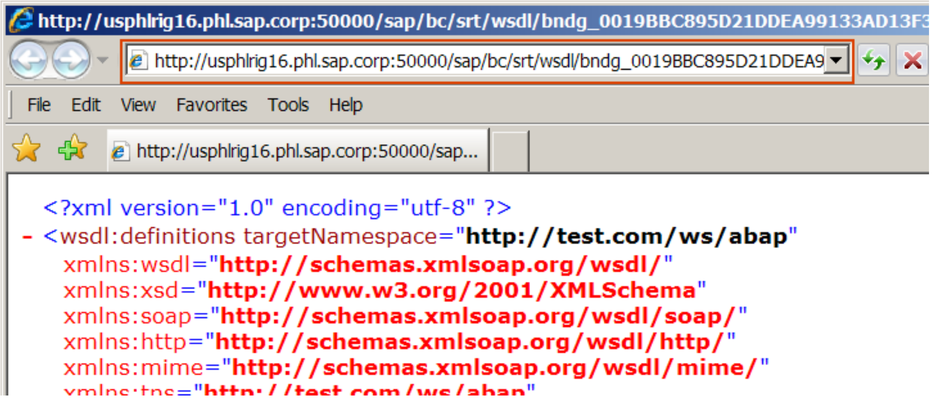
Internal Name	External Name	Namespace
ZORDER_ASYNC_IN	order_Async_IN	http://test.com/ws/abap

Details of Service Definition: ZORDER_ASYNC_IN

Object Status:	Services: 1 / Endpoints: 1
Porttype Namespace:	http://test.com/ws/abap
Porttype Name:	order_Async_IN
Internal Name:	ZORDER_ASYNC_IN
SOAP Applikation:	URN:SAP-COM:SOAP:APP
Package Name:	$TMP

WSORDER_SERVICE::ws

In our case, the following URL will be displayed on the browser:



http://usphlrig16.phl.sap.corp:50000/sap/bc/srt/wsdl/bndg_0019BBC895D21DDEA99133AD13F3
http://usphlrig16.phl.sap.corp:50000/sap/bc/srt/wsdl/bndg_0019BBC895D21DDEA9
File Edit View Favorites Tools Help
http://usphlrig16.phl.sap.corp:50000/sap...

```
  <?xml version="1.0" encoding="utf-8" ?>
- <wsdl:definitions targetNamespace="http://test.com/ws/abap"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
    xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
    xmlns:tns="http://test.com/ws/abap"
```

Tip

You can copy-n-paste this URL to a notepad. This URL will be used to create the consumer proxy. We will also use this URL to test the web service in the following step.

5. Test the web service using WSNavigator.

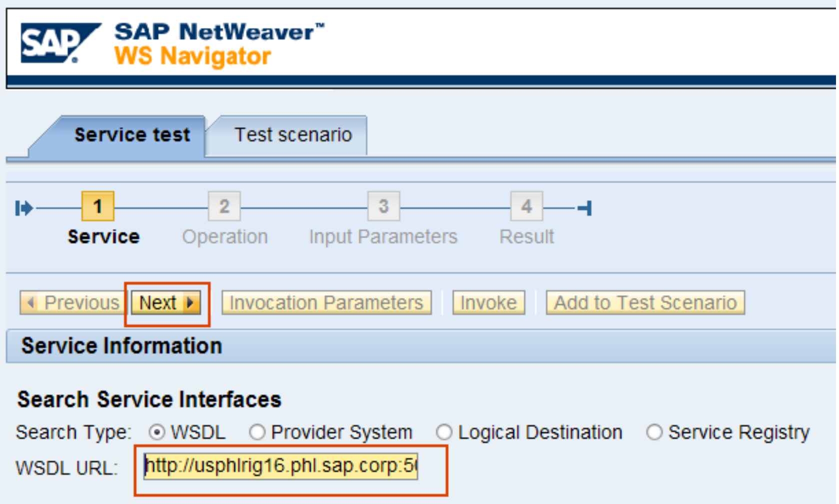
   To test the web service, we can use any WS-RM enabled client tool. WSNavigator on a NW 7.11 system or above can be used.

   Note

   WSNavigator on NW 7.10 or below cannot execute services using WS-RM. This capability is only available since PI 7.11.

   a. On a NW 7.11 system, execute WSNavigator.

   b. Enter the URL obtained from the previous step and click "Next":



Service Information

Search Service Interfaces

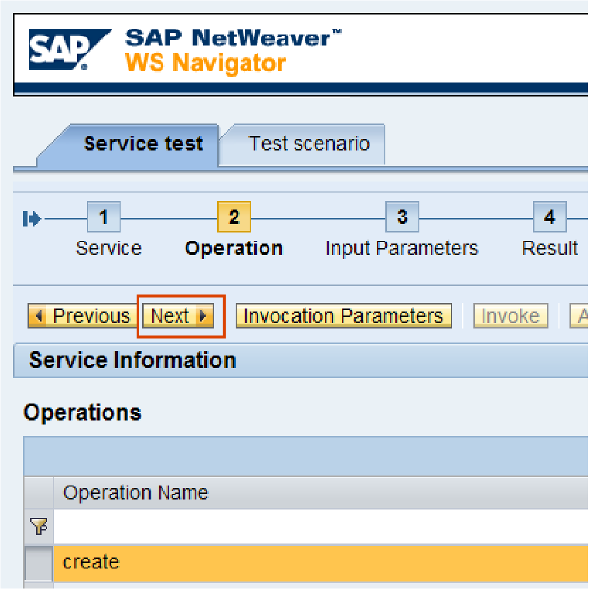
Search Type: ● WSDL ○ Provider System ○ Logical Destination ○ Service Registry

WSDL URL: http://usphlrig16.phl.sap.corp:5(

Tip

Logon will be requested and you will need to enter a valid user with authorization to execute the web service on the provider system.

   c. Click "Next":

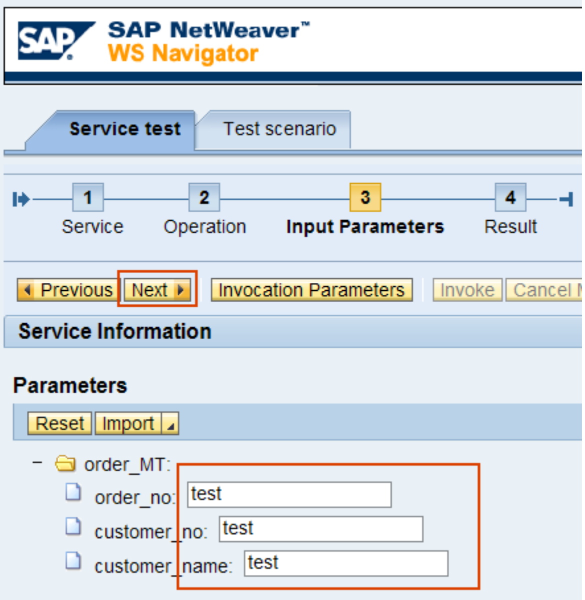


Service Information

Operations

Operation Name
create

d. Enter any test values and click “Next”:

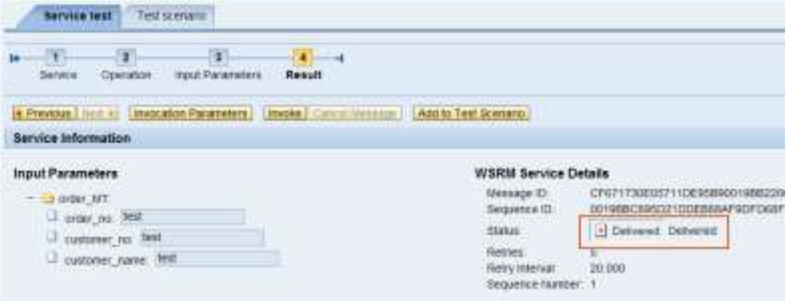


Service Information

Parameters

- order_MT:
  - order_no: test
  - customer_no: test
  - customer_name: test

e. The status will show “Delivering”. Once complete, it will show “Delivered”:



Service Information

Input Parameters

- order_MT
  - order_no: test
  - customer_no: test
  - customer_name: test

WSRM Service Details

Message ID: CF671730E05711DE958900198B229

Sequence ID: 00196BC8B5D21DDEB8AF9DFD6BF

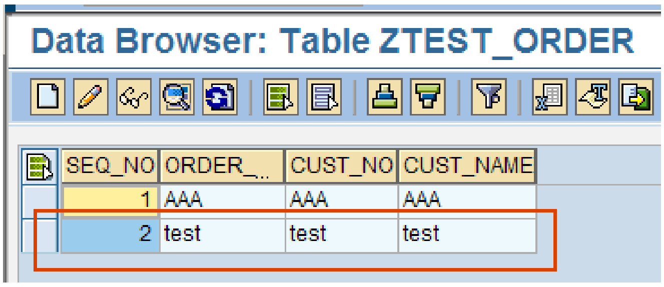
Status: Delivered Delivered

Retries: 5

Retry Interval: 20.000

Sequence number: 1

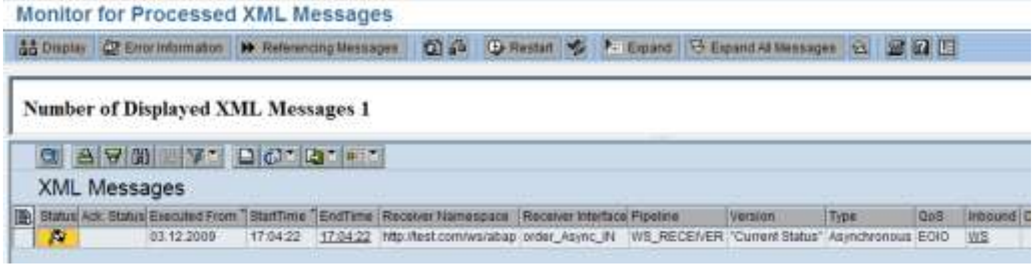
f. We can verify the success by checking the table content:



Data Browser: Table ZTEST_ORDER

SEQ_NO	ORDER_...	CUST_NO	CUST_NAME
1	AAA	AAA	AAA
2	test	test	test

We can also verify the message delivery using SXMB_MONI on the provider system:



Number of Displayed XML Messages 1

XML Messages

Status	Ack. Status	Executed From	StartTime	EndTime	Receiver Namespace	Receiver Interface	Pipeline	Version	Type	QoS	Inbound
		03.12.2009	17:04:22	17:04:22	http://test.com/ws/abap	order_Async_IN	WS_RECEIVER	"Current Status"	Asynchronous	EOIO	WS

4.2 Create the Consumer Application

1. Create consumer proxy using the URL of the provider web service.

   Note

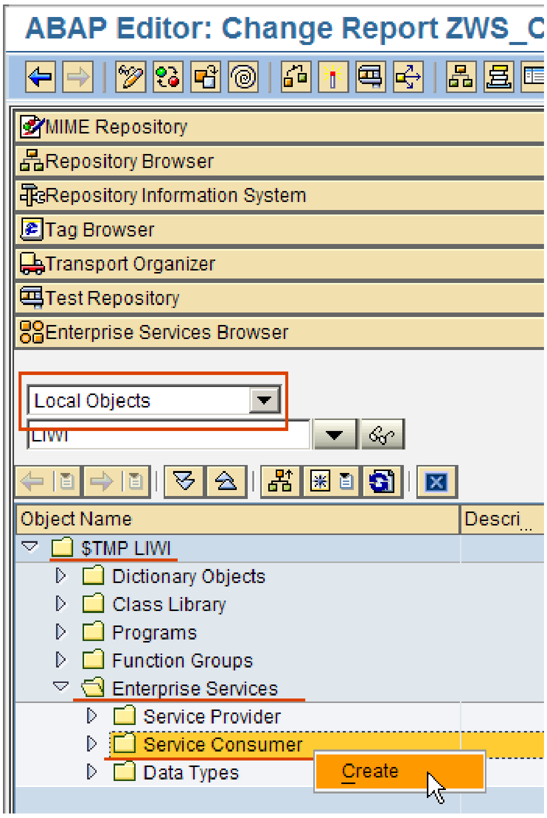
   We will need the URL saved in a previous step, in section 4.1.4.i.

   a. Logon to the consumer system, execute transaction SE80.

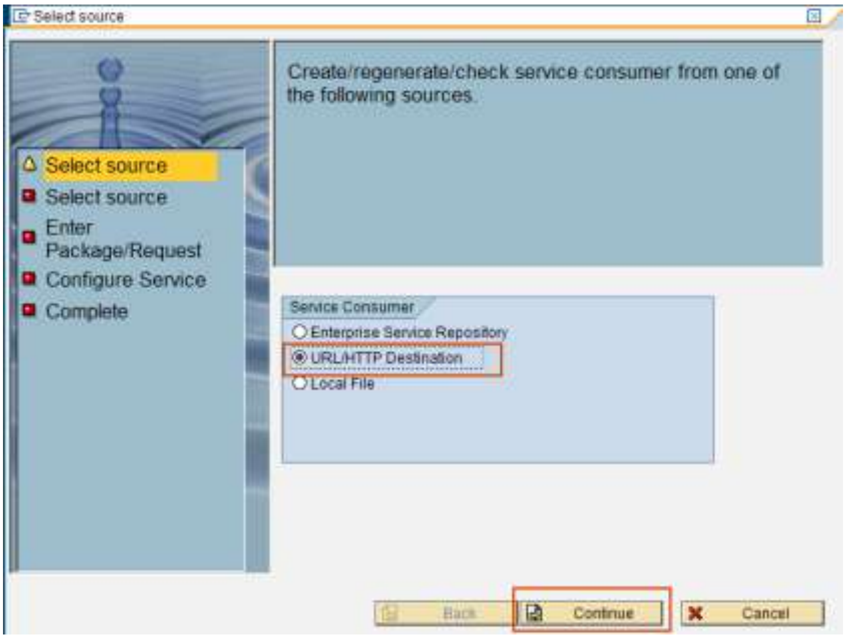
   Important

   The consumer system must also have WS-RM configured. If not, then see section 4.1.

   b. Select “Local Objects” and navigate to “Service Consumer”. Right-click and select “Create”:



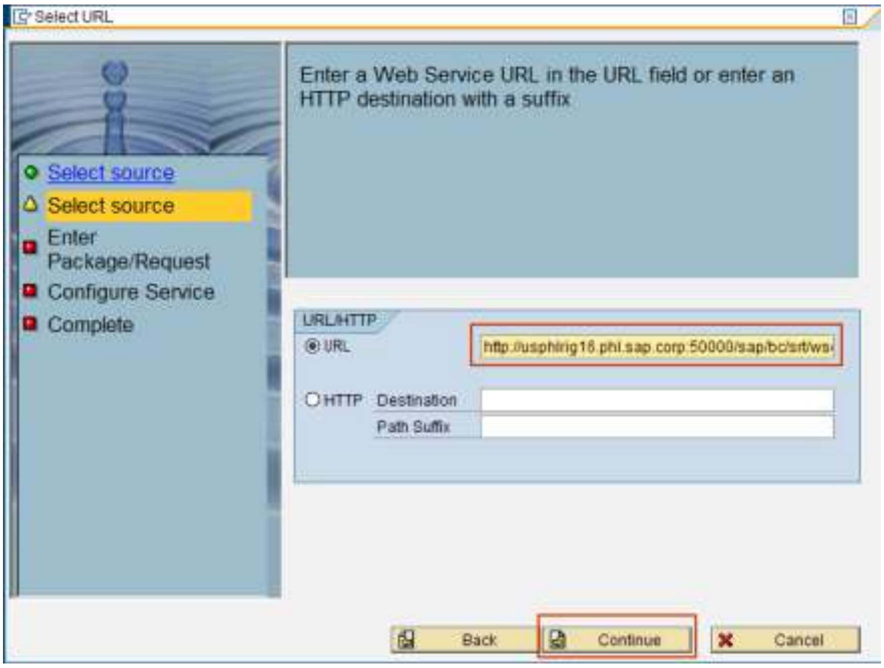
c. Select “URL/HTTP Destination” and click “Continue”:



Create/regenerate/check service consumer from one of the following sources.

Service Consumer
○ Enterprise Service Repository
● URL/HTTP Destination
○ Local File

d. Paste the URL obtain from the provider web service, and click “Continue”:



Select URL

Enter a Web Service URL in the URL field or enter an HTTP destination with a suffix

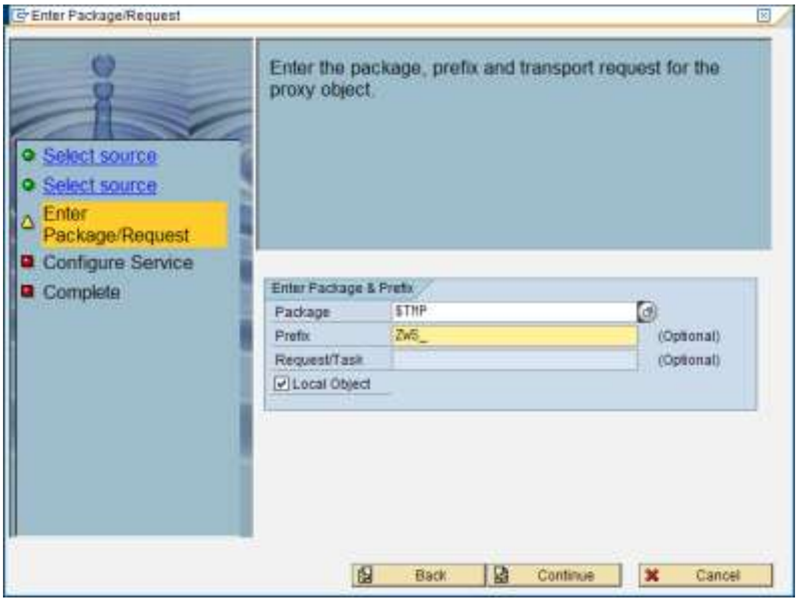
URL/HTTP

● URL http://usphlrig18.phl.sap.corp:50000/sap/bc/srt/ws

○ HTTP Destination

Path Suffix

e. Enter a prefix for the proxy object and click “Continue”:



Enter Package/Request

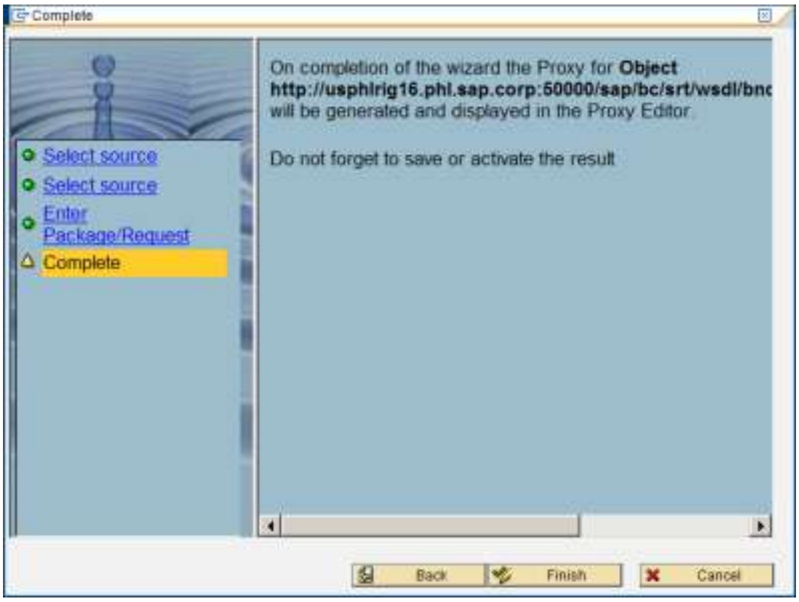
Enter the package, prefix and transport request for the proxy object.

Enter Package & Prefix

Package	$TMP	
Prefix	ZW5_	(Optional)
Request/Task		(Optional)

☑ Local Object

f. Click “Finish”:



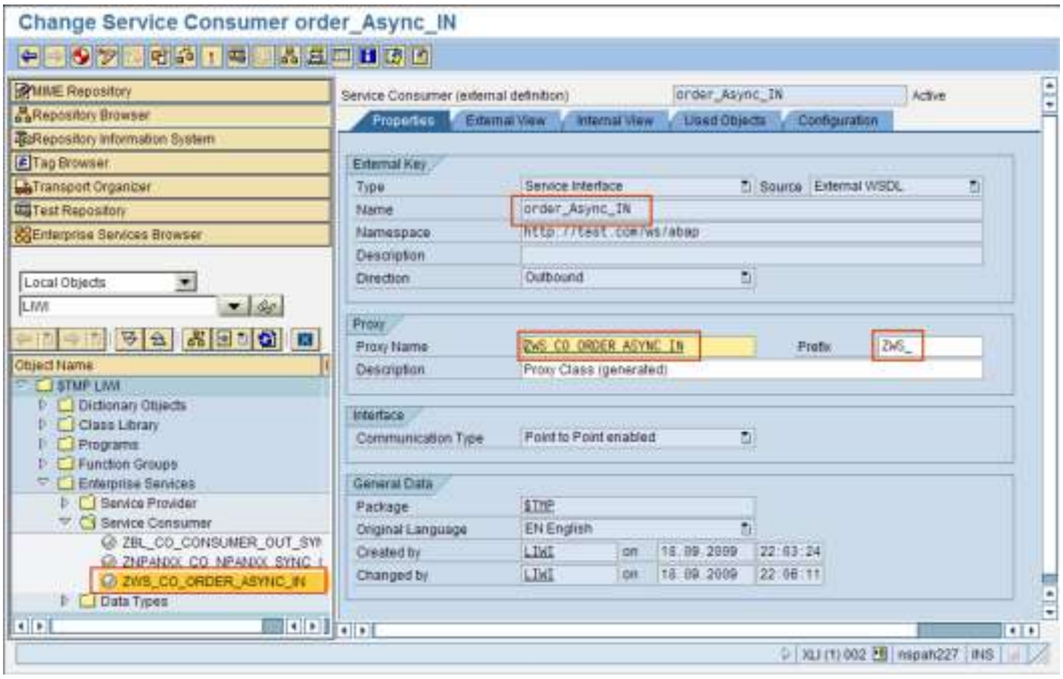
On completion of the wizard the Proxy for **Object http://usphlrig16.phl.sap.corp:50000/sap/bc/srt/wsdl/bnc** will be generated and displayed in the Proxy Editor.

Do not forget to save or activate the result

Important

You will be prompted for logon user to access the provider systems. This is necessary for the wizard to retrieve the metadata information of the web service.

g. The consumer proxy will be created:



Change Service Consumer order_Async_IN

Service Consumer (external definition): order_Async_IN Active

External Key
- Type: Service Interface — Source: External WSDL
- Name: order_Async_IN
- Namespace: http://test.com/ws/abap
- Description:
- Direction: Outbound

Proxy
- Proxy Name: ZWS_CO_ORDER_ASYNC_IN — Prefix: ZWS_
- Description: Proxy Class (generated)

Interface
- Communication Type: Point to Point enabled

General Data
- Package: $TMP
- Original Language: EN English
- Created by: LIWI on 18.09.2009 22:03:24
- Changed by: LIWI on 18.09.2009 22:06:11

Save and activate the newly created consumer proxy.

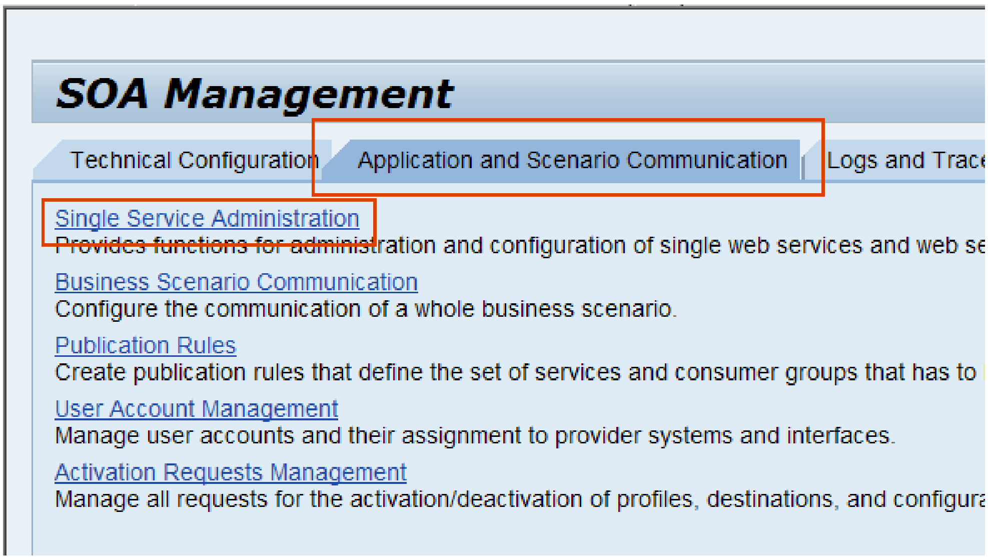
2. Use SOAMANAGER to create the consumer service from the proxy.

   a. Execute transaction SOAMANAGER on the consumer system.

   b. Go to the tab “Application and Scenario Communication” and click on “Single Service Administration”:

Note

The tab and function name might be different depending on the release level. However, the names should be similar, and position of the tab and location of the function should be the same. For example for PI 7.1, the tab is “Business Administration” and the function name is “Web Service Administration”.



SOA Management

Technical Configuration | Application and Scenario Communication | Logs and Trace

Single Service Administration
~~Provides functions for admini~~stration and configuration of single web services and web se

Business Scenario Communication
Configure the communication of a whole business scenario.

Publication Rules
Create publication rules that define the set of services and consumer groups that has to

User Account Management
Manage user accounts and their assignment to provider systems and interfaces.

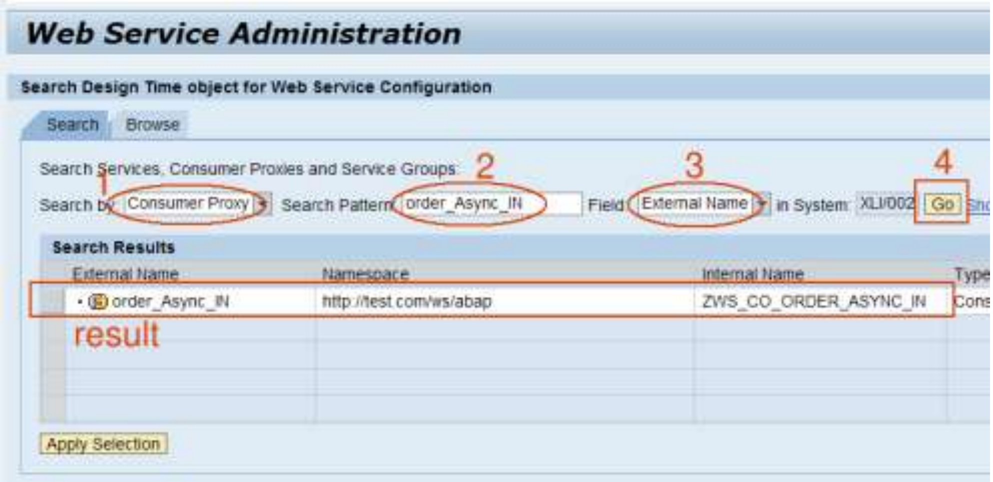
Activation Requests Management
Manage all requests for the activation/deactivation of profiles, destinations, and configura

c. In the Web Service Administration screen, select:

- Search by: Consumer Proxy
- Search Pattern: order_Async_IN (this value can be obtained from the previous step when the consumer proxy is created)
- Field: External Name

Click “Go” to search.

The following screen should result:



Web Service Administration

Search Design Time object for Web Service Configuration

Search | Browse

Search Services, Consumer Proxies and Service Groups:

Search by: Consumer Proxy Search Pattern: order_Async_IN Field: External Name in System: XLI/002 Go

Search Results

External Name	Namespace	Internal Name	Type
order_Async_IN	http://test.com/ws/abap	ZWS_CO_ORDER_ASYNC_IN	Cons

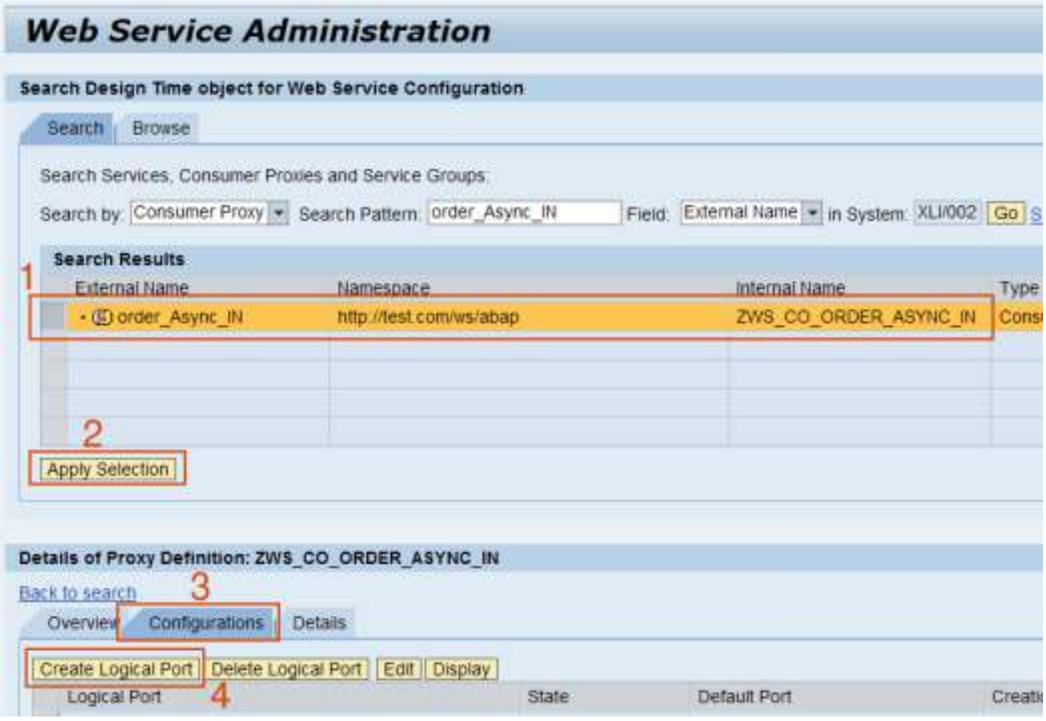
d. Create a port for the consumer service:

i. Select the consumer proxy

ii. Click on “Apply Selection”

iii. From the new screen on the bottom, go to the tab “Configuration”

iv. Click on “Create Logical Port”



Web Service Administration

Search Design Time object for Web Service Configuration

Search Services, Consumer Proxies and Service Groups:

Search by: Consumer Proxy Search Pattern: order_Async_IN Field: External Name in System: XLI/002 Go

Search Results

External Name	Namespace	Internal Name	Type
order_Async_IN	http://test.com/ws/abap	ZWS_CO_ORDER_ASYNC_IN	Cons

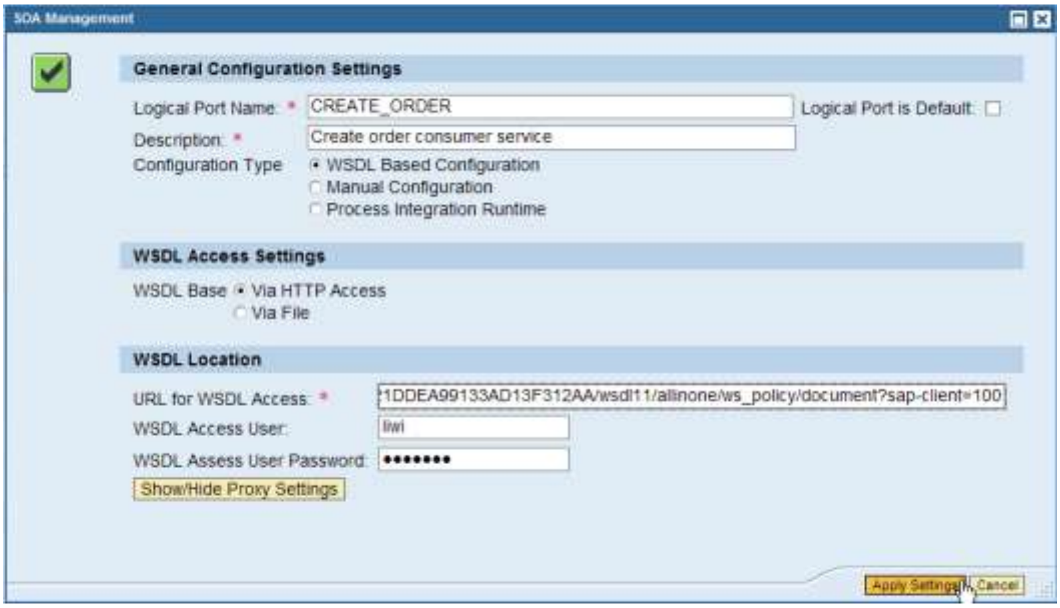
Details of Proxy Definition: ZWS_CO_ORDER_ASYNC_IN

e. Enter all the fields to create the logical port and click on “Apply Settings”:

The “URL for WSDL Access” is the provider’s web service URL from step 4.1.4.i.

Important

Depending on the NW release level, you may be asked to enter the binding name also. If this is requested, enter any binding name, e.g. “create_order_binding”.



SOA Management

General Configuration Settings

Logical Port Name: * CREATE_ORDER Logical Port is Default: □

Description: * Create order consumer service

Configuration Type: ● WSDL Based Configuration ○ Manual Configuration ○ Process Integration Runtime

WSDL Access Settings

WSDL Base: ● Via HTTP Access ○ Via File

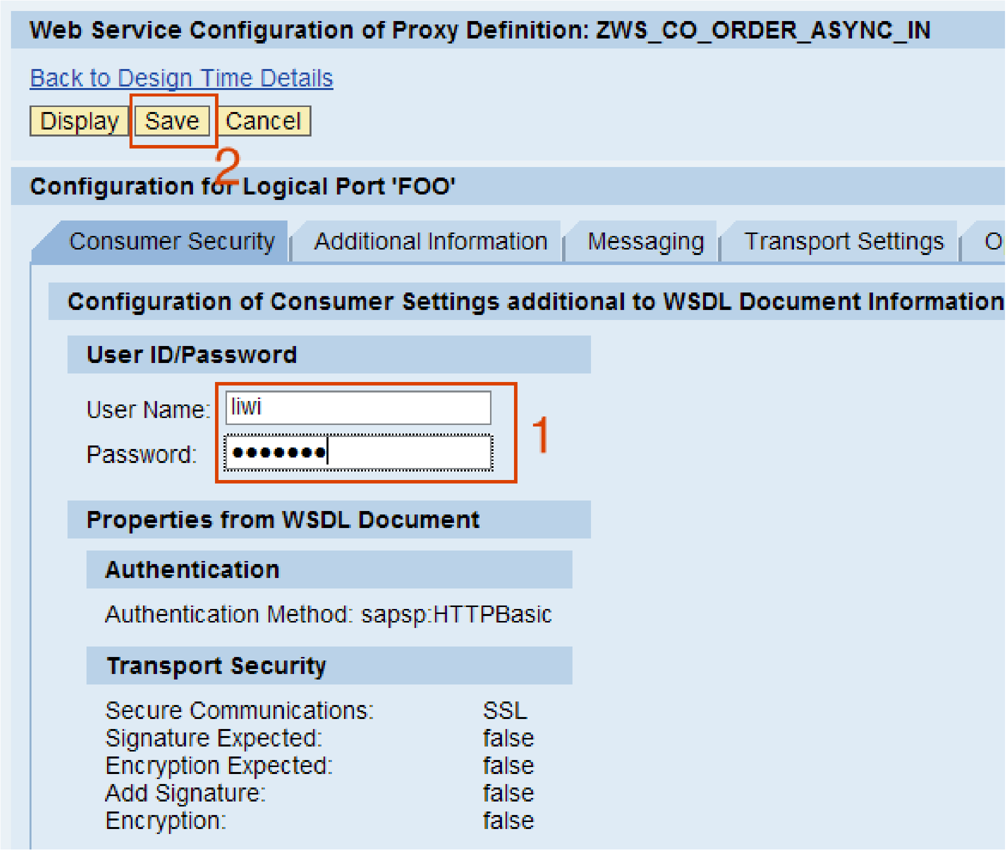
WSDL Location

URL for WSDL Access: * 1DDEA99133AD13F312AA/wsdl11/allinone/ws_policy/document?sap-client=100

WSDL Access User: liwi

WSDL Assess User Password: •••••••

f. On the newly created bottom portion of the screen, enter the authorized user to execute the provider service, and save the configuration:



Web Service Configuration of Proxy Definition: ZWS_CO_ORDER_ASYNC_IN

Back to Design Time Details

Configuration for Logical Port 'FOO'

Configuration of Consumer Settings additional to WSDL Document Information

User ID/Password

User Name: liwi

Password: •••••••

Properties from WSDL Document

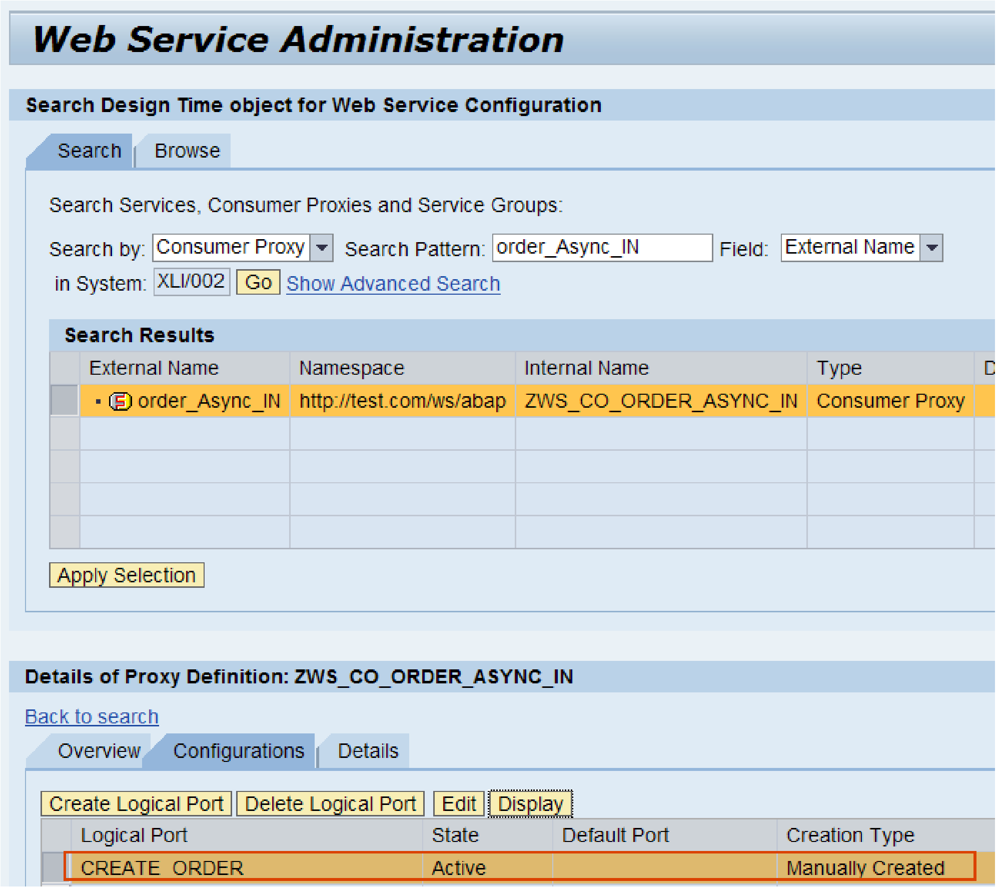
Authentication

Authentication Method: sapsp:HTTPBasic

Transport Security

Secure Communications:	SSL
Signature Expected:	false
Encryption Expected:	false
Add Signature:	false
Encryption:	false

g. You should now see the new port in the display:



Web Service Administration

Search Design Time object for Web Service Configuration

Search Services, Consumer Proxies and Service Groups:

Search by: Consumer Proxy Search Pattern: order_Async_IN Field: External Name

in System: XLI/002

Search Results

External Name	Namespace	Internal Name	Type	D
order_Async_IN	http://test.com/ws/abap	ZWS_CO_ORDER_ASYNC_IN	Consumer Proxy	

Details of Proxy Definition: ZWS_CO_ORDER_ASYNC_IN

Back to search

Logical Port	State	Default Port	Creation Type
CREATE_ORDER	Active		Manually Created

3. Create a program to call the consumer service.

   With a consumer web service, we have to have a program to call it. Below is an example of an ABAP program, created using SE38:

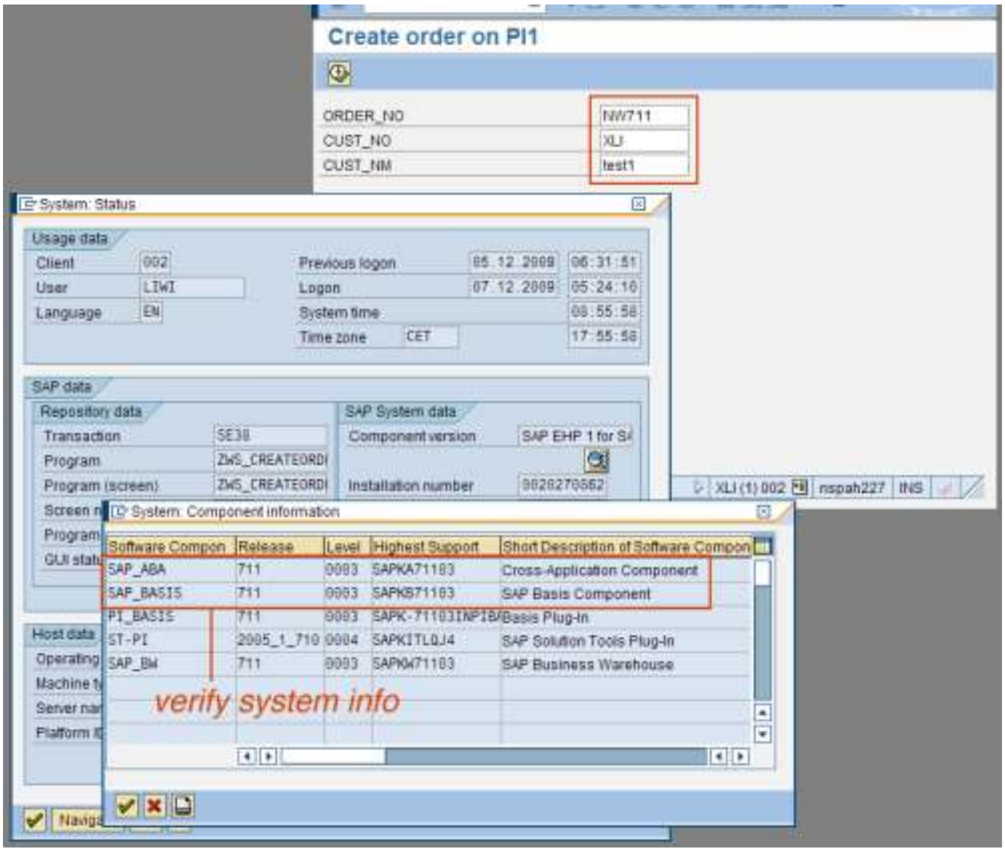
```
*&--------------------------------------------------------------------
-*
*& Report  ZWS_CREATEORDER_PI1
*&
*&--------------------------------------------------------------------
-*
*&
*&
*&--------------------------------------------------------------------
-*

REPORT  ZWS_CREATEORDER_PI1.

DATA: ZCREATEORDER TYPE REF TO ZWS_CO_ORDER_ASYNC_IN,
           m_seq_prot  TYPE ref to IF_WSPROTOCOL_SEQUENCE,
           m_seq  TYPE ref to IF_WS_CLIENT_SEQUENCE,
           l_wsprot  type ref to if_wsprotocol,
           lv_seq TYPE SRT_SEQ_ID.

data: INPUT type ZWS_ORDER_MT .


parameters: Order_No(10) type c LOWER CASE.
parameters: Cust_No(10) type c LOWER CASE.
parameters: Cust_Nm(10) type c LOWER CASE.

input-ORDER_MT-ORDER_NO = Order_No.
input-ORDER_MT-CUSTOMER_NO = Cust_No.
input-ORDER_MT-CUSTOMER_NAME = Cust_nm.

try.
    CREATE OBJECT ZCREATEORDER
      EXPORTING
        logical_port_name = 'CREATE_ORDER'.

*  generation of the sequence protocol and the sequence
    m_seq_prot ?= ZCREATEORDER-
>get_protocol( if_wsprotocol=>sequence ).
    m_seq = m_seq_prot->create_persistent_sequence( ).

*  start sequencing and get id
    m_seq->begin( ).
    m_seq_prot->set_client_sequence( m_seq ).
    lv_seq = m_seq->get_id( ).
    CALL METHOD ZCREATEORDER->CREATE
      EXPORTING
```

```
        INPUT = input.

*  end sequencing and commit work
    m_seq->end( ).
    cl_soap_tx_factory=>commit_work( ).
    write:/ 'Order Successfully Sent'.

  CATCH CX_AI_SYSTEM_FAULT
        CX_AI_APPLICATION_FAULT .
    write : 'Error during proxy call'.
    exit.
ENDTRY.
```

Please note the parts of the program marked in red.

- `ZWS_CO_ORDER_ASYNC_IN`: the proxy name as it was create in SE80 above.
- `ZWS_ORDER_MT`: the ABAP structure containing the input data. This is automatically created during the proxy creation process.
- `CREATE_ORDER`: logical port name created using SOAMANAGER is a previous step.
- `CREATE`: the operation name when we created the service interface in the ESR. You can also see this name in the ABAP proxy as the method of the class.
- `commit_work( )`: we must have this for asynchronous message processing.

4. Execute the consumer program:
   a. On a NW 7.11 consumer system:



Create order on PI1

ORDER_NO	NW711
CUST_NO	XLI
CUST_NM	test1

System: Status

Usage data

Client	002	Previous logon	05.12.2009	06:31:51
User	LIWI	Logon	07.12.2009	05:24:10
Language	EN	System time		08:55:58
		Time zone	CET	17:55:58

SAP data

Repository data

Transaction	SE38
Program	ZWS_CREATEORD
Program (screen)	ZWS_CREATEORD

SAP System data

Component version	SAP EHP 1 for S
Installation number	0020270562

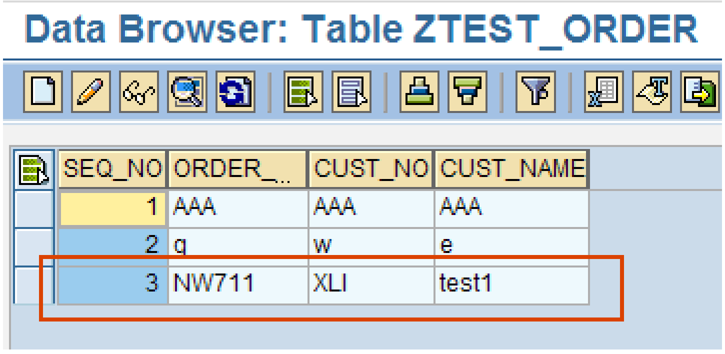
XLI (1) 002 nspah227 INS

System: Component information

Software Compon	Release	Level	Highest Support	Short Description of Software Compon
SAP_ABA	711	0003	SAPKA71103	Cross-Application Component
SAP_BASIS	711	0003	SAPKB71103	SAP Basis Component
PI_BASIS	711	0003	SAPK-71103INPIB	Basis Plug-In
ST-PI	2005_1_710	0004	SAPKITLQJ4	SAP Solution Tools Plug-In
SAP_BW	711	0003	SAPKW71103	SAP Business Warehouse

verify system info

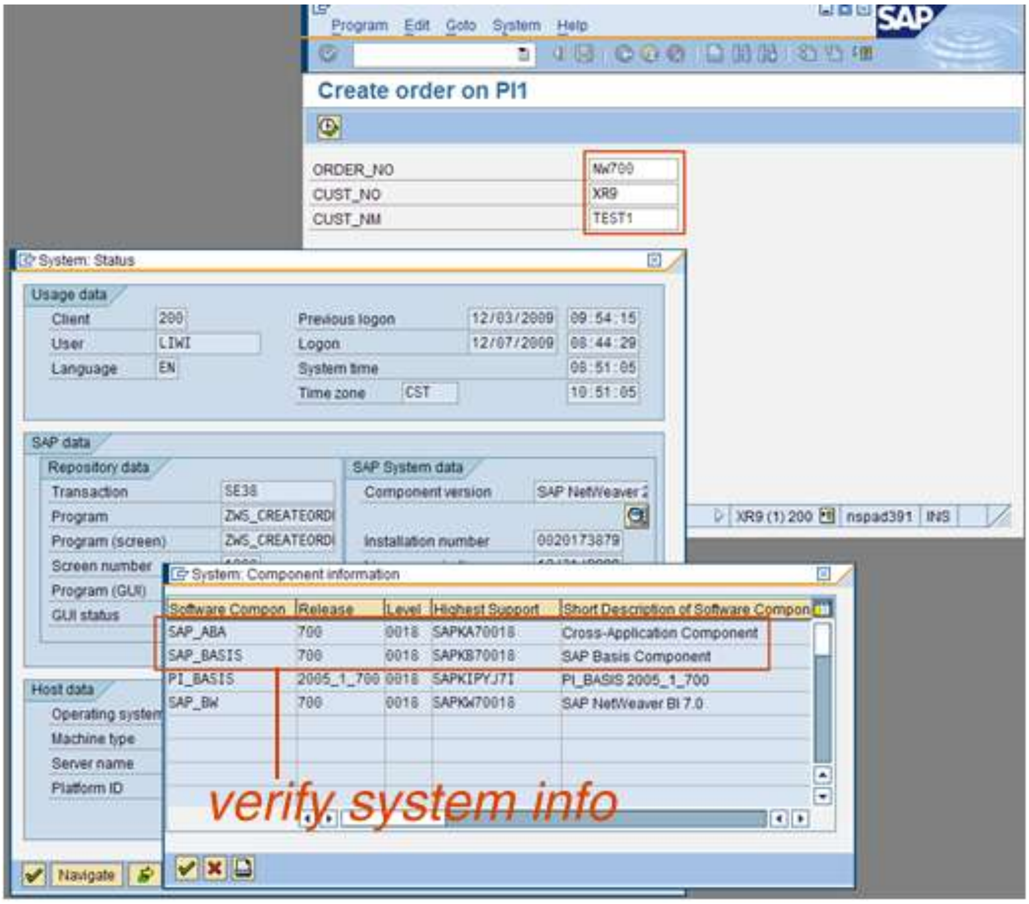
Result on the provider system:



Data Browser: Table ZTEST_ORDER

SEQ_NO	ORDER_...	CUST_NO	CUST_NAME
1	AAA	AAA	AAA
2	q	w	e
3	NW711	XLI	test1

b. On a NW 7.00 SP18 consumer system:



Create order on PI1

ORDER_NO	NW700
CUST_NO	XR9
CUST_NM	TEST1

System: Status

Usage data

Client	200	Previous logon	12/03/2009	09:54:15
User	LIWI	Logon	12/07/2009	08:44:29
Language	EN	System time		08:51:05
		Time zone	CST	10:51:05

SAP data

Repository data

Transaction	SE38
Program	ZWS_CREATEORD
Program (screen)	ZWS_CREATEORD
Screen number	
Program (GUI)	
GUI status	

SAP System data

Component version	SAP NetWeaver 2
Installation number	0020173879

Host data

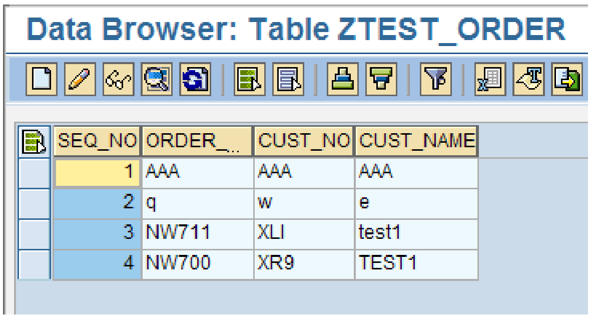
Operating system
Machine type
Server name
Platform ID

XR9 (1) 200 nspad391 INS

System: Component information

Software Compon	Release	Level	Highest Support	Short Description of Software Compon
SAP_ABA	700	0018	SAPKA70018	Cross-Application Component
SAP_BASIS	700	0018	SAPKB70018	SAP Basis Component
PI_BASIS	2005_1_700	0018	SAPKIPYJ7I	PI_BASIS 2005_1_700
SAP_BW	700	0018	SAPKW70018	SAP NetWeaver BI 7.0

Result on the provider system:



Data Browser: Table ZTEST_ORDER

SEQ_NO	ORDER_...	CUST_NO	CUST_NAME
1	AAA	AAA	AAA
2	q	w	e
3	NW711	XLI	test1
4	NW700	XR9	TEST1

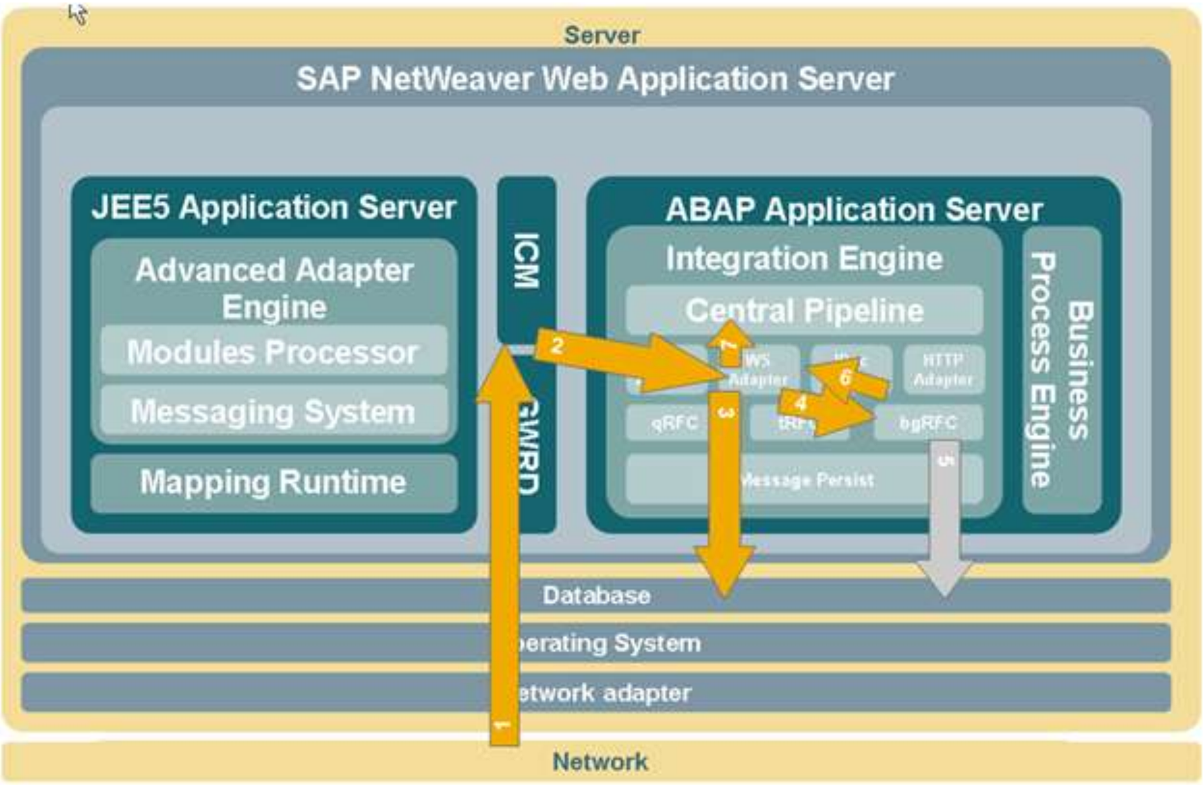
4.3 Monitoring and Debugging

WS-RM has its own runtime time engine. Consequently, monitoring and debugging will include a set of different tools in addition to the traditional ABAP and Integration Engine tools.

In this section, we will describe some of the available tools.

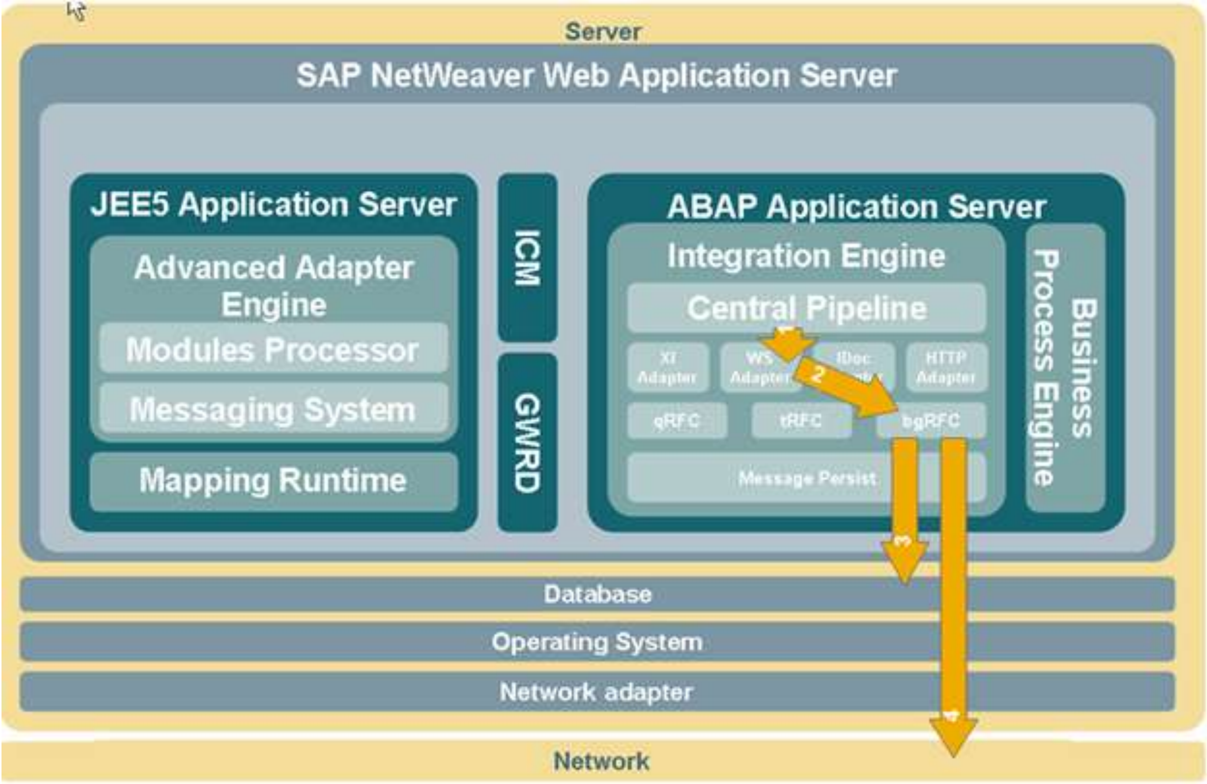
In order to determine the tools, an overview of the message flow during WS-RM runtime can be helpful. Although the diagrams below outlines the flow in PI Integration Servier, it is also applicable to application servers since the Integration Engine is also part of the ABAP stack.

Sending message from the network to WS-RM:



Network → ICM → WS Adapter → bgRFC → WS Adapter → Central Pipeline

Sending message from WS-RM to the network:



Central Pipeline → WS Adapter → bgRFC → Network (via HTTP Post)

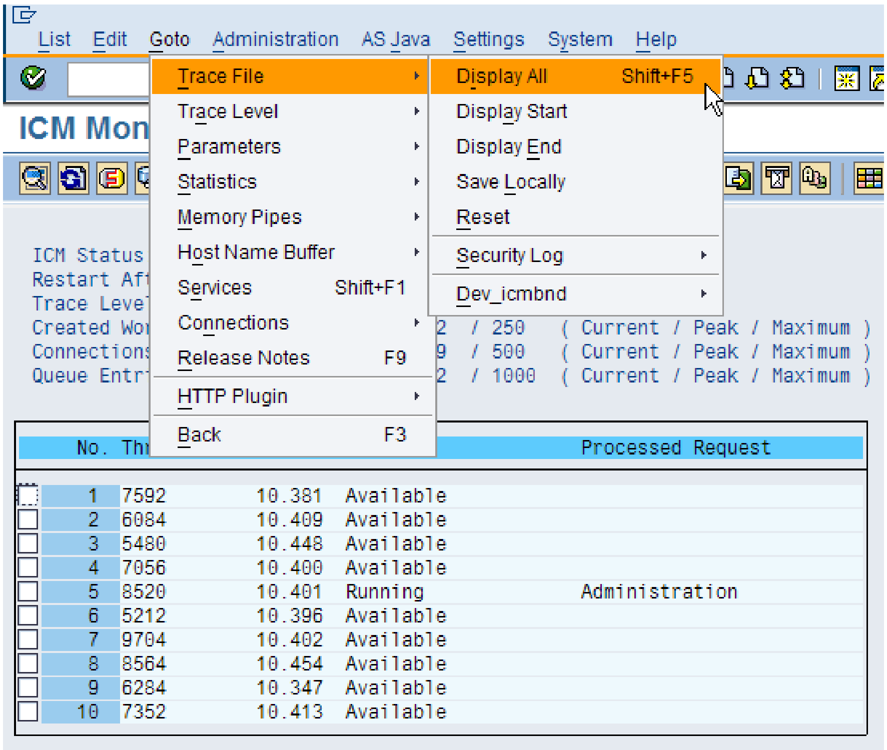
Therefore, during our monitoring process, we can use the message processing paths indicated above to determine the tools to use.

1. ICM: monitoring the ICM for incoming messages.

   ICM can be monitored from either ABAP or Java:

   - ABAP: transaction SMICM

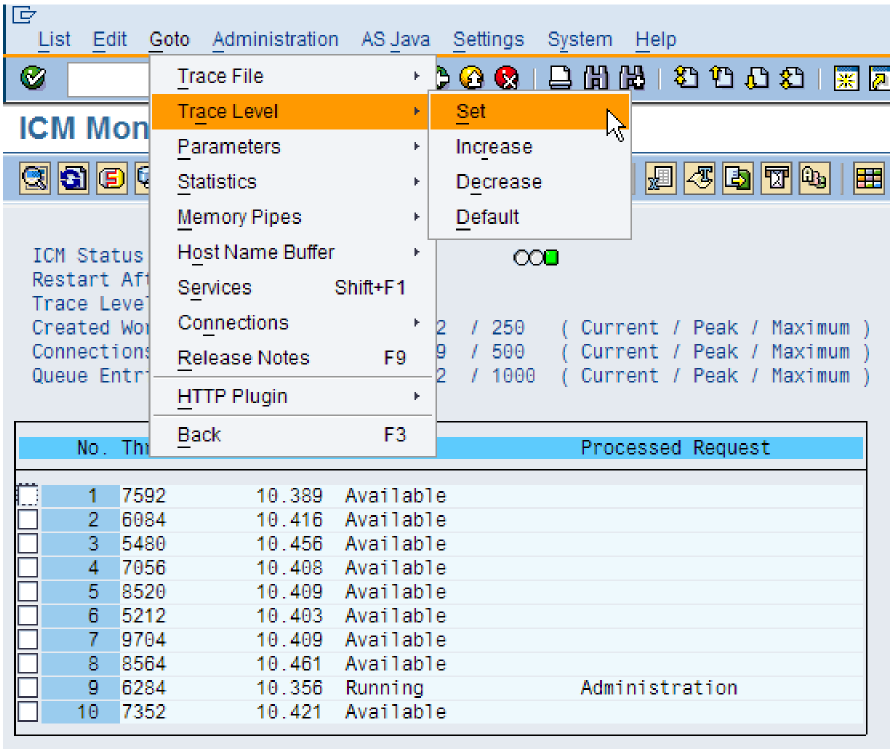
     From the main menu: Goto → Trace File → Display…



```
ICM Status
Restart Aft
Trace Leve
Created Wo    2  / 250   ( Current / Peak / Maximum )
Connections   9  / 500   ( Current / Peak / Maximum )
Queue Entr    2  / 1000  ( Current / Peak / Maximum )
```

No.	Thr			Processed Request
1	7592	10.381	Available	
2	6084	10.409	Available	
3	5480	10.448	Available	
4	7056	10.400	Available	
5	8520	10.401	Running	Administration
6	5212	10.396	Available	
7	9704	10.402	Available	
8	8564	10.454	Available	
9	6284	10.347	Available	
10	7352	10.413	Available	

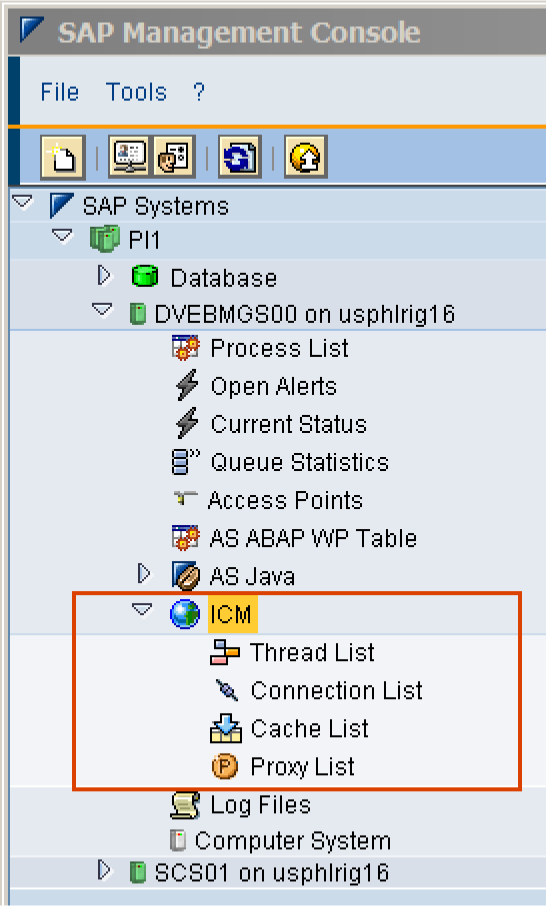
The trace level can be changed using the menu:
Goto → Trace Level → …



```
ICM Status
Restart Aft
Trace Leve
Created Wo    2  / 250   ( Current / Peak / Maximum )
Connections   9  / 500   ( Current / Peak / Maximum )
Queue Entr    2  / 1000  ( Current / Peak / Maximum )
```

No.	Thr			Processed Request
1	7592	10.389	Available	
2	6084	10.416	Available	
3	5480	10.456	Available	
4	7056	10.408	Available	
5	8520	10.409	Available	
6	5212	10.403	Available	
7	9704	10.409	Available	
8	8564	10.461	Available	
9	6284	10.356	Running	Administration
10	7352	10.421	Available	

- Java: http://<server>:5xx13 (Management Console)

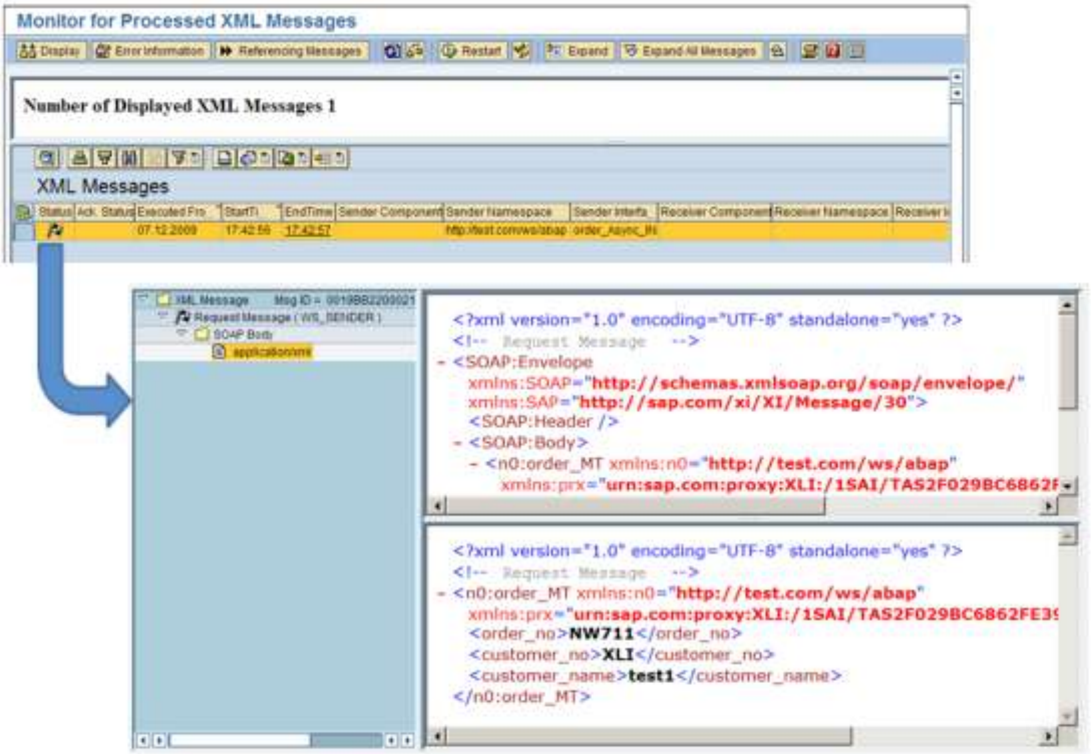


Note

Please reference the following for the creation of administrator users with authorization to view ICM from the browser:

http://help.sap.com/saphelp_nw70ehp1/helpdata/en/48/7f2d997df935e1e10000000a42189c/frameset.htm

2. Transaction SXMB_MONI or SXI_MONITOR: This is the existing tool used to monitor Integration Engine messages. From this display, we can get payload information.



Monitor for Processed XML Messages

Number of Displayed XML Messages 1

XML Messages

Status	Ack. Status	Executed Fro	StartTi	EndTime	Sender Component	Sender Namespace	Sender Interfa	Receiver Component	Receiver Namespace	Receiver I
		07.12.2009	17:42:56	17:42:57		http://test.com/ws/abap	order_Async_IN			

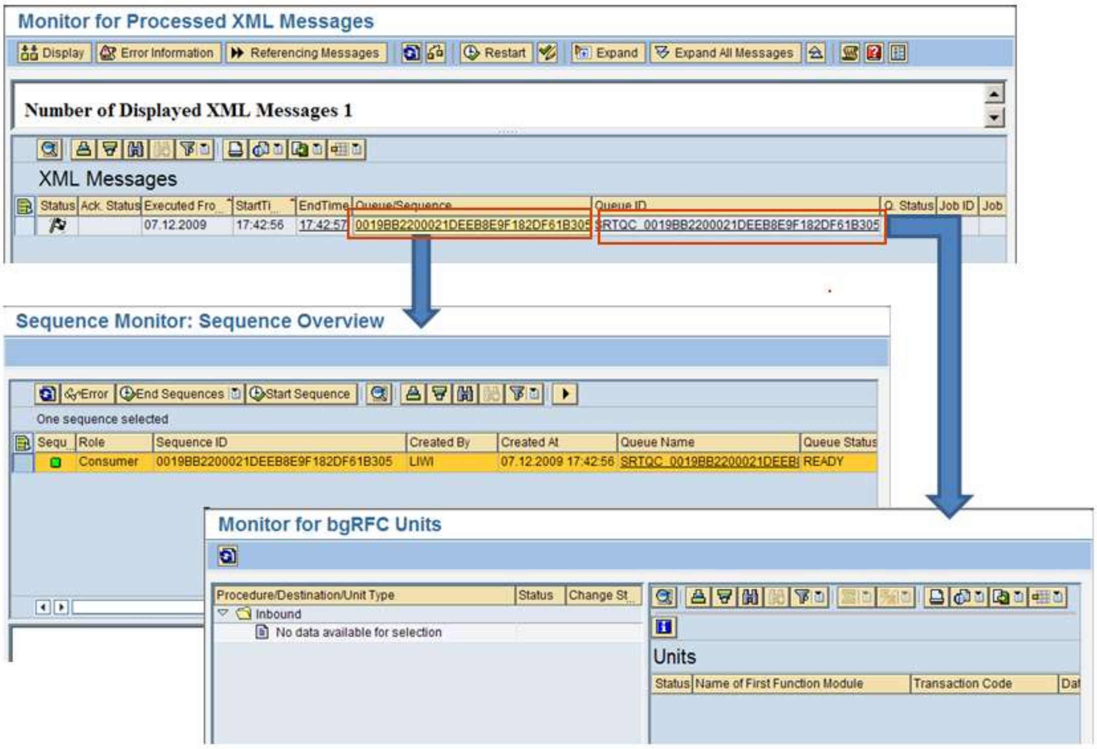
```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!-- Request Message -->
- <SOAP:Envelope
    xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:SAP="http://sap.com/xi/XI/Message/30">
    <SOAP:Header />
  - <SOAP:Body>
    - <n0:order_MT xmlns:n0="http://test.com/ws/abap"
        xmlns:prx="urn:sap.com:proxy:XLI:/1SAI/TAS2F029BC6862F
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!-- Request Message -->
- <n0:order_MT xmlns:n0="http://test.com/ws/abap"
    xmlns:prx="urn:sap.com:proxy:XLI:/1SAI/TAS2F029BC6862FE3
    <order_no>NW711</order_no>
    <customer_no>XLI</customer_no>
    <customer_name>test1</customer_name>
  </n0:order_MT>
```

3. With SXMB_MONI or SXI_MONITOR, scroll to the right to display the columns “Queue Sequence” and “Queue ID”. By clicking on the value of those columns will can get additional monitoring information:

   WS-RM messages are sent as sequences, where multiple messages can be sent in specific sequences with guaranteed delivery. SOAMANGER’s sequence monitor provides information on errors during message processing. The messages can be restarted and terminated using the monitor.

   WS-RM is processed by the background RFC, therefore, monitoring the bgRFC is important to get any additional details. The logging of successful asynchronous WS-RM messages is not kept, which is why for successful messages, we do not see anything in the monitor.



Monitor for Processed XML Messages

Number of Displayed XML Messages 1

XML Messages

Status	Ack. Status	Executed Fro...	StartTi...	EndTime	Queue/Sequence	Queue ID	Q. Status	Job ID	Job
🏁		07.12.2009	17:42:56	17:42:57	0019BB2200021DEEB8E9F182DF61B305	SRTQC_0019BB2200021DEEB8E9F182DF61B305			

Sequence Monitor: Sequence Overview

One sequence selected

Sequ...	Role	Sequence ID	Created By	Created At	Queue Name	Queue Status
🟩	Consumer	0019BB2200021DEEB8E9F182DF61B305	LIWI	07.12.2009 17:42:56	SRTQC_0019BB2200021DEEB	READY

Monitor for bgRFC Units

Procedure/Destination/Unit Type	Status	Change St...
Inbound		
No data available for selection		

Units

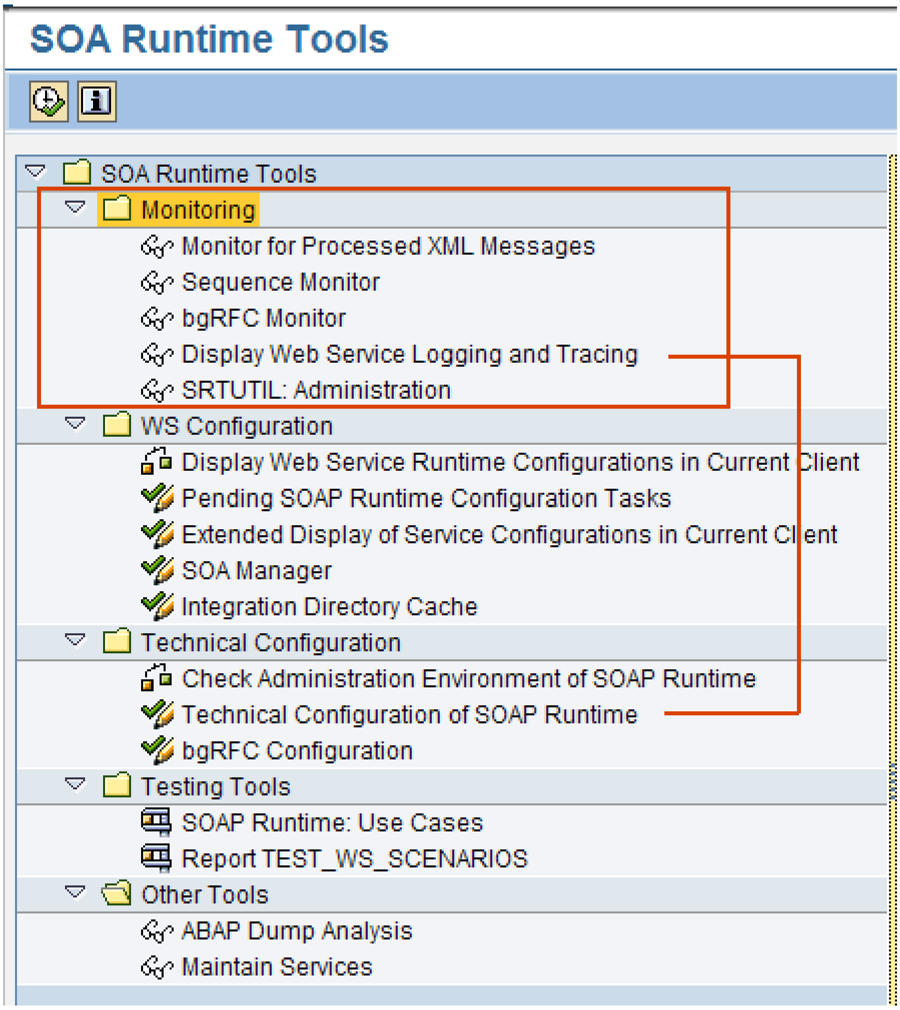
Status	Name of First Function Module	Transaction Code	Dat

Note

bgRFC monitoring is also accessible by using the transaction: SBGRFCMON

These 2 displays will give us additional information of the message being processed. When error does occur, the error details will be displayed.

4. Transaction SRT_TOOLS: This tool provides a centralized location for access to SOA runtime tools.



SOA Runtime Tools

- SOA Runtime Tools
  - Monitoring
    - Monitor for Processed XML Messages
    - Sequence Monitor
    - bgRFC Monitor
    - Display Web Service Logging and Tracing
    - SRTUTIL: Administration
  - WS Configuration
    - Display Web Service Runtime Configurations in Current Client
    - Pending SOAP Runtime Configuration Tasks
    - Extended Display of Service Configurations in Current Client
    - SOA Manager
    - Integration Directory Cache
  - Technical Configuration
    - Check Administration Environment of SOAP Runtime
    - Technical Configuration of SOAP Runtime
    - bgRFC Configuration
  - Testing Tools
    - SOAP Runtime: Use Cases
    - Report TEST_WS_SCENARIOS
  - Other Tools
    - ABAP Dump Analysis
    - Maintain Services

For monitoring:

a. Monitoring for Processed XML Messages: This is the same as the monitoring tool which can be accessed using SXMB_MONI.

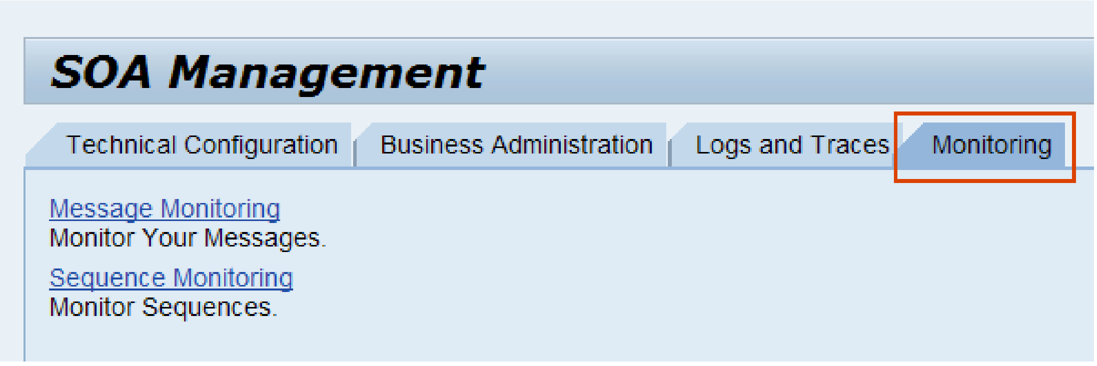
b. Sequence Monitor: This is the same tool discussed above.

c. bgRFC Monitor: This is the same tool discussed above.

d. Display Web Service Logging and Tracing: This tool monitors the messages and trace information as defined by the configurations in “Technical Configuration of SOAP Runtime”. The logging and trace level are configured there.

e. SRTUTIL: Display messages based on users.

5. Transaction SOAMANAGER →Monitoring: The SOAMANAGER tool is a webdynpro application which provides not only the administration tools of managing consumer and provider services, as seen in the previous sections, but it also provides monitoring, logs and traces.



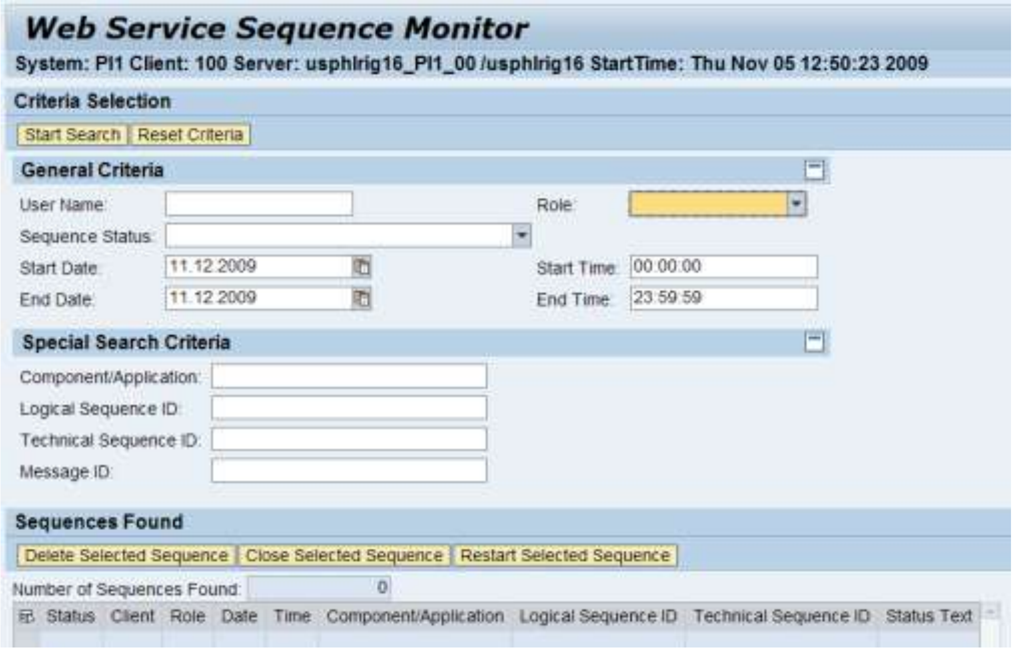
SOA Management

Message Monitoring
Monitor Your Messages.

Sequence Monitoring
Monitor Sequences.

a. Message Monitoring: This is the equivalent to SXMB_MONI.

b. Sequence Monitoring: This is also similar to the sequence monitoring in ABAP, as mentioned above. However, it provides a little different user interface.



Web Service Sequence Monitor

System: PI1 Client: 100 Server: usphlrig16_PI1_00 /usphlrig16 StartTime: Thu Nov 05 12:50:23 2009

Criteria Selection

General Criteria

User Name:		Role:	
Sequence Status:			
Start Date:	11.12.2009	Start Time:	00:00:00
End Date:	11.12.2009	End Time:	23:59:59

Special Search Criteria

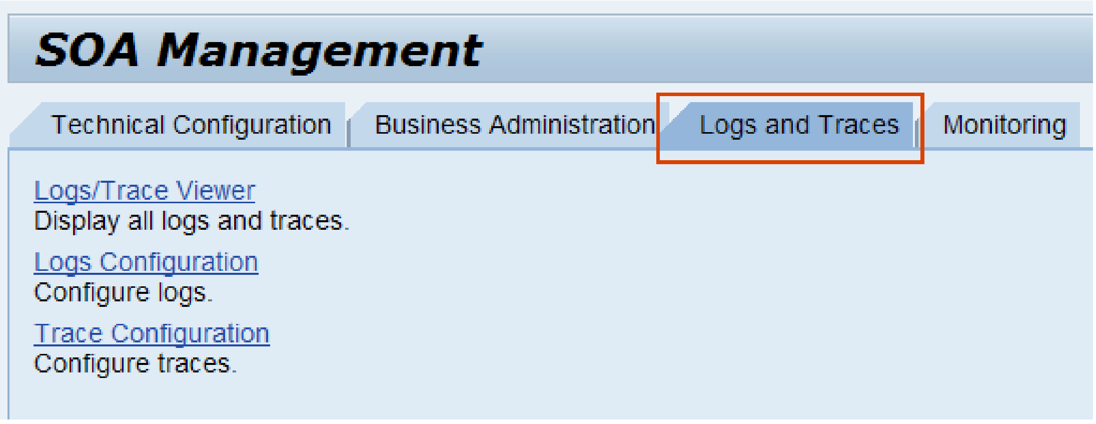
Component/Application:	
Logical Sequence ID:	
Technical Sequence ID:	
Message ID:	

Sequences Found

Number of Sequences Found: 0

Status	Client	Role	Date	Time	Component/Application	Logical Sequence ID	Technical Sequence ID	Status Text

6. Transaction SOAMANAGER → Logs and Traces:



SOA Management

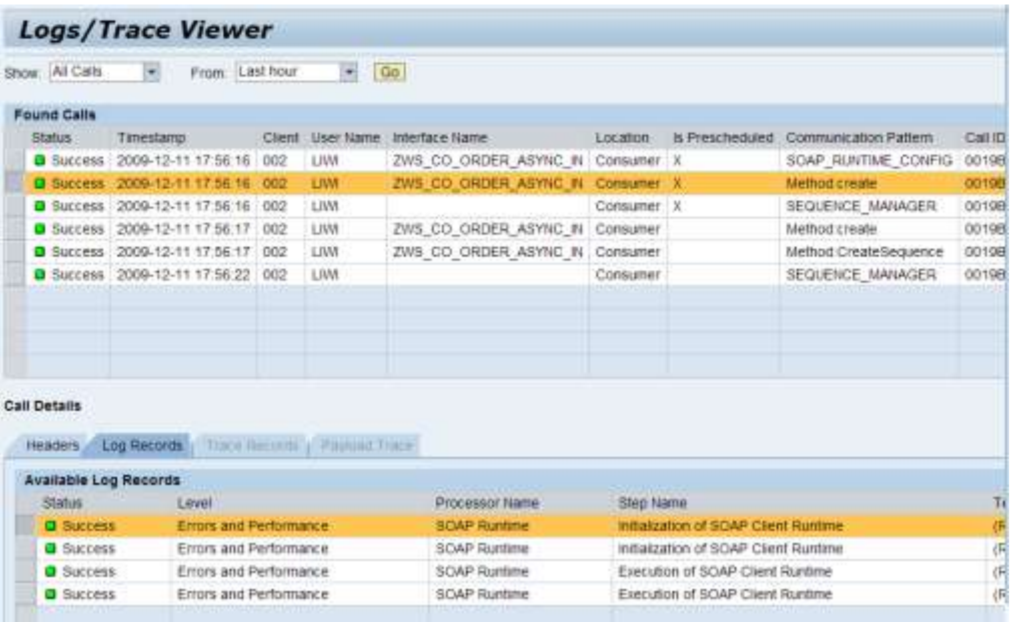
Logs/Trace Viewer
Display all logs and traces.

Logs Configuration
Configure logs.

Trace Configuration
Configure traces.

This provides detailed message information based on the configuration in “Logs Configuration” and “Trace Configuration”.

Example:



Logs/Trace Viewer

Show: All Calls From: Last hour Go

Found Calls

Status	Timestamp	Client	User Name	Interface Name	Location	Is Prescheduled	Communication Pattern	Call ID
🟩 Success	2009-12-11 17:56:16	002	LIWI	ZWS_CO_ORDER_ASYNC_IN	Consumer	X	SOAP_RUNTIME_CONFIG	0019B
🟩 Success	2009-12-11 17:56:16	002	LIWI	ZWS_CO_ORDER_ASYNC_IN	Consumer	X	Method create	0019B
🟩 Success	2009-12-11 17:56:16	002	LIWI		Consumer	X	SEQUENCE_MANAGER	0019B
🟩 Success	2009-12-11 17:56:17	002	LIWI	ZWS_CO_ORDER_ASYNC_IN	Consumer		Method create	0019B
🟩 Success	2009-12-11 17:56:17	002	LIWI	ZWS_CO_ORDER_ASYNC_IN	Consumer		Method CreateSequence	0019B
🟩 Success	2009-12-11 17:56:22	002	LIWI		Consumer		SEQUENCE_MANAGER	0019B

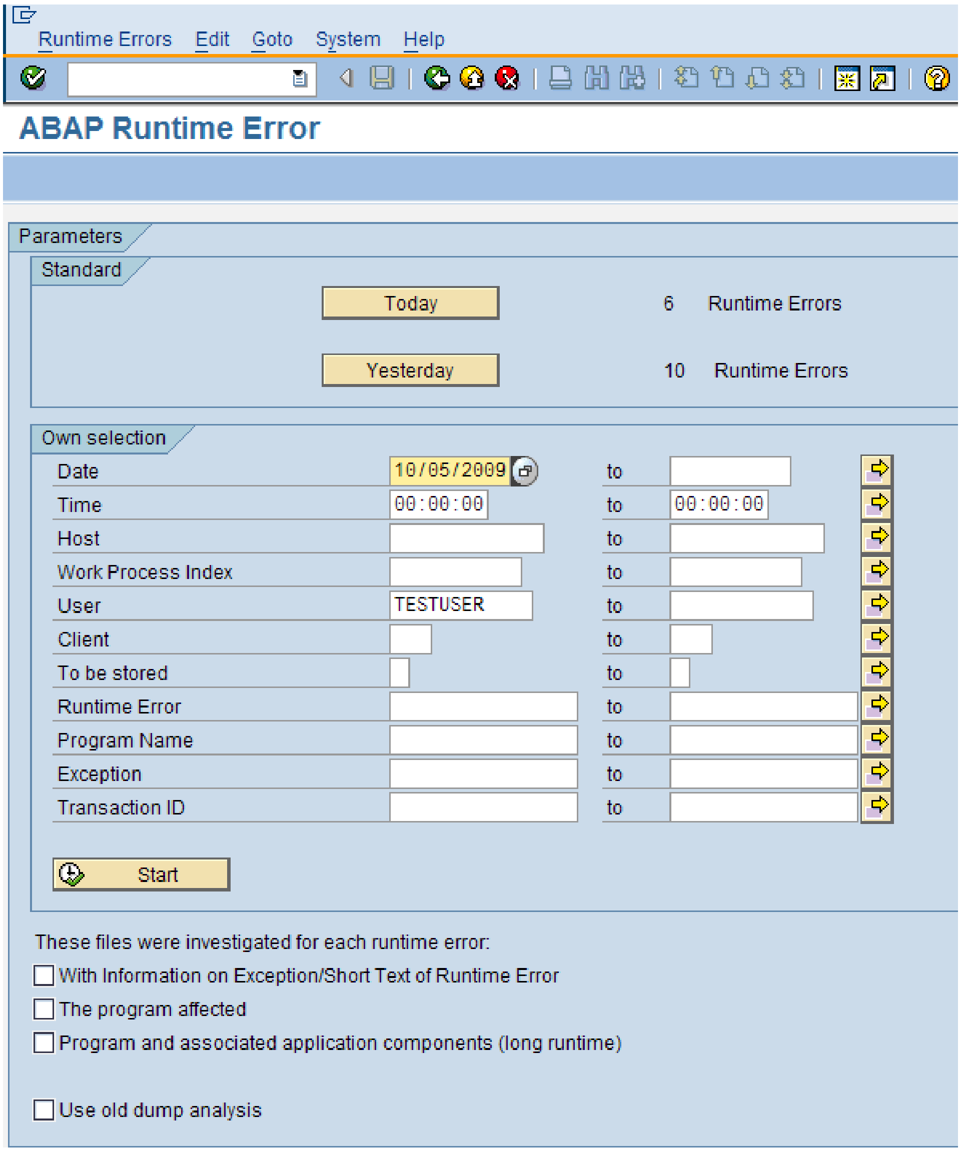
Call Details

Available Log Records

Status	Level	Processor Name	Step Name	Te
🟩 Success	Errors and Performance	SOAP Runtime	Initialization of SOAP Client Runtime	(F
🟩 Success	Errors and Performance	SOAP Runtime	Initialization of SOAP Client Runtime	(F
🟩 Success	Errors and Performance	SOAP Runtime	Execution of SOAP Client Runtime	(F
🟩 Success	Errors and Performance	SOAP Runtime	Execution of SOAP Client Runtime	(F

7. ABAP Runtime Error: It might also be necessary to go to the runtime error log to determine the problem.

   ABAP Transaction: ST22



6 Runtime Errors

10 Runtime Errors

Date	10/05/2009	to	
Time	00:00:00	to	00:00:00
Host		to	
Work Process Index		to	
User	TESTUSER	to	
Client		to	
To be stored		to	
Runtime Error		to	
Program Name		to	
Exception		to	
Transaction ID		to	

These files were investigated for each runtime error:

□ With Information on Exception/Short Text of Runtime Error

□ The program affected

□ Program and associated application components (long runtime)

□ Use old dump analysis

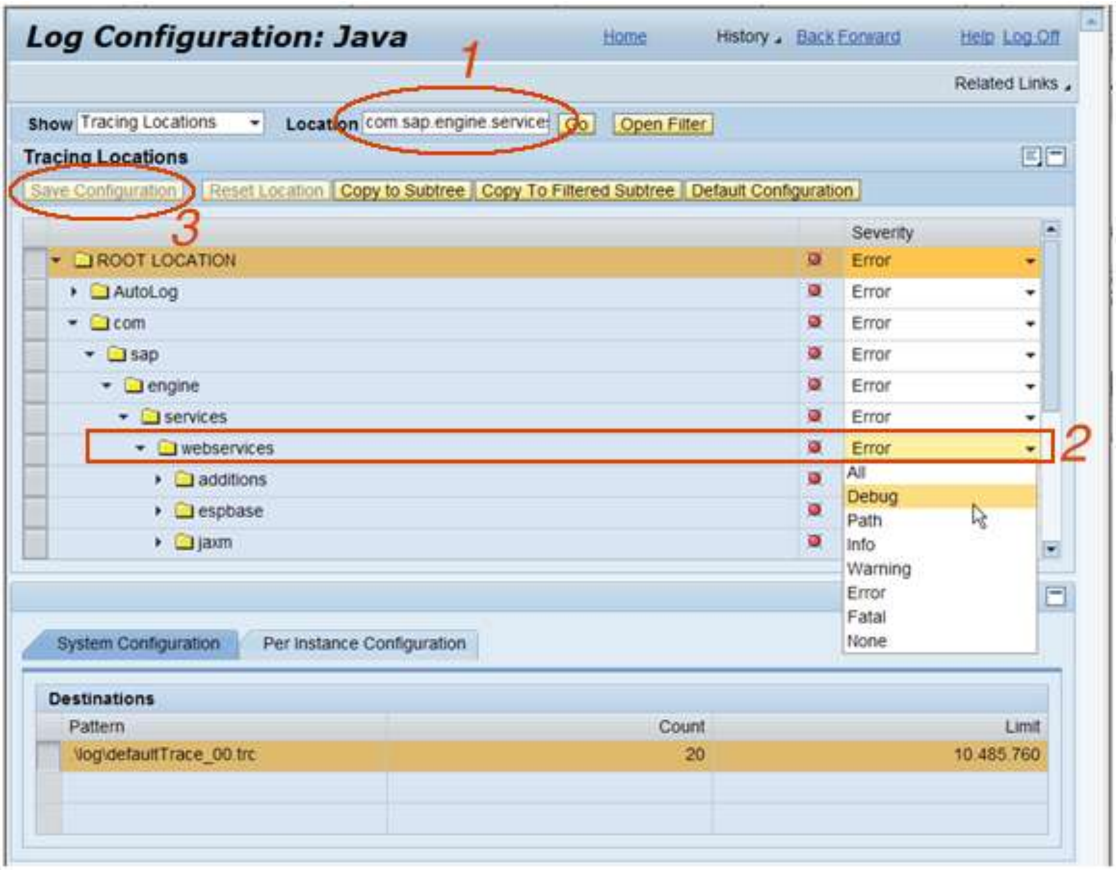
4.4 Use WSNavigator to Debug WS

Use WSNavigator to Debug WS:

1. Change trace level to “Debug” on for the packages:

   com.sap.engine.services.webservices

   com.sap.engine.services.wsrm



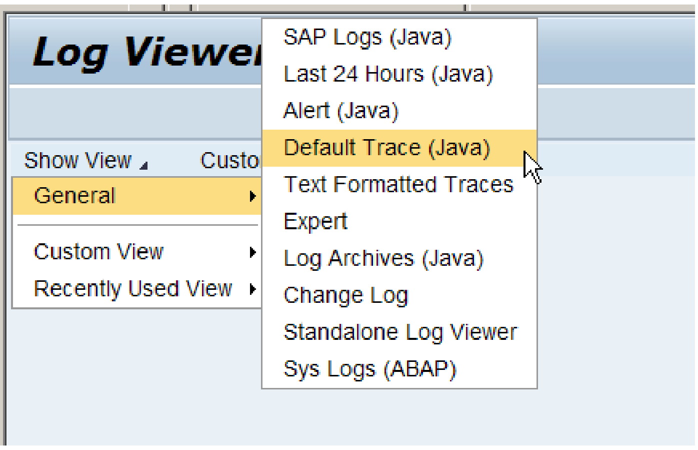
	Severity
ROOT LOCATION	Error
AutoLog	Error
com	Error
sap	Error
engine	Error
services	Error
webservices	Error
additions	
espbase	
jaxm	

Destinations

Pattern	Count	Limit
\log\defaultTrace_00.trc	20	10.485.760

2. Execute the service using WSNavigator.

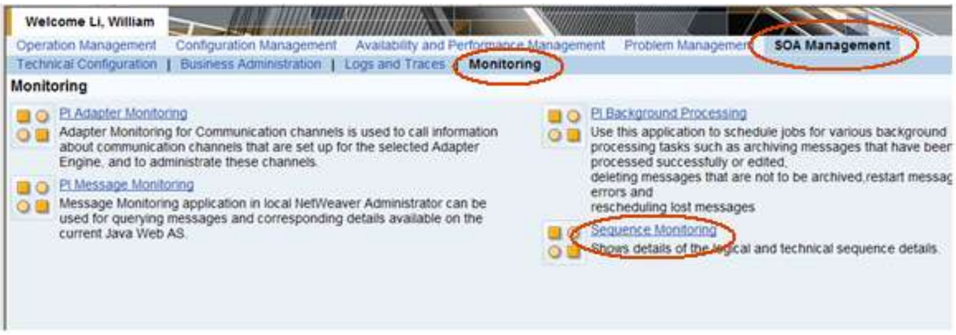
3. Go to the default trace in the Log Viewer:



a. Enter in the filter for “Location” column: com.sap.engine.services

b. Using SOAMANAGER:

i. In /nwa:


Welcome Li, William
Operation Management
Configuration Management
Availability and Performance Management
Problem Management
SOA Management
Technical Configuration | Business Administration | Logs and Traces | Monitoring

Monitoring

PI Adapter Monitoring
Adapter Monitoring for Communication channels is used to call information about communication channels that are set up for the selected Adapter Engine, and to administrate these channels.

PI Message Monitoring
Message Monitoring application in local NetWeaver Administrator can be used for querying messages and corresponding details available on the current Java Web AS.

PI Background Processing
Use this application to schedule jobs for various background processing tasks such as archiving messages that have beer processed successfully or edited,
deleting messages that are not to be archived,restart messag errors and
rescheduling lost messages

Sequence Monitoring
Shows details of the logical and technical sequence details.

ii. In ABAP:

TX: ST22

To examine the dumps (if any) generated by the WS call.

TX: SE38 - report SRT_ADMIN_CHECK

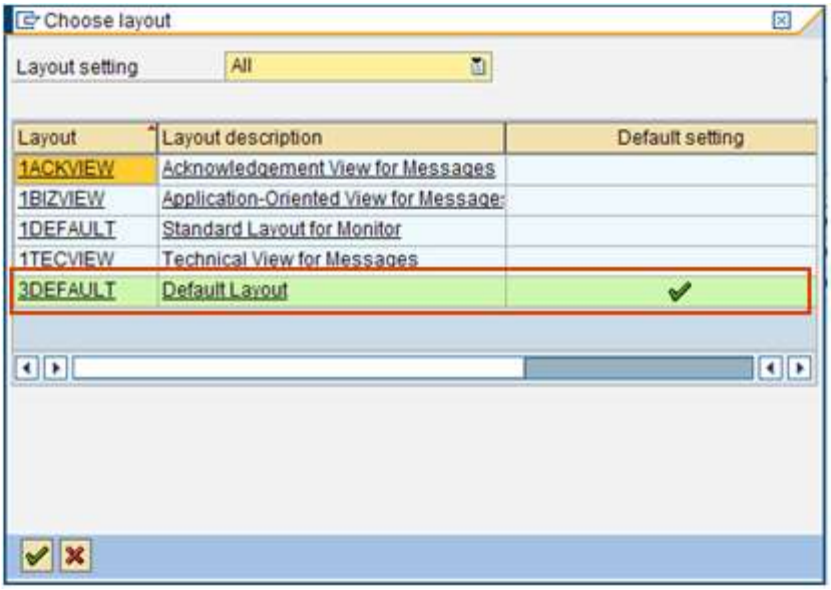
To determine any missing WS configurations.

c. Where to monitor?

Sender System:

SXMB_MONI (aka SXI_MONITOR)

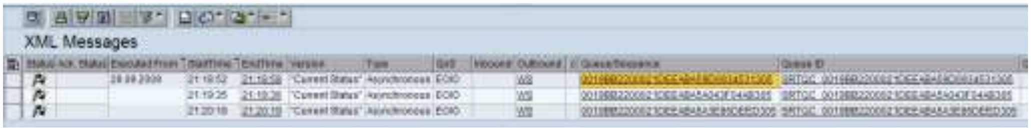
Use the default display:



Choose layout

Layout setting: All

Layout	Layout description	Default setting
1ACKVIEW	Acknowledgement View for Messages	
1BIZVIEW	Application-Oriented View for Message	
1DEFAULT	Standard Layout for Monitor	
1TECVIEW	Technical View for Messages	
3DEFAULT	Default Layout	✓

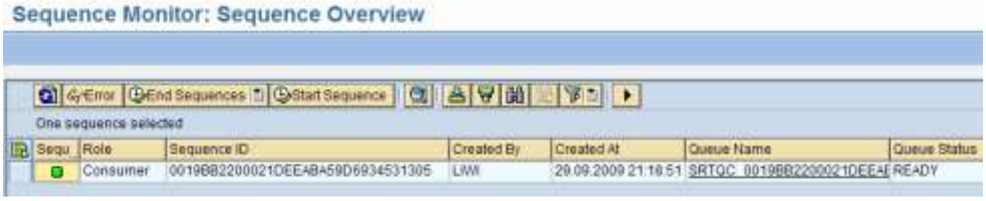


XML Messages

Status	Ack. Status	Executed From	StartTime	EndTime	Version	Type	QoS	Inbound	Outbound	Queue/Sequence	Queue ID
		28.09.2009	21:18:52	21:18:58	"Current Status"	Asynchronous	EOIO		WS	[illegible]	[illegible]
			21:19:25	21:19:28	"Current Status"	Asynchronous	EOIO		WS	[illegible]	[illegible]
			21:20:18	21:20:19	"Current Status"	Asynchronous	EOIO		WS	[illegible]	[illegible]

This display will provide the Queue/Sequence id and the queue name/ID:

d. From the sequence ID, we can get to the Sequence Monitor, where we can get additional information about our message. We can also examine errors and start/end the sequence.

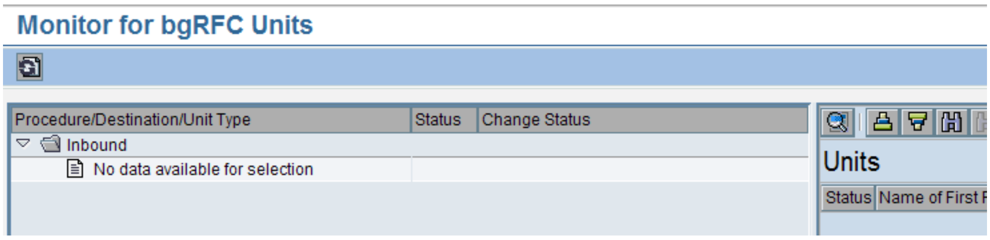


Sequence Monitor: Sequence Overview

One sequence selected

Sequ	Role	Sequence ID	Created By	Created At	Queue Name	Queue Status
🟩	Consumer	0019BB2200021DEEABA59D6934531305	LIWI	29.09.2009 21:18:51	SRTQC_0019BB2200021DEEAE	READY

e. From the Queue ID, we can examine the bgRFC queue where is message is processed. The Queue ID will indicated is our message is waiting in the queue and the reason it might be stuck.



Monitor for bgRFC Units

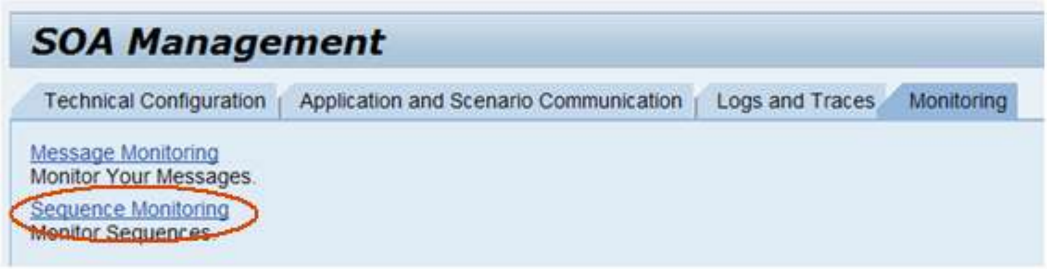
Procedure/Destination/Unit Type	Status	Change Status
Inbound		
No data available for selection		

Units

Status	Name of First F

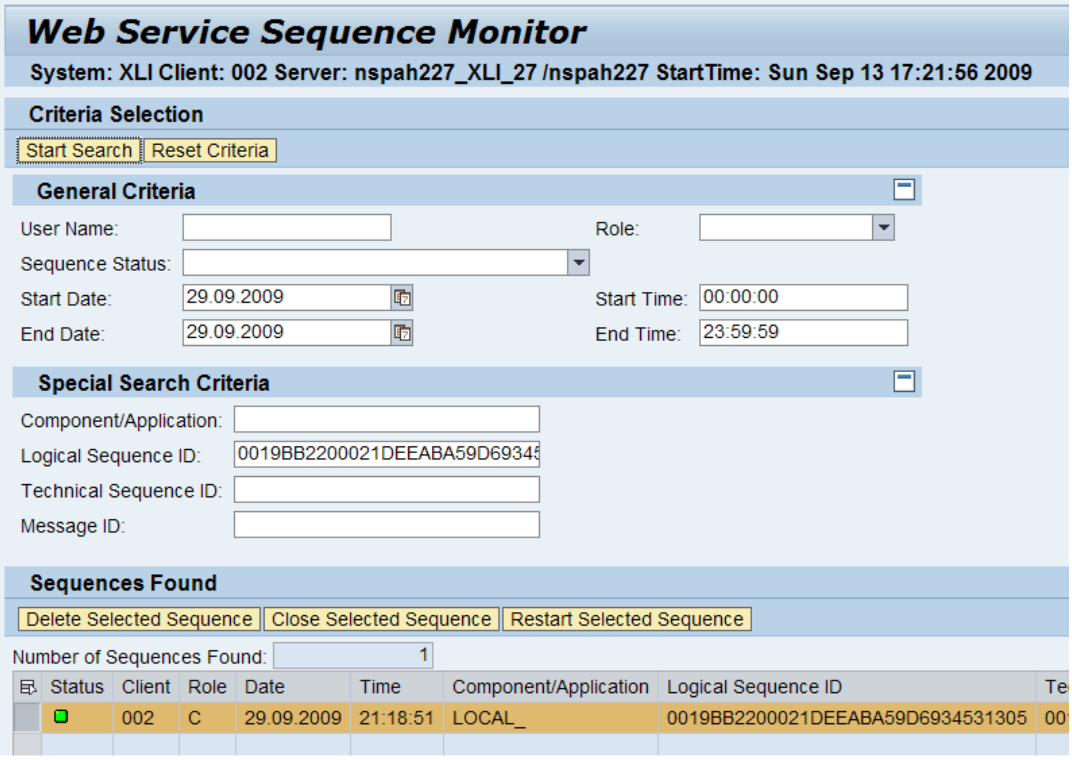
In addition to monitoring the message from the ABAP stack, we can also monitor the message from SOAMANAGER in the web browser.

In SOAMANAGER, select Monitoring → Sequence Monitoring:



(Message Monitoring is a link to the ABAP's SXMB_MONI.)

In SOAMANAGER's Sequence Monitor, we can copy-n-paste the Sequence ID from SXMB_MONI, or any other search criteria:



Web Service Sequence Monitor

System: XLI Client: 002 Server: nspah227_XLI_27 /nspah227 StartTime: Sun Sep 13 17:21:56 2009

Criteria Selection

General Criteria

User Name:

Role:

Sequence Status:

Start Date: 29.09.2009

Start Time: 00:00:00

End Date: 29.09.2009

End Time: 23:59:59

Special Search Criteria

Component/Application:

Logical Sequence ID: 0019BB2200021DEEABA59D69345

Technical Sequence ID:

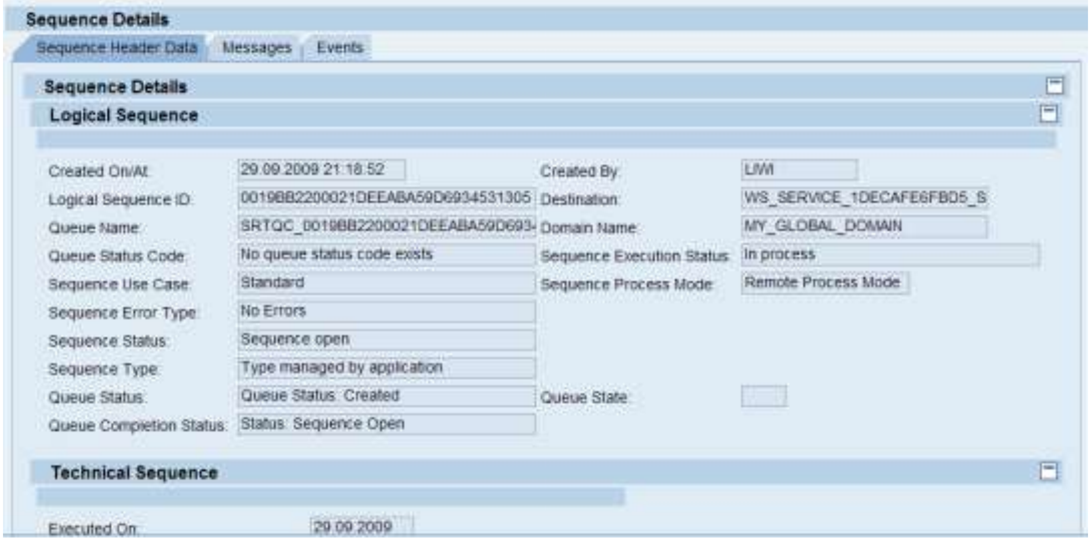
Message ID:

Sequences Found

Number of Sequences Found: 1

Status	Client	Role	Date	Time	Component/Application	Logical Sequence ID	Te
🟩	002	C	29.09.2009	21:18:51	LOCAL_	0019BB2200021DEEABA59D6934531305	00

From here, we can delete, close or restart the sequence. We can also examine additional detail about the sequence:



Sequence Details

Sequence Details

Logical Sequence

Created On/At:	29.09.2009 21:18:52	Created By:	LIWI
Logical Sequence ID:	0019BB2200021DEEABA59D6934531305	Destination:	WS_SERVICE_1DECAFE6FBD5_S
Queue Name:	SRTQC_0019BB2200021DEEABA59D693	Domain Name:	MY_GLOBAL_DOMAIN
Queue Status Code:	No queue status code exists	Sequence Execution Status:	In process
Sequence Use Case:	Standard	Sequence Process Mode:	Remote Process Mode
Sequence Error Type:	No Errors		
Sequence Status:	Sequence open		
Sequence Type:	Type managed by application		
Queue Status:	Queue Status: Created	Queue State:	
Queue Completion Status:	Status: Sequence Open		

Technical Sequence

Executed On: 29.09.2009

Receiver System:

Similarly to Sender System, SXMB_MONI and SOAMANAGER can be used to monitor the messages.

www.sdn.sap.com/irj/sdn/howtoguides