Oracle® Cloud

Using the SAP Adapter

Release 16.4

E68142-10

December 2016

This guide describes how to configure and add the SAP Adapter to an integration in Oracle Integration Cloud Service.

ORACLE

Oracle Cloud Using the SAP Adapter, Release 16.4

E68142-10

Primary Author: Mark Kennedy

Contents

Preface

Using the SAP Adapter describes how to configure the SAP Adapter as a connection in an integration in Oracle Integration Cloud Service.

Topics:

Audience

Using the SAP Adapter is intended for developers who want to use the SAP Adapter in integrations in Oracle Integration Cloud Service.

Related Resources

For more information, see these Oracle resources:

- Oracle Cloud

  `http://cloud.oracle.com`
- *Using Oracle Integration Cloud Service*
- *Using the Oracle Mapper*
- *Getting Started with Oracle Cloud*
- *Managing and Monitoring Oracle Cloud*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.

Convention	Meaning
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Getting Started with the SAP Adapter

Review the following conceptual topics to learn about the SAP Adapter and how to use it as a connection in integrations in Integration Cloud Service. A typical workflow of adapter and integration tasks is also provided.

Topics

About the SAP Adapter

The SAP Adapter enables you to perform operations on SAP objects as part of an integration in Oracle Integration Cloud Service.

The SAP Adapter provides the following benefits:

- Business objects (BAPIs), function modules (RFCs), or ALE/EDI messages (IDOCs) supported
- BAPI synchronous communication
- RFC synchronous communication
- IDOC execution through a queue in SAP
- Filtering of BAPI and RFC objects by functional area
- Search functionality at the SAP object level
- Support of direct connection to SAP
- Connection testing during configuration
- Caching of SAP metadata

The SAP Adapter is one of many predefined adapters included with Oracle Integration Cloud Service. You can configure the SAP Adapter as a connection in an integration in Oracle Integration Cloud Service. For information about Oracle Integration Cloud Service, connections, and integrations, see the following sections:

- About Oracle Integration Cloud Service
- About Oracle Integration Cloud Service Connections
- About Oracle Integration Cloud Service Integrations

Video

What Application Versions Does the SAP Adapter Support?

The SAP Adapter is certified with SAP versions ECC 6, ERP 6.0, and ERP 6.0 EhP7.

About Oracle Integration Cloud Service

Oracle Integration Cloud Service is a complete, secure, but lightweight integration solution that enables you to connect your applications in the cloud. It simplifies connectivity between your applications and connects both your applications that live in the cloud and your applications that still live on premises. Oracle Integration Cloud Service provides secure, enterprise-grade connectivity regardless of the applications you are connecting or where they reside.

Oracle Integration Cloud Service provides native connectivity to Oracle Software as a Service (SaaS) applications, such as Oracle Sales Cloud, Oracle RightNow Cloud, and so on. Oracle Integration Cloud Service *adapters* simplify connectivity by handling the underlying complexities of connecting to applications using industry-wide best practices. You only need to create a *connection* that provides minimal connectivity information for each system. Oracle Integration Cloud Service *lookups* map the different codes or terms used by the applications you are integrating to describe similar items (such as country or gender codes). Finally, the visual data mapper enables you to quickly create direct mappings between the trigger and invoke data structures. From the mapper, you can also access lookup tables and use standard XPath functions to map data between your applications.

Once you integrate your applications and activate the integrations to the runtime environment, the dashboard displays information about the running integrations so you can monitor the status and processing statistics for each integration. The dashboard measures and tracks the performance of your transactions by capturing and reporting key information, such as throughput, the number of messages processed successfully, and the number of messages that failed processing. You can also manage business identifiers that track fields in messages and manage errors by integrations, connections, or specific integration instances.

About Oracle Integration Cloud Service Connections

Connections define information about the instances of each predefined configuration you are integrating. Oracle Integration Cloud Service includes a set of predefined *adapters*, which are the types of applications on which you can base your connections, such as Oracle Sales Cloud, Oracle Eloqua Cloud, Oracle RightNow Cloud, and others. A connection is based on an adapter. A connection includes the additional information required by the adapter to communicate with a specific instance of an application (this can be referred to as metadata or as connection details). For example, to create a connection to a specific RightNow Cloud application instance, you must select the Oracle RightNow adapter and then specify the WSDL URL, security policy, and security credentials to connect to it.

Video

About Oracle Integration Cloud Service Integrations

Integrations are the main ingredient of Oracle Integration Cloud Service. An integration includes at the least a trigger (source) connection (for requests sent to Oracle Integration Cloud Service) and invoke (target) connection (for requests sent from Oracle Integration Cloud Service to the target) and the field mapping between those two connections.

When you create your integrations, you build on the connections you already created by defining how to process the data for the trigger (source) and invoke (target) connections. This can include defining the type of operations to perform on the data, the business objects and fields against which to perform those operations, required schemas, and so on. To make this easier, the most complex configuration tasks are handled by Oracle Integration Cloud Service. Once your trigger (source) and invoke (target) connections are configured, the mappers between the two are enabled so you can define how the information is transferred between the trigger (source) and invoke (target) data structures for both the request and response messages.

 Video

Typical Workflow for Creating and Including an Adapter Connection in an Integration

You follow a very simple workflow to create a connection with an adapter and include the connection in an integration in Integration Cloud Service.

Step	Description	More Information
1	Create the adapter connections for the applications you want to integrate. The connections can be reused in multiple integrations and are typically created by the administrator.	Creating an SAP Adapter Connection
2	Create the integration. When you do this, you add trigger and invoke connections to the integration.	Creating an Integration and Adding the SAP Adapter Connection to an Integration
3	Map data between the trigger connection data structure and the invoke connection data structure.	Mapping Integration Cloud Service Data of *Using the Oracle Mapper*
4	(Optional) Create lookups that map the different values used by those applications to identify the same type of object (such as gender codes or country codes).	Creating Lookups of *Using Oracle Integration Cloud Service*
5	Activate the integration.	Managing Integrations of *Using Oracle Integration Cloud Service*
6	Monitor the integration on the dashboard.	Monitoring Integration Cloud Services of *Using Oracle Integration Cloud Service*

Step	Description	More Information
7	Track payload fields in messages during runtime.	Assigning Business Identifiers for Tracking Fields in Messages and Managing Business Identifiers for Tracking Fields in Messages of *Using Oracle Integration Cloud Service*
8	Manage errors at the integration level, connection level, or specific integration instance level.	Managing Errors of *Using Oracle Integration Cloud Service*

2

Creating an SAP Adapter Connection

A connection is based on an adapter. You define connections to the specific cloud applications that you want to integrate. The following topics describe how to define connections:

Topics

Prerequisites for Creating a Connection

You must satisfy the following prerequisites for creating a connection with the SAP Adapter.

1. Know the **Client** login parameter, an ID with a numeric value.
2. Know the code for the **Language** login parameter.

   For example, the code for English is en.
3. Know the host name or IP address of the **Application Server** upon which the SAP instance runs.
4. Know the **System Number** or **Instance Number** for the application server instance.
5. Know the **System ID** for the application server connection, a value such as N4S.
6. Know the username and password for access.
7. If you are connecting to an on-premises application, know the name of the agent group you are using.
8. Follow the applicable instructions in Configuring Inbound and Outbound Communication for configuring inbound and outbound communication.
9. If you use the on-premises agent with the SAP Adapter, you have to add some additional JAR files to the agent's class path. See Adding JAR Files to the Agent Class Path or more information.

Uploading an SSL Certificate

Certificates are used to validate outbound SSL connections. If you make an SSL connection in which the root certificate does not exist in Oracle Integration Cloud Service, an exception is thrown. In that case, you must upload the appropriate certificate. A certificate enables Oracle Integration Cloud Service to connect with external services. If the external endpoint requires a specific certificate, request the certificate and then upload it into Oracle Integration Cloud Service.

To upload a certificate:

1. From the Oracle Integration Cloud Service home page, click the **Administration** tab in the upper right corner.

   All certificates currently uploaded to the trust store are displayed in the Certificates dialog. A navigation panel on the left side of the dialog displays the following details:

   - **All**: Displays all certificates in Oracle Integration Cloud Service.
   - **System**: Displays the certificates automatically included in Oracle Integration Cloud Service. These certificates cannot be deleted.
   - **Uploaded**: Displays the certificates uploaded by individual users. These certificates can be deleted and updated.

2. Click **Upload Certificate** at the top of the page.
3. In the Upload Certificate dialog box, enter a unique identifier for the certificate.

   This is a name you can use to identify the certificate.

4. Click **Browse** to locate the certificate file (`.cer`).
5. Click **Upload**.
6. Click the certificate name to view details such as the subject of the certificate, the issuer of the certificate, the date the certificate was issued, and the date the certificate expires.

Creating a Connection

The first step in creating an integration is to create the connections to the applications with which you want to share data.

1. In the Integration Cloud Service toolbar, click **Designer**.
2. On the Designer Portal, click **Connections**.
3. Click **New Connection**.

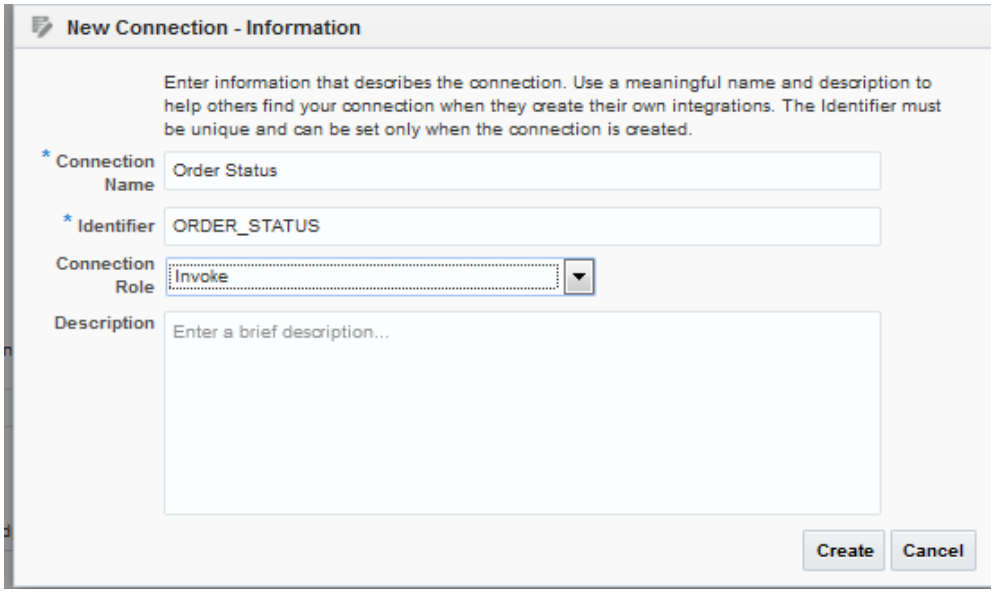
   The Create Connection — Select Adapter dialog is displayed.

4. Select an adapter from the dialog. You can also search for the type of adapter to use by entering a partial or full name in the Search field, and clicking **Search**.

   The New Connection — Information dialog is displayed.

5. Enter the information to describe the connection.

- Enter a meaningful name to help others find your connection when they begin to create their own integrations. The name you enter is automatically added in capital letters to the **Identifier** field. If you modify the identifier name, do not include a blank space (for example, `OSC Inbound`).
- Select the role (direction) in which to use this connection (trigger, invoke, or both). Only the roles supported by this adapter are displayed for selection. When you select a role, only the connection properties and security policies appropriate to that role are displayed on the Connections page. If you select an adapter that supports both invoke and trigger, but select only one of those roles, then try to drag the adapter into the section you did not select, you receive an error (for example, configure an Oracle RightNow Cloud Adapter as only an invoke, but drag the adapter to the trigger section).
- Enter an optional description of the connection.



6. Click **Create**.

   Your connection is created and you are now ready to configure connection details, such as email contact, connection properties, security policies, and connection login credentials.

Adding a Contact Email

From the Connection Administrator section of the connection, you can add a contact email address for notifications.

1. In the **Email Address** field, enter an email address to receive email notifications when problems occur.
2. In the upper right corner, click **Save**.

Configuring Connection Properties

Enter connection information so your application can process requests.

1. Click **Configure Connectivity**.

   The Connection Properties dialog is displayed.

2. Click the **Upload File** checkbox.

3. Select the JCO connection properties file to use. For information about available files, see JCO Connection Properties Files.

4. Click **Upload**.

5. Click **OK**.

Configuring Connection Security

Configure security for your SAP connection by selecting the security policy and setting login credentials.

1. Click **Configure Credentials**.

2. Enter your login credentials.

   a. Select the security policy. Only the Username Password Token policy is supported. It cannot be deselected.

   b. Enter a username and password to connect to the SAP instance.

   c. Reenter the password a second time.

3. Click **OK**.

Configuring an Agent Group

Configure an agent group for accessing your on-premises application.

1. Click **Configure Agents**.

   The Select an Agent Group window appears.

2. Click the name of the agent group.

3. Click **Use**.

   You are now ready to test your connection.

Related Topics:

About Agents and Integrations Between On-Premises Applications and Oracle Integration Cloud Service

Managing Agent Groups and the On-Premises Agent

Monitoring Agents

Testing the Connection

Test your connection to ensure that it is successfully configured.

1. In the upper right corner of the page, click **Test**.

If successful, the following message is displayed and the progress indicator shows 100%.

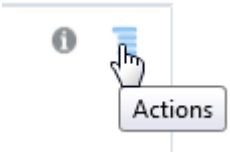
```
The connection test was successful!
```

2. If your connection was unsuccessful, an error message is displayed with details. Verify that the configuration details you entered are correct.
3. When complete, click **Save**.

Editing a Connection

You can edit connection settings after creating a new connection.

1. In the Oracle Integration Cloud Service toolbar, click **Designer**.
2. On the Designer Portal, click **Connections**.
3. On the Connections page, search for the connection name.
4. Select **Edit** from the connection **Actions** menu or click the connection name.



The Connection page is displayed.

5. To edit the notification email contact, change the email address in the **Email Address** field.
6. To edit the connection properties, click **Configure Connectivity**. Note that some connections do not include this button. If your connector does not include a **Configure Connectivity** button, then click the **Configure Credentials** button.

Cloning a Connection

You can clone a copy of an existing connection. It is a quick way to create a new connection.

1. In the Oracle Integration Cloud Service toolbar, click **Designer**.
2. On the Designer Portal, click **Connections**.
3. On the Connections page, search for the connection name.
4. Select **Clone** from the connection **Actions** menu.

The Clone Connection dialog is displayed.

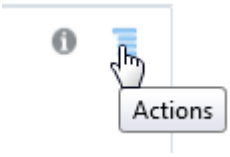
5. Enter the connection information.
6. Click **Clone**.
7. Click **Edit** to configure the credentials of your cloned connection. Cloning a connection does not copy the credentials.

See Editing a Connection for instructions.

Deleting a Connection

You can delete a connection from the connection menu.

1. In the Oracle Integration Cloud Service toolbar, click **Designer**.
2. On the Designer Portal, click **Connections**.
3. On the Connections page, search for the connection name.
4. Click **Delete** from the connection **Actions** menu.



The Delete Connection dialog is displayed if the connection is not used in an integration.

5. Click **Yes** to confirm deletion.

3

Creating an Integration

Integrations use the adapter connections you created to your applications, and define how information is shared between those applications. You can create, import, modify, or delete integrations; create integrations to publish or subscribe to messages; add and remove request and response enrichment triggers; and create routing paths for different invoke endpoints in integrations. Click the following topics for more information.

Topic

- Creating Integrations (in *Using Oracle Integration Cloud Service*)

4

Adding the SAP Adapter Connection to an Integration

When you drag the SAP Adapter into an integration, the Adapter Endpoint Configuration Wizard is invoked. This wizard guides you through configuration of SAP Adapter endpoint properties.

The following sections describe the wizard pages that guide you through configuration of the SAP Adapter as an endpoint in an integration.

Topics

For general information about the SAP Adapter, see About the SAP Adapter.

Configuring Basic Information Properties

You can enter a name and description on the Basic Info page of each adapter in your integration.

Topics

What You Can Do from the Basic Info Page

You can specify the following values on the Basic Info page. The Basic Info page is the initial wizard page that is displayed whenever you drag an adapter to the section of the integration canvas supported by your adapter.

- Specify a meaningful name.
- Specify a description of the responsibilities.

What You See on the Basic Info Page

The following table describes the key information on the Basic Info page.

Element	Description
What do you want to call your endpoint?	Provide a meaningful name so that others can understand the responsibilities of this connection. You can include English alphabetic characters, numbers, underscores, and dashes in the name. You cannot include the following: • Blank spaces (for example, `My Inbound Connection`) • Special characters (for example, `#;83&` or `righ(t)now4`) • Multibyte characters
What does this endpoint do?	Enter an optional description of the connection's responsibilities. For example: `This connection receives an inbound request to synchronize account information with the cloud application.`

Configuring SAP Adapter Trigger Objects and Methods Properties

Enter the SAP Adapter trigger object and method property values for your integration.

Topics

What You Can Do from the Trigger Objects and Methods Page

You can specify the following values and actions on the Objects and Methods page.

- Choose between:
  - Business Objects (BAPIs)
  - Function Modules (RFCs)
  - ALE/EDI Messages (IDOCs)
- Select a functional area.
- Select an object and one of its methods.
- Apply processing options to change runtime behavior.

What You See on the Trigger Objects and Methods Page

The Objects and Methods page changes depending on which category you choose.

When the Objects and Methods page is displayed, you can choose which of the following categories to use:

- **Business Objects (BAPIs):** The Business Application Programming Interface is the standard SAP interface. BAPIs allow integration at the business level rather than the technical level. This provides for greater linkage stability and independence from the underlying communication technology.
- **Function Modules (RFCs):** RFC allows for remote calls between two SAP systems (R/3 or R/2) or between an SAP system and a non-SAP system.
- **ALE/EDI Messages (IDOCs):** Intermediate Document is a standard data structure for electronic data interchange (EDI) between application programs written for the popular SAP business system or between an SAP application and an external program.

Depending on which category you choose, you are asked to select objects or methods. After you choose objects or methods, click the **Processing Options** link to change runtime behavior.

Business Objects (BAPIs)

The following table describes the page you see if you select Business Objects (BAPIs).

Element	Description
Processing Options	Program ID. Enter a case-sensitive program identifier specified on the SAP gateway server. The program ID is a unique identifier for your communication session specified by your system administrator. The value entered in this field must match the one exposed on the gateway. For the gateway service property, enter the service name.
Application Components	Expose a hierarchy of components such as sales, finance and HR.
Select Functional Area	Select a functional area, such as Sales and Distribution.
Choose Objects to Filter BAPI Methods	Select an object to see the BAPI methods available in it. You are shown the list of available objects based on the functional area you selected above.
Objects	Select an object, such as Sales Order.
Methods	Select a method, such as CreateFromData.

Function Modules (RFCs)

The following table describes the page you see if you select Function Modules (RFCs).

Element	Description
Processing Options	Program ID. Enter a case-sensitive program identifier specified on the SAP gateway server. The program ID is a unique identifier for your communication session specified by your system administrator. The value entered in this field must match the one exposed on the gateway. For the gateway service property, enter the service name.
Select Functional Area	Select a functional area that is available in the selected RFC category to filter the RFC method list. If you select a functional area, the RFC method list and the Groups UI list are updated.
Methods	Select a method, such as CreateFromData.

ALE/EDI Messages (IDOCs)

The following table describes the page you see if you select ALE/EDI Messages (IDOCs).

Element	Description
Processing Options	There are the following processing options for IDOCs: **AutoSYSTAT01**, **EncodeIDOC**, **ControlCharacter** and **ProgramID**. • **AutoSYSTAT01** – Yes: The adapter sends a `SYSTAT01` message upon a successful reception of an IDOC message. – No: Nothing is sent back to SAP by the adapter upon successful reception of an IDOC message. • **EncodeIDOC** – Flatfile: SAP uses a non-XML text-based format called the Flatfile IDOC format for serializing IDOC messages to the file system. In a Flatfile IDOC, all IDOC records, including the control record and the data record, are stored in lines of text separated by a line delimiter. – No: SAP uses the XML format to send field names and complete data to IDOC records. • **ControlCharacter** This property dictates how the adapter deals with characters in the payload that are not supported by the XML 1.0 standard. – Remove: The adapter removes the character from the payload. – Space: The adapter replaces the character with a space. – Encode: The adapter encodes the character into its decimal format. • **Program ID** Enter a case-sensitive program identifier specified on the SAP gateway server. The program ID is a unique identifier for your communication session specified by your system administrator. The value entered in this field must match the one exposed on the gateway. For the Gateway Service property, enter the service name. **Note**: The program ID provided at design time overrides the Program ID provided inside the properties file.
Groups	Select a group of methods such as `matmas`, rather than an individual method..
Methods	Select an individual method, such as `matmas01`.

Configuring SAP Adapter Invoke Objects and Methods Properties

Enter the SAP Adapter invoke object and method property values for your integration.

Topics

What You Can Do from the Invoke Objects and Methods Page

You can specify the following values and actions on the Objects and Methods page.

- Choose between:
  - Business Objects (BAPIs)
  - Function Modules (RFCs)
  - ALE/EDI Messages (IDOCs)
- Select a functional area.
- Select an object and one of its methods.
- Apply processing options to change runtime behavior.

What You See on the Invoke Objects and Methods Page

The Objects and Methods page changes depending on which category you choose.

When the Objects and Methods page is displayed, you can choose which of the following categories to use:

- **Business Objects (BAPIs):** The Business Application Programming Interface is the standard SAP interface. BAPIs allow integration at the business level rather than the technical level. This provides for greater linkage stability and independence from the underlying communication technology.
- **Function Modules (RFCs):** RFC allows for remote calls between two SAP systems (R/3 or R/2) or between an SAP system and a non-SAP system.
- **ALE/EDI Messages (IDOCs):** Intermediate Document is a standard data structure for electronic data interchange (EDI) between application programs written for the popular SAP business system or between an SAP application and an external program.

Depending on which category you choose, you are asked to select objects or methods. After you choose objects or methods, click the **Processing Options** link to change runtime behavior.

Business Objects (BAPIs)

The following table describes the page you see if you select Business Objects (BAPIs).

Element	Description
Processing Options	Use the **Commit Transaction** option to specify whether the interaction with SAP is stateful or stateless.
Select Functional Area	Select a functional area, such as Sales and Distribution.

Element	Description
Objects	Select an object, such as Sales Order.
Methods	Select a method, such as CreateFromData.

Function Modules (RFCs)

These are the SAP communication methods that are supported by the SAP adapter for outbound processing.

Transactional RFC (tRFC): This is an asynchronous communication method that executes the called function in the target system only once. The listener to the port need not be available at the time the SAP RFC client program executes a tRFC. The tRFC component stores the called RFC function together with the corresponding data in the SAP database under a unique transaction ID (TID).

Queued RFC (qRFC): This is also an asynchronous communication method that guarantees that multiple requests are processed in the order specified by the sender. tRFC can be serialized using queues (inbound and outbound queues). The tRFC requests that are serialized using the inbound/outbound queues in SAP are called queued RFC (qRFC). qRFC is an extension of tRFC that processes requests that have no predecessors in the same queue. You can use qRFC to guarantee that several requests are processed in a defined order.

The following table describes the page you see if you select Function Modules (RFCs).

Element	Description
Processing Options	Use the **Commit Transaction** option to specify whether the interaction with SAP is stateful or stateless.. Use the **RFCOptions** option to specify: • **SYNC RFC** — No RFC processing. • **Transactional RFC** — Transactional RFC communication. • **Queued RFC** — Process the requests in a queue. You are prompted for the name of the queue which is already defined in SAP.
Select Functional Area	Select a functional area, such as Sales and Distribution.
Methods	Select a method, such as CreateFromData.

ALE/EDI Messages (IDOCs)

The following table describes the page you see if you selectALE/EDI Messages (IDOCs).

Element	Description
Processing Options	There is one processing option for IDOCs — **QueueName**. Use the **QueueName** option to process the requests in a queue. You are prompted for the name of the queue which is already defined in SAP.
Groups	Select a group of methods such as `matmas`, rather than an individual method..
Methods	Select an individual method, such as `matmas01`.

Reviewing Configuration Values on the Summary Page

You can review the specified adapter configuration values on the Summary page.

Topics

What You Can Do from the Summary Page

You can review configuration details from the Summary page. The Summary page is the final wizard page for each adapter after you have completed your configuration.

- View the configuration details you defined for the adapter. For example, if you have defined an inbound trigger (source) adapter with a request business object and immediate response business object, specific details about this configuration are displayed on the Summary page.
- Click **Done** if you want to save your configuration details.
- Click a specific tab in the left panel or click **Back** to access a specific page to update your configuration definitions.
- Click **Cancel** to cancel your configuration details.

What You See on the Summary Page

The following table describes the key information on the Summary page.

Element	Description
Summary	Displays a summary of the configuration values you defined on previous pages of the wizard. The information that is displayed can vary by adapter. For some adapters, the selected business objects and operation name are displayed. For adapters for which a generated XSD file is provided, click the XSD link to view a read-only version of the file. To return to a previous page to update any values, click the appropriate tab in the left panel or click **Back**.

5

Creating Mappings and Lookups in Integrations

You must map data between trigger (source) connections and invoke (target) connections in integrations. You can also optionally create lookups in integrations.

Topics

- Mapping Integration Cloud Service Data (in *Using Oracle Integration Cloud Service*)
- Creating Lookups (in *Using Oracle Integration Cloud Service*)

6

Administering Integrations

Oracle Integration Cloud Service provides you with the information and tools required to activate, monitor, and manage your integrations in the runtime environment.

Topic

- Administering Integration Cloud Service (in *Using Oracle Integration Cloud Service*)

A

Configuring Inbound and Outbound Communication

As part of the prerequisites for setting up the SAP Adapter, you have to configure inbound and outbound communication.

The following sections describe how to configure the SAP Adapter for inbound and outbound communication.

Topics

SAP Inbound Communication

During SAP inbound communication, the SAP Adapter acts as a client sending requests to the SAP system.

This section describes how to configure the adapter for communication.

Topics

Prerequisites

Take the following actions before you begin configuration.

> **Note:** You may need to consult with your SAP Administrator for the following configuration tasks.

The following entries need to be updated on the system on which the Oracle Weblogic Server is running.

- The `hosts` file of the system (maintained in the `etc` folder) should have the following entry:

```
SAP_System_Host_IP  SAP_System_Hostname  SAP_System_Hostname_With_Domain_Name
```

- The `services` file of the system (maintained in the `etc` folder) should have the following entries. You must replace *sysnr* with the actual SAP system number (such as `00`), and not the port number.

  ```
  sapgwsysnr 33sys_no/tcp
  sapdpsysnr 32sys_no/tcp
  ```

  Where *sysnr* is the system number of the SAP server. This entry is *not* the port number.

To connect to SAP using a message server, the following information must be maintained in the `services` file in the `etc` folder, in addition to the above two entries. Replace *`sysnr`* and *`SID`*.

```
sapmsSID36sysnr/tcp
```

Where *SID* is the system ID of the SAP server.

Configuring a Logical System

Use the following steps to configure a logical system.

Prerequisite Steps

1. To connect to SAP using the host name, the following entries must be in the `Hosts` file:

   ```
   IP Hostname FQHostname
   ```

2. To connect to SAP using MS, the following info must be maintained in the `service` file:

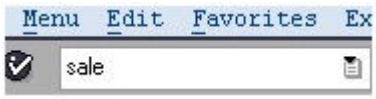
   ```
   SapmsSID36sysnr/tcp
   ```

Define a Logical System

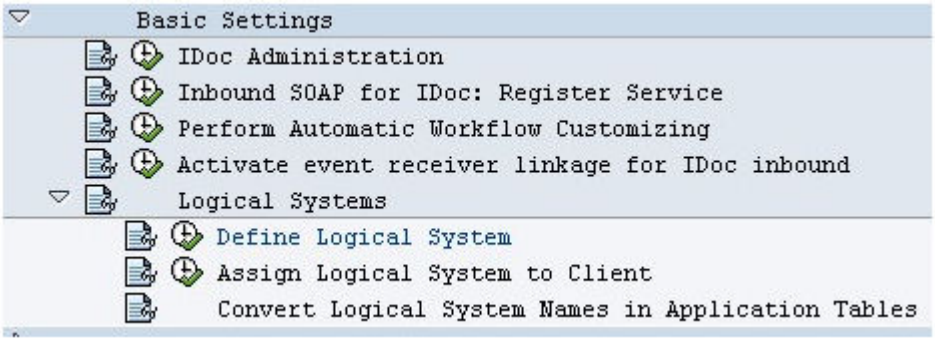
The logical system is used to identify an individual client in a system in ALE communication between SAP systems.

Use the following steps to define a logical system:

1. From the SAP easy access screen, navigate to the SALE transaction.



2. Open the basic settings and then the **Logical Systems** node.



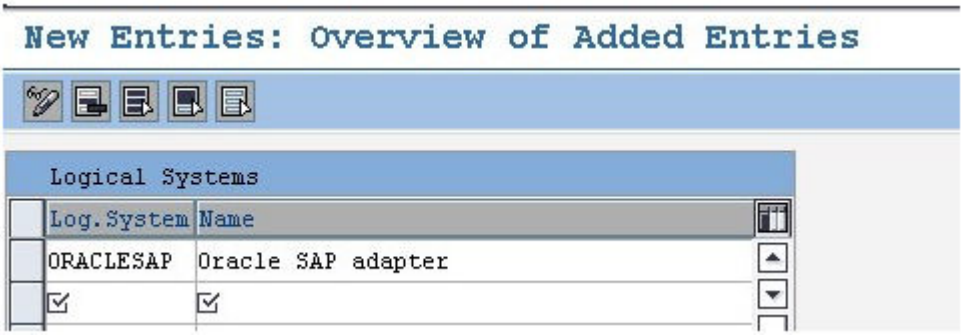
3. In the list, click **Define Logical Systems**.

   A popup window appears with the following message: `Caution: The table is cross-client`.

4. Click **Enter**.

5. Click **New Entries**.

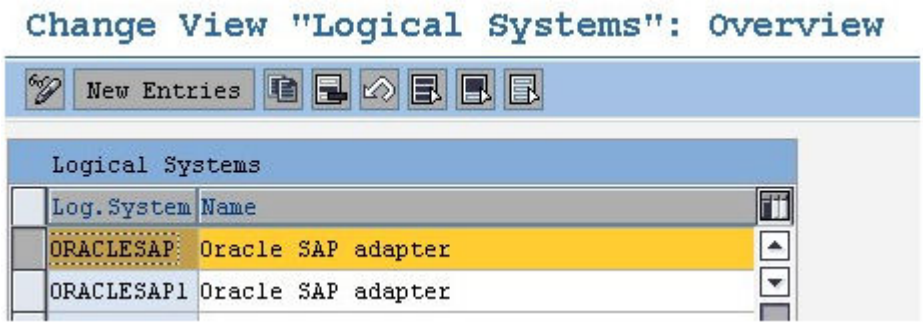
6. Enter the Logical System name and description.



7. Click **Save**.

8. Press **Enter** when the popup window appears.

   The entry for Logical System will now be visible in the table.



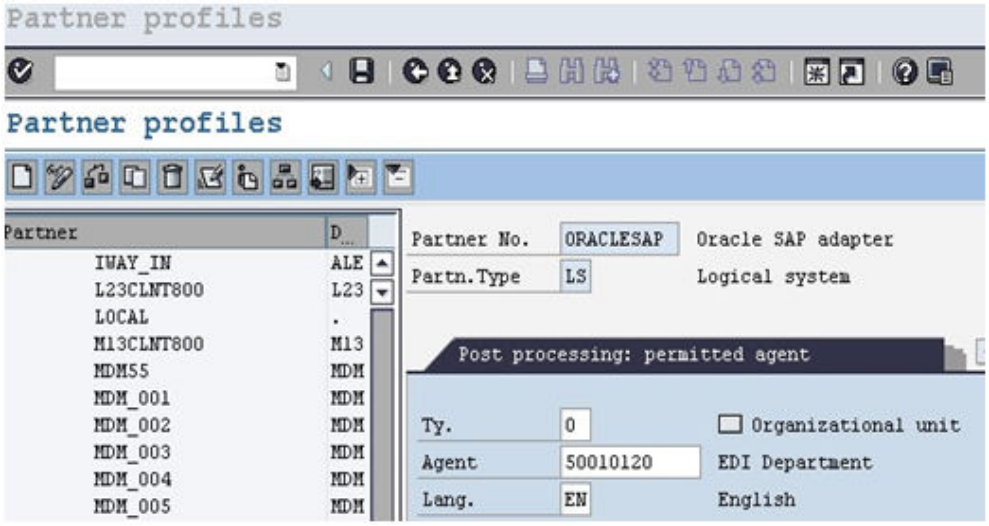
Configuring a Partner Profile

In SAP, all partner systems involved in a distribution model have a profile. There are several profile types, such as customer profiles and vendor profiles. This distinction is generally not necessary and you usually create your partners profiles using a generic Logical System type.

To create a partner profile:

1. Run the `we20` transaction.

2. **Click Partner Type LS**.
3. Click **Create**.
4. Enter the Partner No. — the logical system name that was created earlier.

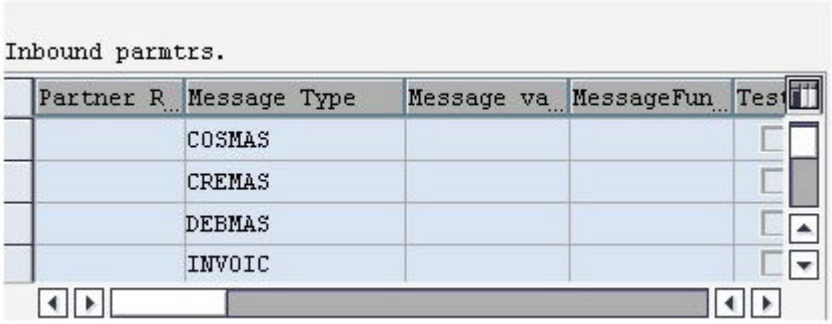


5. Click **Save**.
6. Click the **Add** icon to add the inbound parameters.

For a sender partner system (inbound parameters are filled in), the following important settings are set per the message type in the partner profile:

- A process code used to indicate which function module to use to convert the IDoc data to SAP data.
- The time the IDoc was input — when the IDoc is created in the system, or on request (using the `RBDAPP01` program).
- The post processing agent that will treat the data input errors if required. The post processing agent can be either a user or any other HR organizational unit.

7. Enter the message types that must be received from the partner systems.

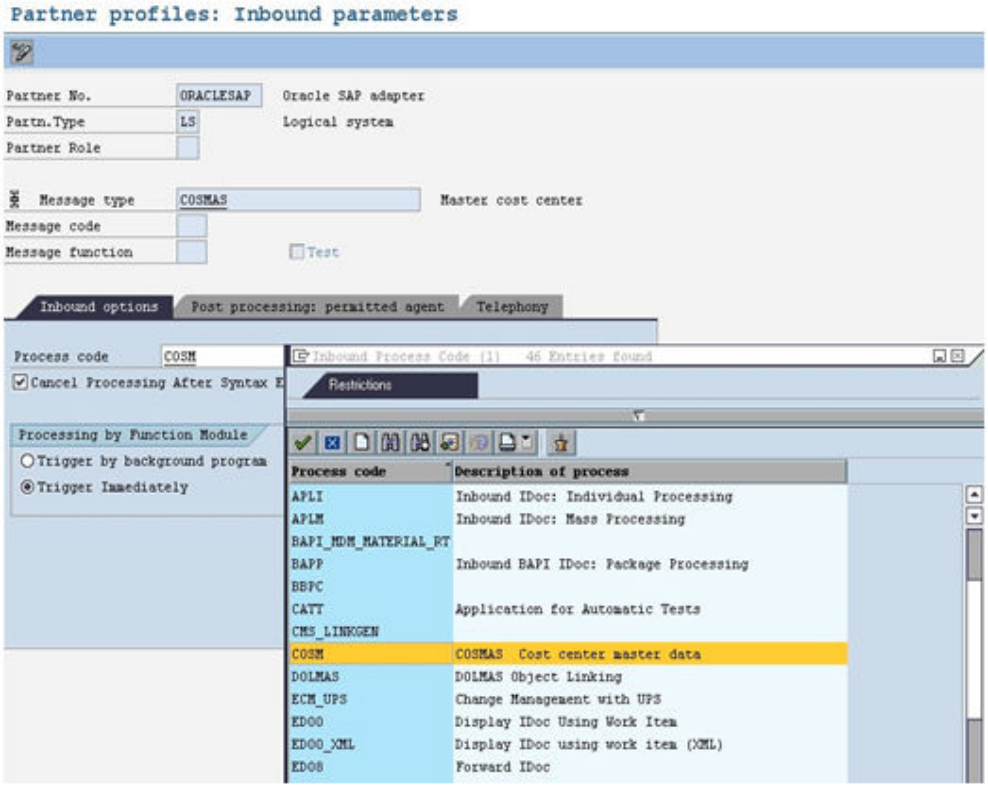


Configuring Inbound Process Code

The process code contains the details of the function module that are used for IDoc processing. The message type can be linked to the process code.

To define the process code:

1. Click on the message type in inbound parameters.
2. Click on the process code and press F4 to get the process codes available in the SAP system.
3. Choose the appropriate process code for that particular message type.
4. Check both the **Trigger Immediately** radio button and the **Cancel Processing After Syntax Error** check box.



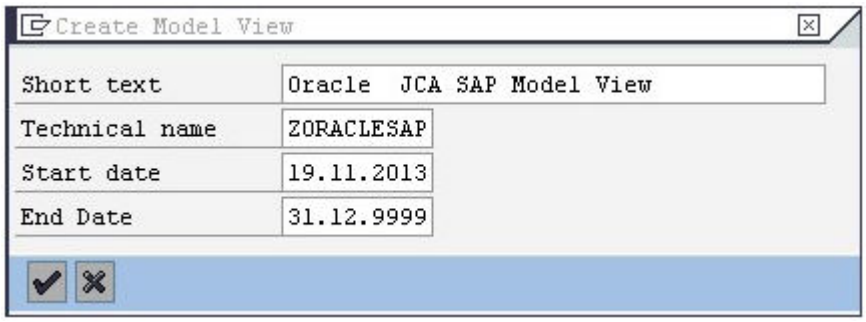
5. Click **Save**.

Configuring a Distribution Model

The distribution model determines the sender and receiver of the IDoc's and defines the transfer rules.

To create a distribution model:

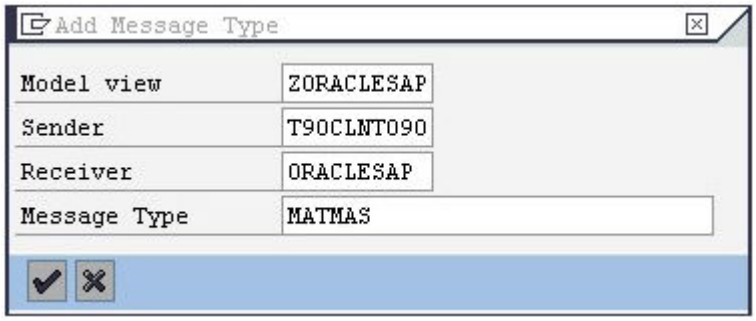
1. Run the **bd64** transaction.
2. Click the **Edit** icon.
3. Click the **Create model view** button.
4. Enter the distribution model name and description.



5. Highlight the model view you created.

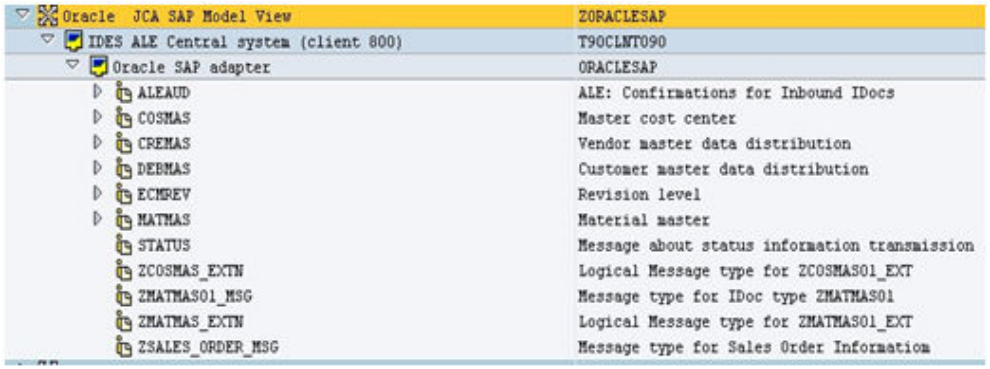
6. Click the **Add message type** button.

7. Enter the **Sender** (the logical system maintained for that SAP system), **Receiver** (the logical system name for the partner system), and the **Message Type** being sent to the partner system.



8. Add all the required message types.

   After you add all the required message types, the model view should look like the following image.



SAP Outbound Communication

During SAP outbound communication, the SAP Adapter acts as a server that receives requests from the SAP System.

The following configurations are required for outbound SAP communication.

Topics

- Configuring an RFC Destination and Program ID
- Creating a Port

- Configuring a Logical System
- Configuring a Distribution Model
- Configuring a Partner Profile

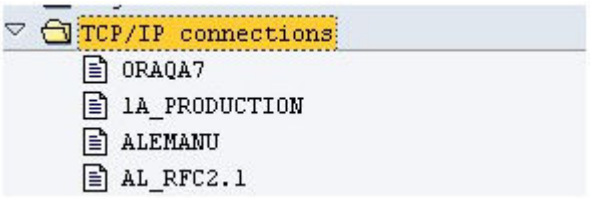
Configuring an RFC Destination and Program ID

An RFC destination may be seen as a set of settings necessary to connect to a system using the RFC protocol. RFC settings include the address and partner system type, along with connection information such as the user ID and password to use. The RFC destinations of all partner systems must be defined on all systems included in the distribution model. The transaction to use for this purpose is SM59.

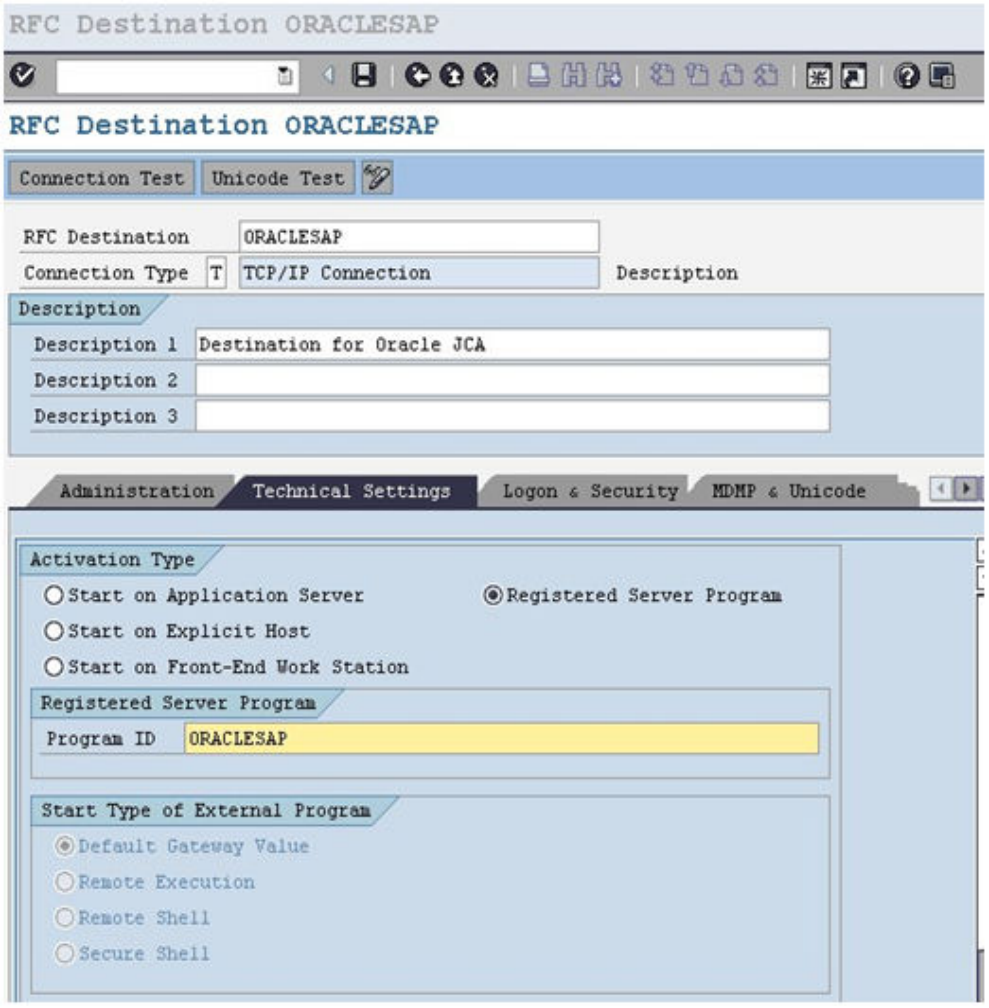
To define an RFC destination:

1. Navigate to the SM59 transaction

2. Click on **TCP/IP connections**.

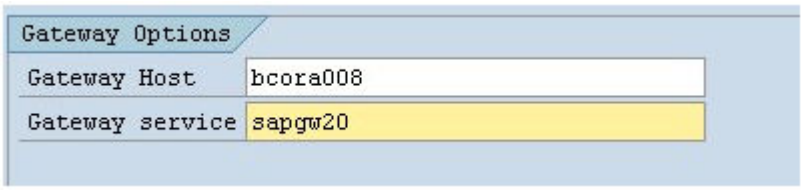


3. Click **Create**.
4. Enter the RFC destination name and the description along with the program ID and click on **Registered Server Program**.



An RFC server program registers itself under the Program ID.

5. Enter the **Gateway Host** and **Gateway Service** name.



6. Click **Save**.

The RFC destination is now configured.

> **Note:** The program ID is case sensitive. For example, "ORAQA1" is *not* equivalent to "oraqa1".

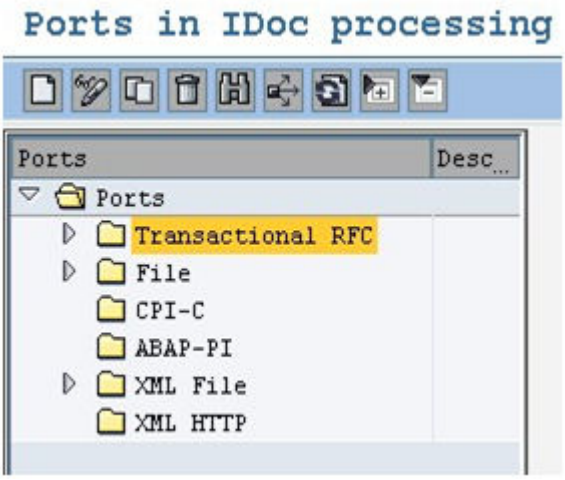
Creating a Port

The IDoc port contains the information about the way data is sent between the trigger (source) or invoke (target) systems. The type of port defines the information contained within the port. For the "Internet" port type, the port contains the IP address of the invoke system. For the "file" port type, the directory or file name information is maintained. The "tRFC" port contains information about the RFC destination of the invoke system. "tRFC" ports are used for IDoc transmission using ALE.

To create a tRFC port:

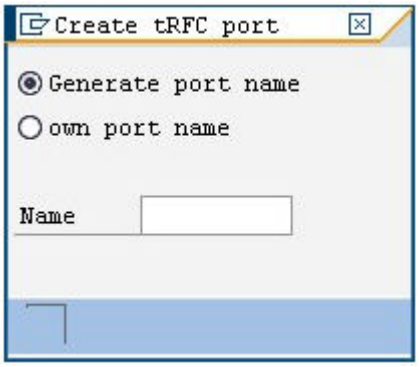
1. Run the we21 transaction.

2. Click on the **Transactional RFC** entry in the ports list.

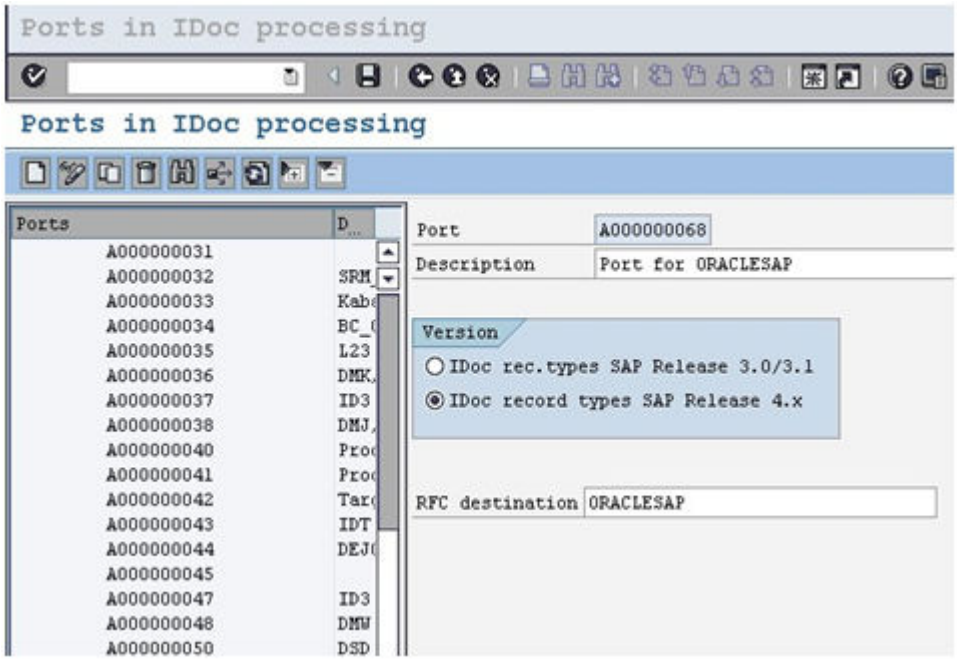


3. Click **Create**.

4. Click on the **Generate port name** radio button, or click on the **own port name** radio button and enter your own port name.



5. Enter the description in the **Description** field and the RFC destination in the **RFC destination** field.



6. Click **Save**.

Configuring a Logical System

The logical system is used to identify an individual client in a system in ALE communication between SAP systems.

The procedure for configuring an outbound logical systems is identical to the same task for inbound logical systems. See the inbound procedure at Configuring a Logical System for instructions.

Configuring a Distribution Model

The distribution model determines the sender and receiver of the IDoc's and defines the transfer rules.

The procedure for configuring an outbound distribution model is identical to the same task for inbound distribution models. See the inbound procedure at Configuring a Distribution Model for instructions.

Configuring a Partner Profile

In SAP, all partner systems involved in a distribution model have a profile. There are several profile types, such as customer profiles and vendor profiles. This distinction is generally not necessary and you usually create your partners profiles using a generic logical system type.

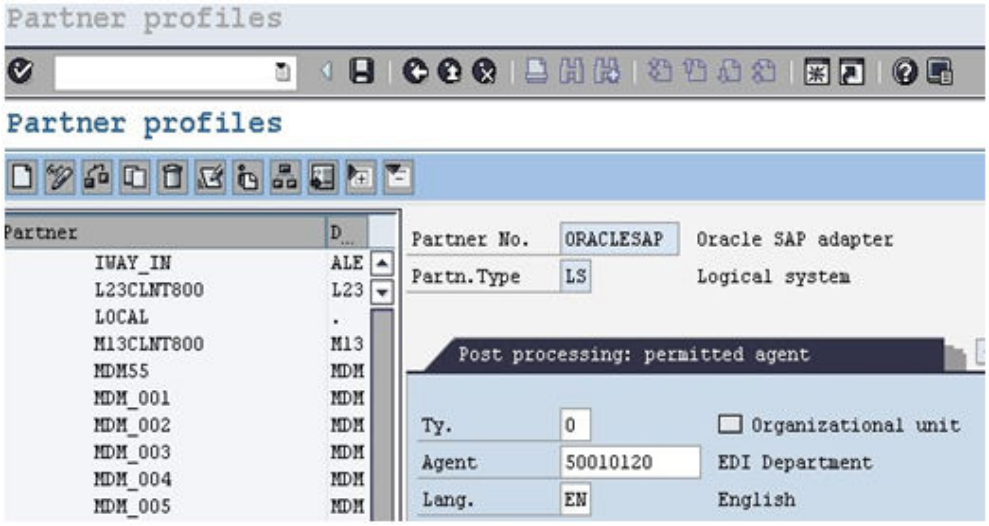
For a receiver partner system (outbound parameters are filled in), the following settings are specified in the partner profile:

- The receiver port to which the data will be sent.
- The sending method: either one IDoc at a time, or by packets.
- The IDoc type that will be sent to that partner. For a given message type, the type of IDoc might vary depending on the receiver system. You might have different versions of SAP in your system landscape.

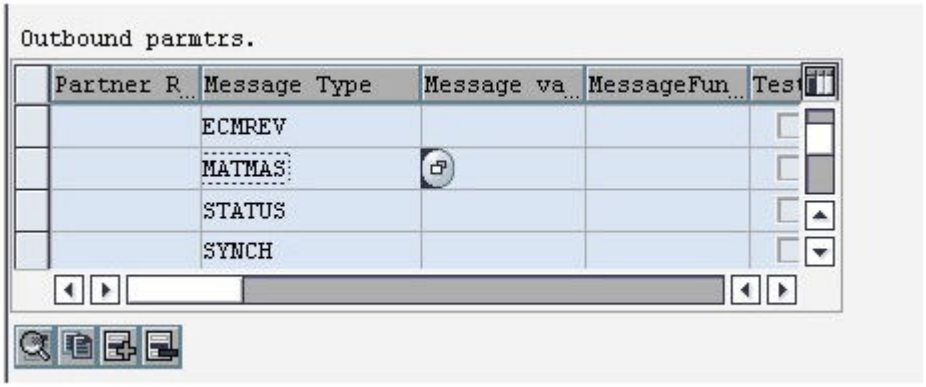
To create a partner profile:

1. Run the `we20` transaction.

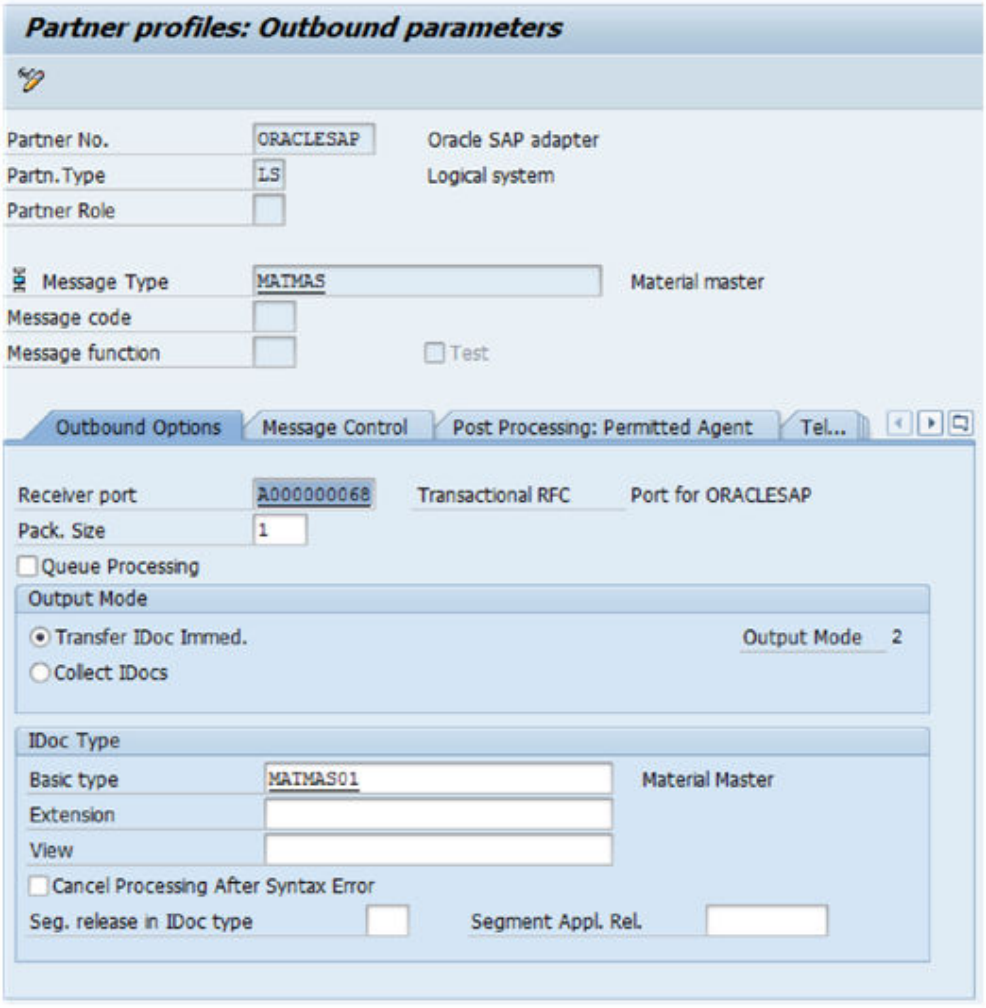
2. **Click Partner Type LS**.
3. Click **Create**.
4. Enter the Partner No. — the logical system name that was created earlier.



5. Click **Save**.
6. Click the **Add** icon to add the outbound parameters.



7. Enter the **Message Type**, **Port name** and the **Basic type** for the particular message type.



8. Click **Save**.

Summary

The inbound and outbound configurations are now ready for IDoc exchange.

When sending or receiving IDocs from SAP, you can see the inbound and outbound IDocs and their status in the SAP application window

B

Adding JAR Files to the Agent Class Path

If you use the on-premises agent with the SAP Adapter, you have to add some additional JAR files to the agent's class path.

Add the following files to the *$AGENT_HOME*`/thirdparty/lib` directory.

- `sapjco3.jar`
- `sapidoc3.jar`
- `sapjoc3.dll` (for Windows)
- `libsapjco3.so` (for Linux)

These files are provided by the SAP administrator or you can download the installation files from the SAP service marketplace under `service.sap.com/connectors`.

C

JCO Connection Properties Files

When configuring a connection on the Connections page for the SAP Adapter, you must select the JCO connection properties file to use. This appendix describes the different file types you can upload.

Adapter_inbound_Direct.properties

```
#Common properties for Load Balanced/Direct Connection Type: These parameters will
be used in both connection types.
    #Description:
        #jco.client.client    = Client represent a self-contained unit in an SAP
system with separate master records and its own set of tables. Eg: 811
        #jco.client.lang    = The language determines the code page used for
communicating between SAP Adapter and the application server. Eg: EN

jco.client.client                       =
jco.client.lang                         =

############################################################
#SAP Direct connection properties: These patameters should be defined if user
intends to access ERP Application directly without gateway.
    #Description:
        #jco.client.ashost   = ERP Application Server Host(IP). Eg: 10.30.XX.XX
        #jco.client.sysnr    = System number. Eg: 01

jco.client.ashost                       =
jco.client.sysnr                        =

############################################################
#SAP Outbound (Adapter inbound) connection properties: Parameters required to
recieve data from SAP. Particularly used for SAP Outbound scenario where SAP will
send data to adapter.
    #Description:
        #jco.server.gwhost       = Gateway host (IP). Eg: 10.30.XX.XX
        #jco.server.gwserv       = Gateway service name. Eg: sapgw00
        #jco.server.progid       = Identifier Used to get Register with SAP to
receive data. Eg: SAPPROGRAMID

jco.server.gwhost                   =
jco.server.gwserv                   =
jco.server.progid                   =
```

Adapter_inbound_Direct_SNC.properties

```
#Common properties for Load Balanced/Direct Connection Type: These parameters will
be used in both connection types.
    #Description:
        #jco.client.client    = Client represent a self-contained unit in an SAP
system with separate master records and its own set of tables. Eg: 811
        #jco.client.lang    = The language determines the code page used for
```

```
communicating between SAP Adapter and the application server. Eg: EN

jco.client.client                       =
jco.client.lang                         =

############################################################
#SAP Direct connection properties: These patameters should be defined if user
intends to access ERP Application directly without gateway.
    #Description:
        #jco.client.ashost   = ERP Application Server Host(IP). Eg: 10.30.XX.XX
        #jco.client.sysnr    = System number. Eg: 01

jco.client.ashost                       =
jco.client.sysnr                        =

############################################################
#SAP SNC connection properties: Parameters required to establish a secured
connection between Agent and SAP. When these parameters are filled in, direct
connection parameters are disregarded if present.
    #Description:
        #jco.client.snc_mode             = Enable/disable secured mode. Eg: 1 to
enable or 0 to disable.
        #jco.client.snc_partnername      = String used to generate secured
certificate in SAP server to be used by Agent. Eg:  p:CN=ER7, OU=B0020070395, OU=SAP
Web AS, O=SAP Trust Community, C=DE
        #jco.client.snc_qop              = The quality of protection level.
Available options:
        #                                   1 - Apply authentication only.
        #                                   2 - Apply integrity protection
(authentication).
        #                                   3 - Apply privacy protection (integrity
and authentication).
        #                                   8 - Apply the default protection.
        #                                   9 - Apply the maximum protection.
        #jco.client.snc_myname           = String used to generate the secured
certificate on the server on which Agent is deployed. Eg:  p:CN=HAR, OU=IT, O=CSW,
C=DE
        #jco.client.snc_lib              = Location of SNC library on the Agent.
Eg: /home/oracle/sec/libsapcrypto.so
    #Note: The respective certificates must already be exchanged between SAP and
Agent (or the machine having adapter).

jco.client.snc_mode                     =
jco.client.snc_partnername              =
jco.client.snc_qop                      =
jco.client.snc_myname                   =
jco.client.snc_lib                      =

############################################################
#SAP Outbound (Adapter inbound) connection properties: Parameters required to
recieve data from SAP. Particularly used for SAP Outbound scenario where SAP will
send data to adapter.
    #Description:
        #jco.server.gwhost       = Gateway host (IP). Eg: 10.30.XX.XX
        #jco.server.gwserv       = Gateway service name. Eg: sapgw00
        #jco.server.progid       = Identifier Used to get Register with SAP to
receive data. Eg: SAPPROGRAMID

jco.server.gwhost                       =
jco.server.gwserv                       =
jco.server.progid                       =
```

Adapter_inbound_Loadbalanced.properties

```
#Common properties for Load Balanced/Direct Connection Type: These parameters will
be used in both connection types.
    #Description:
        #jco.client.client    = Client represent a self-contained unit in an SAP
system with separate master records and its own set of tables. Eg: 811
        #jco.client.lang    = The language determines the code page used for
communicating between SAP Adapter and the application server. Eg: EN

jco.client.client                       =
jco.client.lang                         =

############################################################
#SAP Load balanced connection properties: These patameters should be defined if user
wants to access SAP system which is behind the message Server.
    #Description:
        #jco.client.group         = Group Name of the messaging server. Eg: PUBLIC
        #jco.client.r3name      = SAP system name to identify the system. Eg: R/3
        #jco.client.mshost      = The message server is responsible for
communication between SAP application servers. It passes requests from one
application server to another within the system. Eg: 10.30.XX.XXX
        #jco.client.msserv      = Name of the service in SAP Gateway HOST. Eg:
sapgw00

jco.client.group                        =
jco.client.r3name                       =
jco.client.mshost                       =
jco.client.msserv                       =

############################################################
#SAP Outbound (Adapter inbound) connection properties: Parameters required to
recieve data from SAP. Particularly used for SAP Outbound scenario where SAP will
send data to adapter.
    #Description:
        #jco.server.gwhost       = Gateway host (IP). Eg: 10.30.XX.XX
        #jco.server.gwserv       = Gateway service name. Eg: sapgw00
        #jco.server.progid       = Identifier Used to get Register with SAP to
receive data. Eg: SAPPROGRAMID

jco.server.gwhost                       =
jco.server.gwserv                       =
jco.server.progid                       =
```

Adapter_inbound_Loadbalanced_SNC.properties

```
#Common properties for Load Balanced/Direct Connection Type: These parameters will
be used in both connection types.
    #Description:
        #jco.client.client    = Client represent a self-contained unit in an SAP
system with separate master records and its own set of tables. Eg: 811
        #jco.client.lang    = The language determines the code page used for
communicating between SAP Adapter and the application server. Eg: EN

jco.client.client                       =
jco.client.lang                         =

############################################################
#SAP Load balanced connection properties: These patameters should be defined if user
wants to access SAP system which is behind the message Server.
    #Description:
```

```
        #jco.client.group         = Group Name of the messaging server. Eg: PUBLIC
        #jco.client.r3name      = SAP system name to identify the system. Eg: R/3
        #jco.client.mshost      = The message server is responsible for
communication between SAP application servers. It passes requests from one
application server to another within the system. Eg: 10.30.XX.XXX
        #jco.client.msserv      = Name of the service in SAP Gateway HOST. Eg:
sapgw00

jco.client.group                        =
jco.client.r3name                       =
jco.client.mshost                       =
jco.client.msserv                       =

############################################################
#SAP Outbound (Adapter inbound) connection properties: Parameters required to
recieve data from SAP. Particularly used for SAP Outbound scenario where SAP will
send data to adapter.
    #Description:
        #jco.server.gwhost       = Gateway host (IP). Eg: 10.30.XX.XX
        #jco.server.gwserv       = Gateway service name. Eg: sapgw00
        #jco.server.progid       = Identifier Used to get Register with SAP to
receive data. Eg: SAPPROGRAMID

jco.server.gwhost                       =
jco.server.gwserv                       =
jco.server.progid                       =

############################################################
#SAP SNC connection properties: Parameters required to establish a secured
connection between Agent and SAP. When these parameters are filled in, direct
connection parameters are disregarded if present.
    #Description:
        #jco.client.snc_mode             = Enable/disable secured mode. Eg: 1 to
enable or 0 to disable.
        #jco.client.snc_partnername      = String used to generate secured
certificate in SAP server to be used by Agent. Eg:  p:CN=ER7, OU=B0020070395, OU=SAP
Web AS, O=SAP Trust Community, C=DE
        #jco.client.snc_qop              = The quality of protection level.
Available options:
        #                                   1 - Apply authentication only.
        #                                   2 - Apply integrity protection
(authentication).
        #                                   3 - Apply privacy protection (integrity
and authentication).
        #                                   8 - Apply the default protection.
        #                                   9 - Apply the maximum protection.
        #jco.client.snc_myname           = String used to generate the secured
certificate on the server on which Agent is deployed. Eg:  p:CN=HAR, OU=IT, O=CSW,
C=DE
        #jco.client.snc_lib              = Location of SNC library on the Agent.
Eg: /home/oracle/sec/libsapcrypto.so
    #Note: The respective certificates must already be exchanged between SAP and
Agent (or the machine having adapter).

jco.client.snc_mode                     =
jco.client.snc_partnername              =
jco.client.snc_qop                      =
jco.client.snc_myname                   =
jco.client.snc_lib                      =
```

Adapter_outbound_Direct.properties

```
#Common properties for Load Balanced/Direct Connection Type: These parameters will
be used in both connection types.
    #Description:
        #jco.client.client    = Client represent a self-contained unit in an SAP
system with separate master records and its own set of tables. Eg: 811
        #jco.client.lang    = The language determines the code page used for
communicating between SAP Adapter and the application server. Eg: EN

jco.client.client                       =
jco.client.lang                         =

############################################################
#SAP Direct connection properties: These patameters should be defined if user
intends to access ERP Application directly without gateway.
    #Description:
        #jco.client.ashost   = ERP Application Server Host(IP). Eg: 10.30.XX.XX
        #jco.client.sysnr    = System number. Eg: 01

jco.client.ashost                       =
jco.client.sysnr                        =
```

Adapter_outbound_Direct_SNC.properties

```
#Common properties for Load Balanced/Direct Connection Type: These parameters will
be used in both connection types.
    #Description:
        #jco.client.client    = Client represent a self-contained unit in an SAP
system with separate master records and its own set of tables. Eg: 811
        #jco.client.lang    = The language determines the code page used for
communicating between SAP Adapter and the application server. Eg: EN

jco.client.client                       =
jco.client.lang                         =

############################################################
#SAP Direct connection properties: These patameters should be defined if user
intends to access ERP Application directly without gateway.
    #Description:
        #jco.client.ashost   = ERP Application Server Host(IP). Eg: 10.30.XX.XX
        #jco.client.sysnr    = System number. Eg: 01

jco.client.ashost                       =
jco.client.sysnr                        =

############################################################
#SAP SNC connection properties: Parameters required to establish a secured
connection between Agent and SAP. When these parameters are filled in, direct
connection parameters are disregarded if present.
    #Description:
        #jco.client.snc_mode             = Enable/disable secured mode. Eg: 1 to
enable or 0 to disable.
        #jco.client.snc_partnername      = String used to generate secured
certificate in SAP server to be used by Agent. Eg:  p:CN=ER7, OU=B0020070395, OU=SAP
Web AS, O=SAP Trust Community, C=DE
        #jco.client.snc_qop              = The quality of protection level.
Available options:
        #                                   1 - Apply authentication only.
        #                                   2 - Apply integrity protection
(authentication).
```

```
        #                                   3 - Apply privacy protection (integrity
and authentication).
        #                                   8 - Apply the default protection.
        #                                   9 - Apply the maximum protection.
        #jco.client.snc_myname          = String used to generate the secured
certificate on the server on which Agent is deployed. Eg:  p:CN=HAR, OU=IT, O=CSW,
C=DE
        #jco.client.snc_lib             = Location of SNC library on the Agent.
Eg: /home/oracle/sec/libsapcrypto.so
    #Note: The respective certificates must already be exchanged between SAP and
Agent (or the machine having adapter).

jco.client.snc_mode                    =
jco.client.snc_partnername             =
jco.client.snc_qop                     =
jco.client.snc_myname                  =
jco.client.snc_lib                     =
```

Adapter_outbound_Loadbalanced.properties

```
#Common properties for Load Balanced/Direct Connection Type: These parameters will
be used in both connection types.
    #Description:
        #jco.client.client    = Client represent a self-contained unit in an SAP
system with separate master records and its own set of tables. Eg: 811
        #jco.client.lang    = The language determines the code page used for
communicating between SAP Adapter and the application server. Eg: EN

jco.client.client                      =
jco.client.lang                        =

############################################################
#SAP Load balanced connection properties: These patameters should be defined if user
wants to access SAP system which is behind the message Server.
    #Description:
        #jco.client.group         = Group Name of the messaging server. Eg: PUBLIC
        #jco.client.r3name      = SAP system name to identify the system. Eg: R/3
        #jco.client.mshost      = The message server is responsible for
communication between SAP application servers. It passes requests from one
application server to another within the system. Eg: 10.30.XX.XXX
        #jco.client.msserv      = Name of the service in SAP Gateway HOST. Eg:
sapgw00

jco.client.group                       =
jco.client.r3name                      =
jco.client.mshost                      =
jco.client.msserv                      =
```

Adapter_outbound_Loadbalanced_SNC.properties

```
#Common properties for Load Balanced/Direct Connection Type: These parameters will
be used in both connection types.
    #Description:
        #jco.client.client    = Client represent a self-contained unit in an SAP
system with separate master records and its own set of tables. Eg: 811
        #jco.client.lang    = The language determines the code page used for
communicating between SAP Adapter and the application server. Eg: EN

jco.client.client                      =
jco.client.lang                        =
```

```
############################################################
#SAP Load balanced connection properties: These patameters should be defined if user
wants to access SAP system which is behind the message Server.
    #Description:
        #jco.client.group         = Group Name of the messaging server. Eg: PUBLIC
        #jco.client.r3name      = SAP system name to identify the system. Eg: R/3
        #jco.client.mshost      = The message server is responsible for
communication between SAP application servers. It passes requests from one
application server to another within the system. Eg: 10.30.XX.XXX
        #jco.client.msserv      = Name of the service in SAP Gateway HOST. Eg:
sapgw00

jco.client.group                        =
jco.client.r3name                       =
jco.client.mshost                       =
jco.client.msserv                       =

############################################################
#SAP SNC connection properties: Parameters required to establish a secured
connection between Agent and SAP. When these parameters are filled in, direct
connection parameters are disregarded if present.
    #Description:
        #jco.client.snc_mode             = Enable/disable secured mode. Eg: 1 to
enable or 0 to disable.
        #jco.client.snc_partnername      = String used to generate secured
certificate in SAP server to be used by Agent. Eg:  p:CN=ER7, OU=B0020070395, OU=SAP
Web AS, O=SAP Trust Community, C=DE
        #jco.client.snc_qop              = The quality of protection level.
Available options:
        #                                   1 - Apply authentication only.
        #                                   2 - Apply integrity protection
(authentication).
        #                                   3 - Apply privacy protection (integrity
and authentication).
        #                                   8 - Apply the default protection.
        #                                   9 - Apply the maximum protection.
        #jco.client.snc_myname           = String used to generate the secured
certificate on the server on which Agent is deployed. Eg:  p:CN=HAR, OU=IT, O=CSW,
C=DE
        #jco.client.snc_lib              = Location of SNC library on the Agent.
Eg: /home/oracle/sec/libsapcrypto.so
    #Note: The respective certificates must already be exchanged between SAP and
Agent (or the machine having adapter).

jco.client.snc_mode                     =
jco.client.snc_partnername              =
jco.client.snc_qop                      =
jco.client.snc_myname                   =
jco.client.snc_lib                      =
```