Oracle® Cloud
Using the SAP Adapter with Oracle Integration 3

F45603-07
October 2024

ORACLE®

Oracle Cloud Using the SAP Adapter with Oracle Integration 3,

F45603-07

Contents

1   Understand the SAP Adapter

2   Create an SAP Adapter Connection

3   Add the SAP Adapter Connection to an Integration

4 Troubleshoot the SAP Adapter

A Configure Inbound and Outbound Communication

B Add JAR Files to the Agent Class Path

C JCO Connection Properties Files

Preface

This guide describes how to configure this adapter as a connection in an integration in Oracle Integration.

> **✏ Note:**
>
> The use of this adapter may differ depending on the features you have, or whether your instance was provisioned using Standard or Enterprise edition. These differences are noted throughout this guide.

Topics:

- Audience
- Documentation Accessibility
- Diversity and Inclusion
- Related Resources
- Conventions

Audience

This guide is intended for developers who want to use this adapter in integrations in Oracle Integration.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `https://www.oracle.com/corporate/accessibility/`.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit `https://support.oracle.com/portal/` or visit `Oracle Accessibility Learning and Support` if you are hearing impaired.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and

the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Resources

See these Oracle resources:

*   Oracle Cloud at http://cloud.oracle.com

*   *Using Integrations in Oracle Integration 3*

*   *Using the Oracle Mapper with Oracle Integration 3*

*   Oracle Integration documentation on the Oracle Help Center.

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

1
Understand 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 Oracle Integration. A typical workflow of adapter and integration tasks is also provided.

Topics:

- SAP Adapter Capabilities
- What Application Version Is Supported?
- SAP Adapter Use Cases
- Workflow to Create and Add an SAP Adapter Connection in an Integration

> **Note:**
>
> There are overall service limits for Oracle Integration. A service limit is the quota or allowance set on a resource. See Service Limits.

SAP Adapter Capabilities

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

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
- All the RFCs/BAPIs of SAP and its add-in components are supported such as the SAP BPC component, and others. However, the SAP Adapter does not support deep structure's RFC/BAPI.

The SAP Adapter is one of many predefined adapters included with Oracle Integration. You can configure the SAP Adapter as a connection in an integration in Oracle Integration.

▶ Video

What Application Version Is Supported?

For information about which application version is supported by this adapter, see the Connectivity Certification Matrix.

SAP Adapter Use Cases

Common use cases for the SAP Adapter are as follows:

- Closed-loop order management between CRM applications and SAP ERP.
- Account/customer synchronization between Salesforce and SAP.
- Creation of purchase orders in SAP for requisitions in Ariba procurement.
- Employee synchronization between HCM Cloud and an SAP system.

Workflow to Create and Add an SAP 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 Oracle Integration.

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.	Create an SAP Adapter Connection
2	Create the integration. When you do this, you add trigger and invoke connections to the integration.	Create Integrations and Add the SAP Adapter Connection to an Integration
3	Map data between the trigger connection data structure and the invoke connection data structure.	Map Data in *Using Integrations in Oracle Integration 3*
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).	Manage Lookups in *Using Integrations in Oracle Integration 3*
5	Activate the integration.	Manage Integrations in *Using Integrations in Oracle Integration 3*
6	Monitor the integration on the dashboard.	Monitor Integrations During Runtime in *Using Integrations in Oracle Integration 3*
7	Track payload fields in messages during runtime.	Assign Business Identifiers for Tracking Fields in Messages and Track Integration Instances in *Using Integrations in Oracle Integration 3*
8	Manage errors at the integration level, connection level, or specific integration instance level.	Manage Errors in *Using Integrations in Oracle Integration 3*

ORACLE

2

Create an SAP Adapter Connection

A connection is based on an adapter. You define connections to the specific cloud applications that you want to integrate.

Topics:

- Prerequisites for Creating a Connection
- Create a Connection
- Upload a Certificate to Connect with External Services

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 Configure 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 Add JAR Files to the Agent Class Path.

Create a Connection

Before you can build an integration, you must create the connections to the applications with which you want to share data.

To create a connection in Oracle Integration:

1. In the navigation pane, click **Design**, then **Connections**.
2. Click **Create**.

> ✎ **Note:**
>
> You can also create a connection in the integration canvas. See Define Inbound Triggers and Outbound Invokes.

3. In the Create connection panel, select the adapter to use for this connection. To find the adapter, scroll through the list, or enter a partial or full name in the **Search** field.

4. Enter the information that describes this connection.

Element	Description
Name	Enter a meaningful name to help others find your connection when they begin to create their own integrations.
Identifier	Automatically displays the name in capital letters that you entered in the **Name** field. If you modify the identifier name, don't include blank spaces (for example, `SALES OPPORTUNITY`).
Role	Select the role (direction) in which to use this connection (trigger, invoke, or both). Only the roles supported by the 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, you'll get an error when you try to drag the adapter into the section you didn't select. For example, assume you configure a connection for the Oracle Service Cloud (RightNow) Adapter as only an **invoke**. Dragging the adapter to a **trigger** section in the integration produces an error.
Keywords	Enter optional keywords (tags). You can search on the connection keywords on the Connections page.
Description	Enter an optional description of the connection.
Share with other projects	**Note**: This field only appears if you are creating a connection in a project. Select to make this connection publicly available in other projects. Connection sharing eliminates the need to create and maintain separate connections in different projects. When you configure an adapter connection in a different project, the **Use a shared connection** field is displayed at the top of the Connections page. If the connection you are configuring matches the same type and role as the publicly available connection, you can select that connection to reference (inherit) its resources. See Add and Share a Connection Across a Project.

5. Click **Create**.

Your connection is created. You're now ready to configure the connection properties, security policies, and (for some connections) access type.

Configure Connection Properties

Enter connection information so your application can process requests.

1. Go to the **Properties** section.

2. In the **Connection parameters file** field, click **switch to upload** to upload the JCO connection properties file.

3. Click **Upload** ⬆.

    You can upload two types of JCO connection property files:

    *   Adapter inbound properties file: Use to configure the SAP Adapter for both trigger and invoke connections (for example, the `Adapter_inbound_Direct.properties` file).

    *   Adapter outbound properties file: Use to configure the SAP Adapter for invoke connections only (for example, the `Adapter_outbound_Direct.properties` file).

4. Select the JCO connection properties file to use. See JCO Connection Properties Files.

5. Click **Upload**.

Configure Connection Security

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

1. Go to the **Security** section.

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.

Configure the Endpoint Access Type

Configure access to your endpoint. Depending on the capabilities of the adapter you are configuring, options may appear to configure access to the public internet, to a private endpoint, or to an on-premises service hosted behind a fire wall.

Select the Endpoint Access Type

Select the option for accessing your endpoint.

Option	This Option Appears If Your Adapter Supports ...
Public gateway	Connections to endpoints using the public internet.

Option	This Option Appears If Your Adapter Supports ...
Connectivity agent	Connections to on-premises endpoints through the connectivity agent.
	1. Click **Associate agent group**. The Associate agent group panel appears.
	2. Select the agent group, and click **Use**.
	To configure an agent group, you must download and install the on-premises connectivity agent. See Download and Run the Connectivity Agent Installer and About Creating Hybrid Integrations Using Oracle Integration in *Using Integrations in Oracle Integration 3*.

Test the Connection

Test your connection to ensure that it's configured successfully.

1. In the page title bar, click **Test**. What happens next depends on whether your adapter connection uses a Web Services Description Language (WSDL) file. Only some adapter connections use WSDLs.

If Your Connection...	Then...
Doesn't use a WSDL	The test starts automatically and validates the inputs you provided for the connection.
Uses a WSDL	A dialog prompts you to select the type of connection testing to perform:
	• **Validate and Test**: Performs a full validation of the WSDL, including processing of the imported schemas and WSDLs. Complete validation can take several minutes depending on the number of imported schemas and WSDLs. No requests are sent to the operations exposed in the WSDL.
	• **Test**: Connects to the WSDL URL and performs a syntax check on the WSDL. No requests are sent to the operations exposed in the WSDL.
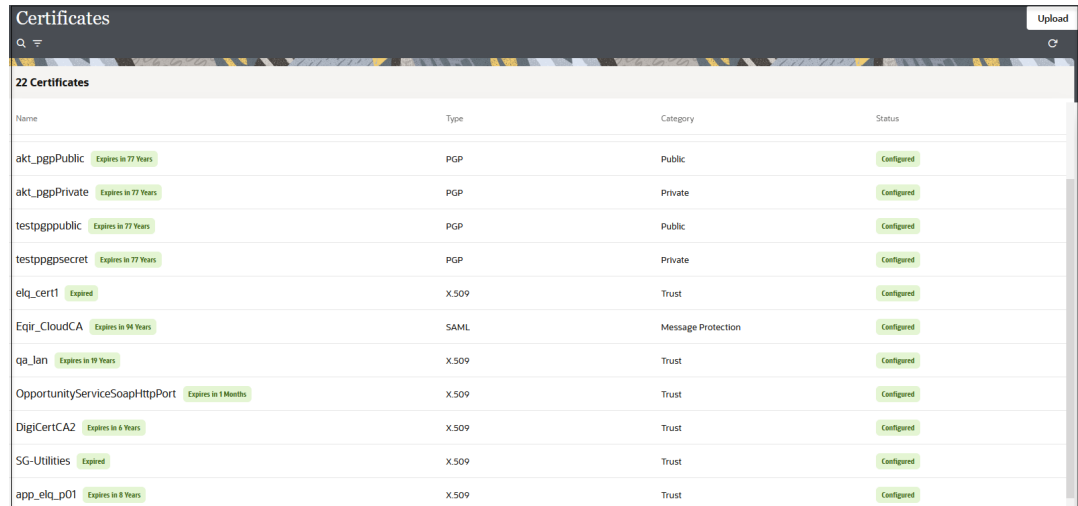
2. Wait for a message about the results of the connection test.

    • If the test was successful, then the connection is configured properly.

    • If the test failed, then edit the configuration details you entered. Check for typos and verify URLs and credentials. Continue to test until the connection is successful.

3. When complete, click **Save**.

Upload a Certificate to Connect with External Services

Certificates allow Oracle Integration to connect with external services. If the external service/ endpoint needs a specific certificate, request the certificate and then import it into Oracle Integration.

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

1. Sign in to Oracle Integration.

2. In the navigation pane, click **Settings**, then **Certificates**.
   All certificates currently uploaded to the trust store are displayed on the Certificates page.

3. Click **Filter** ▼ to filter by name, certificate expiration date, status, type, category, and installation method (user-installed or system-installed). Certificates installed by the system cannot be deleted.

4. Click **Upload** at the top of the page.
   The Upload certificate panel is displayed.

5. Enter an alias name and optional description.

6. In the **Type** field, select the certificate type. Each certificate type enables Oracle Integration to connect with external services.

   • Digital Signature

   • X.509 (SSL transport)

   • SAML (Authentication & Authorization)

   • PGP (Encryption & Decryption)

   • Signing key

Digital Signature

The digital signature security type is typically used with adapters created with the Rapid Adapter Builder. See Learn About the Rapid Adapter Builder in Oracle Integration in *Using the Rapid Adapter Builder with Oracle Integration 3*.

1. Click **Browse** to select the digital certificate. The certificate must be an X509Certificate. This certificate provides inbound RSA signature validation. See RSA Signature Validation in *Using the Rapid Adapter Builder with Oracle Integration 3*.

2. Click **Upload**.

X.509 (SSL transport)

1. Select a certificate category.

   a. **Trust**: Use this option to upload a trust certificate.

    i. Click **Browse**, then select the trust file (for example, `.cer` or `.crt`) to upload.

  b. **Identity**: Use this option to upload a certificate for two-way SSL communication.

    i. Click **Browse**, then select the keystore file (`.jks`) to upload.

    ii. Enter the comma-separated list of passwords corresponding to key aliases.

> **✎ Note:**
>
> When an identity certificate file (`.jks`) contains more than one private key, all the private keys must have the same password. If the private keys are protected with different passwords, the private keys cannot be extracted from the keystore.

    iii. Enter the password of the keystore being imported.

  c. Click **Upload**.

SAML (Authentication & Authorization)

1. Note that **Message Protection** is automatically selected as the only available certificate category and cannot be deselected. Use this option to upload a keystore certificate with SAML token support. Create, read, update, and delete (CRUD) operations are supported with this type of certificate.

2. Click **Browse**, then select the certificate file (`.cer` or `.crt`) to upload.

3. Click **Upload**.

PGP (Encryption & Decryption)

1. Select a certificate category. Pretty Good Privacy (PGP) provides cryptographic privacy and authentication for communication. PGP is used for signing, encrypting, and decrypting files. You can select the private key to use for encryption or decryption when configuring the stage file action.

  a. **Private**: Uses a private key of the target location to decrypt the file.

    i. Click **Browse**, then select the PGP file to upload.

    ii. Enter the PGP private key password.

  b. **Public**: Uses a public key of the target location to encrypt the file.

    i. Click **Browse**, then select the PGP file to upload.

    ii. In the **ASCII-Armor Encryption Format** field, select **Yes** or **No**.

      • **Yes** shows the format of the encrypted message in ASCII armor. ASCII armor is a binary-to-textual encoding converter. ASCII armor formats encrypted messaging in ASCII. This enables messages to be sent in a standard messaging format. This selection impacts the visibility of message content.

      • **No** causes the message to be sent in binary format.

    iii. From the **Cipher Algorithm** list, select the algorithm to use. Symmetric-key algorithms for cryptography use the same cryptographic keys for both encryption of plain text and decryption of cipher text. The following supported cipher algorithms are FIPS-compliant:

      • AES128

      • AES192

- AES256

- TDES

c. Click **Upload**.

Signing key

A signing key is a secret key used to establish trust between applications. Signing keys are used to sign ID tokens, access tokens, SAML assertions, and more. Using a private signing key, the token is digitally signed and the server verifies the authenticity of the token by using a public signing key. You must upload a signing key to use the OAuth Client Credentials using JWT Client Assertion and OAuth using JWT User Assertion security policies in REST Adapter invoke connections. Only PKCS1- and PKCS8-formatted files are supported.

1. Select **Public** or **Private**.

2. Click **Browse** to upload a key file.
   If you selected **Private**, and the private key is encrypted, a field for entering the private signing key password is displayed after key upload is complete.

3. Enter the private signing key password. If the private signing key is not encrypted, you are not required to enter a password.

4. Click **Upload**.

3

Add the SAP Adapter Connection to an Integration

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

These topics describe the wizard pages that guide you through configuration of the SAP Adapter as an endpoint in an integration.

> ✎ **Note:**
>
> All standard and custom objects (which are configured with the **Remote Enabled** checkbox in the SAP application) are supported in both the trigger and invoke directions.

Topics:

- Basic Info Page
- Trigger Objects and Methods Page
- Invoke Objects and Methods Properties
- Summary Page

Basic Info Page

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

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 hyphens in the name. You can't include the following characters: • No blank spaces (for example, `My Inbound Connection`) • No special characters (for example, `#;83&` or `righ(t)now4`) except underscores and hyphens • No 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.`

Trigger Objects and Methods Page

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

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.

> **Note:**
>
> It is recommended that you use a unique program ID in each integration/project for proper functioning.

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.
Objects	Select an object, such as Sales Order.
Methods	Select a method, such as CreateFromData.
Available Methods	Displays a list of methods to select. You are shown the list of available methods based on the functional area and object you selected above. You can use the arrow buttons to move the method to the **Selected Methods** list. The methods are moved to the **Selected Methods** list as you select them.
Selected Methods	Displays the list of methods you have selected.

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.
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.
Available Methods	Displays a list of methods to select. You are shown the list of available methods based on the functional area and object you selected above. You can use the arrow buttons to move the method to the **Selected Methods** list. The methods are moved to the **Selected Methods** list as you select them.
Selected Methods	Displays the list of methods you have selected.

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.
	• **Encode IDOC**
	– 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.
	• **Control Character**
	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.
Available Methods	Displays a list of methods to select. You are shown the list of available methods based on the functional area and object you selected above. You can use the arrow buttons to move the method to the **Selected Methods** list. The methods are moved to the **Selected Methods** list as you select them.
Selected Methods	Displays the list of methods you have selected.

Invoke Objects and Methods Properties

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

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.
Application Components	Expose a hierarchy of components such as sales, finance, and HR.
Functional Area	Select a functional area, such as Sales and Distribution.
Objects	Select an object, such as Sales Order.
Methods	Select a method, such as CreateFromData.
Available Methods	Displays a list of methods to select. You are shown the list of available methods based on the functional area and object you selected above. You can use the arrow buttons to move the method to the **Selected Methods** list. The methods are moved to the **Selected Methods** list as you select them.
Selected Methods	Displays the list of methods you have selected.

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.
Functional Area	Select a functional area, such as Sales and Distribution.
Methods	Select a method, such as CreateFromData.
Available Methods	Displays a list of methods to select. You are shown the list of available methods based on the functional area and object you selected above. You can use the arrow buttons to move the method to the **Selected Methods** list. The methods are moved to the **Selected Methods** list as you select them.
Selected Methods	Displays the list of methods you have selected.

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`.
Available Methods	Displays a list of methods to select. You are shown the list of available methods based on the functional area and object you selected above. You can use the arrow buttons to move the method to the **Selected Methods** list. The methods are moved to the **Selected Methods** list as you select them.
Selected Methods	Displays the list of methods you have selected.

Summary Page

3-7

You can review the specified adapter configuration values 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 **Go back**.
	To cancel your configuration details, click **Cancel**.

4

Troubleshoot the SAP Adapter

Review the following topic to learn about troubleshooting issues with the SAP Adapter.

Topics

* Error When Processing Payloads for Extended Intermediate Documents (IDOCs)

Error When Processing Payloads for Extended Intermediate Documents (IDOCs)

The following error occurs while processing a payload for an extended IDOC.

```
oracle.cloud.cpi.common.core.CpiException:
oracle.tip.adapter.sa.impl.fw.ext.org.collaxa.thirdparty.apache.wsif.WSIFExcep
tion:
file:/home/oracle/oic_connection_agent/agenthome/agent/data/f468bd10-
d06e-431e-b24e-63ab1f6dac2c/
Send_Worker_to_SAP_REQUEST.wsdl
[Send_Worker_to_SAP_REQUEST_PT::HRMD_A09.ZHRMD_A09(parameters,parameters) ]
- WSIF JCA Execute of operation 'HRMD_A09.ZHRMD_A09' failed due to:
SAP-IDC-O-INV-PL-1. □
Adapter Exception: Payload processing error.
; nested exception is:
BINDING.JCA-00001
SAP-IDC-O-INV-PL-1.
AdapterException: Payload processing error.
The payload does not correspond with the selected Idoc fault.
```

Solution: When you map an IDOC to send to SAP from Oracle Integration, ensure that you map all mandatory fields. In the mapper, map the source **CIMTYP**, **MESTYP**, and **IDOCTYP** to the target **CIMTYP**, **MESTYP**, and **IDOCTYP**. Ensure that you add/pass the value to **CIMTYP** for payloads. In addition, ensure that values are passed as follows:

* Standard IDOC type for **IDOCTYP**
* Extended IDOC for **CIMTYP**
* A standard message type for **MESTYP**

See the following examples for `MATMAS` (IDOCs) and `HRMD_A09` (IDOCs) groups.

```
<urn:IDOCTYP>MATMAS01</urn:IDOCTYP>
<!--Optional:-->
<urn:CIMTYP>ZMATMAS01_EXT</urn:CIMTYP>
```

```
<!--Optional:-->
<urn:MESTYP>MATMAS</urn:MESTYP>


<urn:IDOCTYP>HRMD_A09</urn:IDOCTYP>
<!--Optional:-->
<urn:CIMTYP>ZHRMD_EXT</urn:CIMTYP>
<!--Optional:-->
<urn:MESTYP>HRMD_A</urn:MESTYP>
```

A
Configure Inbound and Outbound Communication

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

Topics:

- SAP Inbound Communication
- SAP Outbound Communication
- Summary

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
- Configure a Logical System
- Configure a Partner Profile
- Configure Inbound Process Code
- Configure a Distribution Model

Prerequisites

Take the following actions before you begin configuration. Perform these actions on the host on which the connectivity agent is installed.

> **✎ 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.

Configure 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.
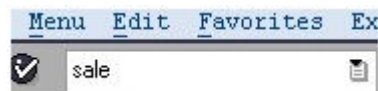
2. Open the basic settings and then the **Logical Systems** node.
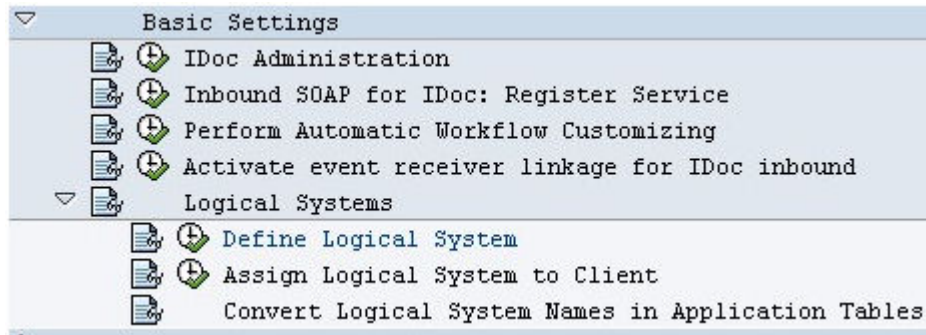
3. In the list, click **Define Logical Systems**.

   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.
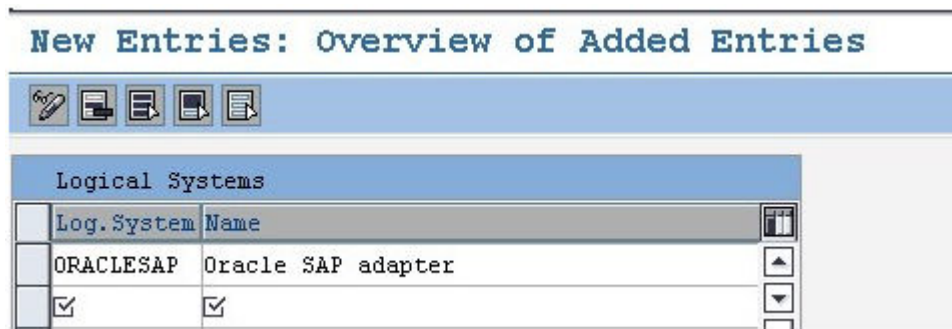
7. Click **Save**.

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

   The entry for Logical System will now be visible in the table.
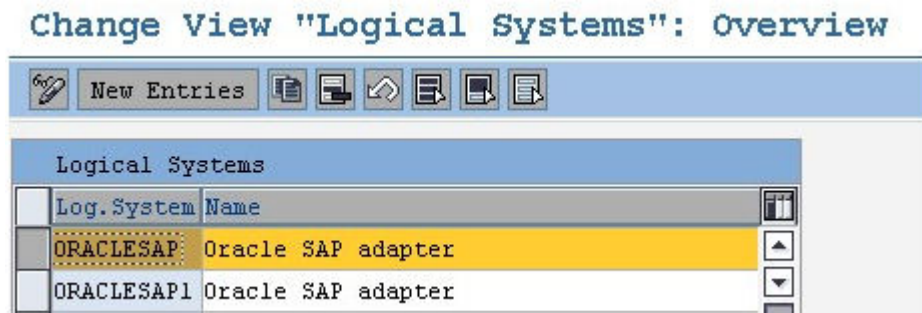
Configure a Partner Profile

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.
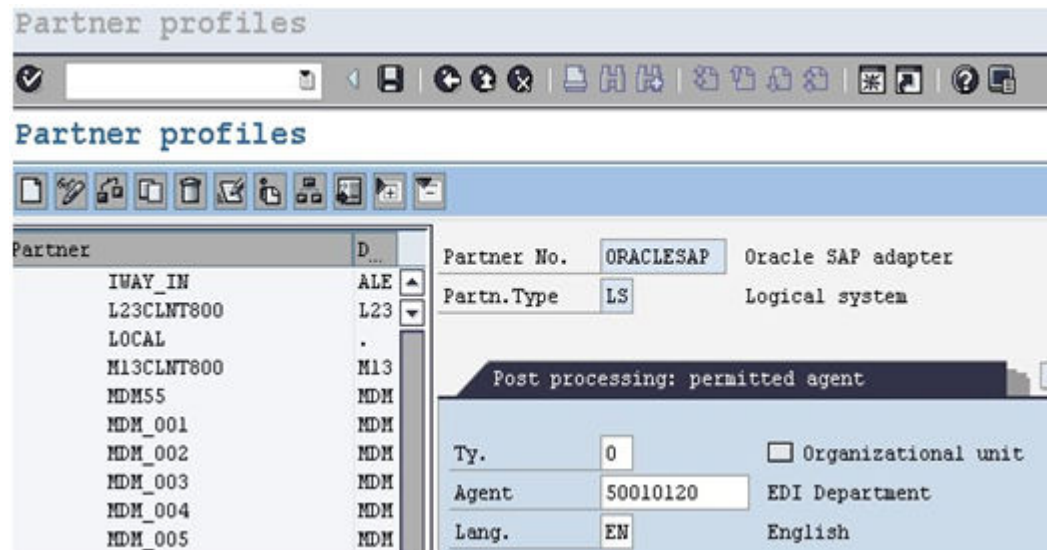
5. Click **Save**.

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.

Configure 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**.

Configure 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.

Create Model View	☒
Short text	Oracle JCA SAP Model View
Technical name	ZORACLESAP
Start date	19.11.2013
End Date	31.12.9999

5. Highlight the model view you created.

Oracle JCA SAP Model View   ZORACLESAP

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.

Add Message Type	☒
Model view	ZORACLESAP
Sender	T90CLNT090
Receiver	ORACLESAP
Message Type	MATMAS

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.

> **✎ Note:**
>
> Application Link Enabling (ALE) configuration is only required if you need to process Intermediate Documents (IDocs) communication to send from SAP and to receive in SAP. Therefore, the following activities are only applicable for processing IDocs. These activities are not applicable for BAPI and RFC communication.
>
> - Create a Port
> - Configure a Logical System
> - Configure a Distribution Model
> - Configure a Partner Profile

Topics

- Configure an RFC Destination and Program ID
- Create a Port
- Configure a Logical System
- Configure a Distribution Model
- Configure a Partner Profile

Configure 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**.
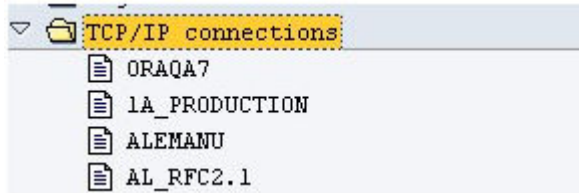
3. Click **Create**.

4. Enter the RFC destination name and the description along with the program ID and click on **Registered Server Program**.
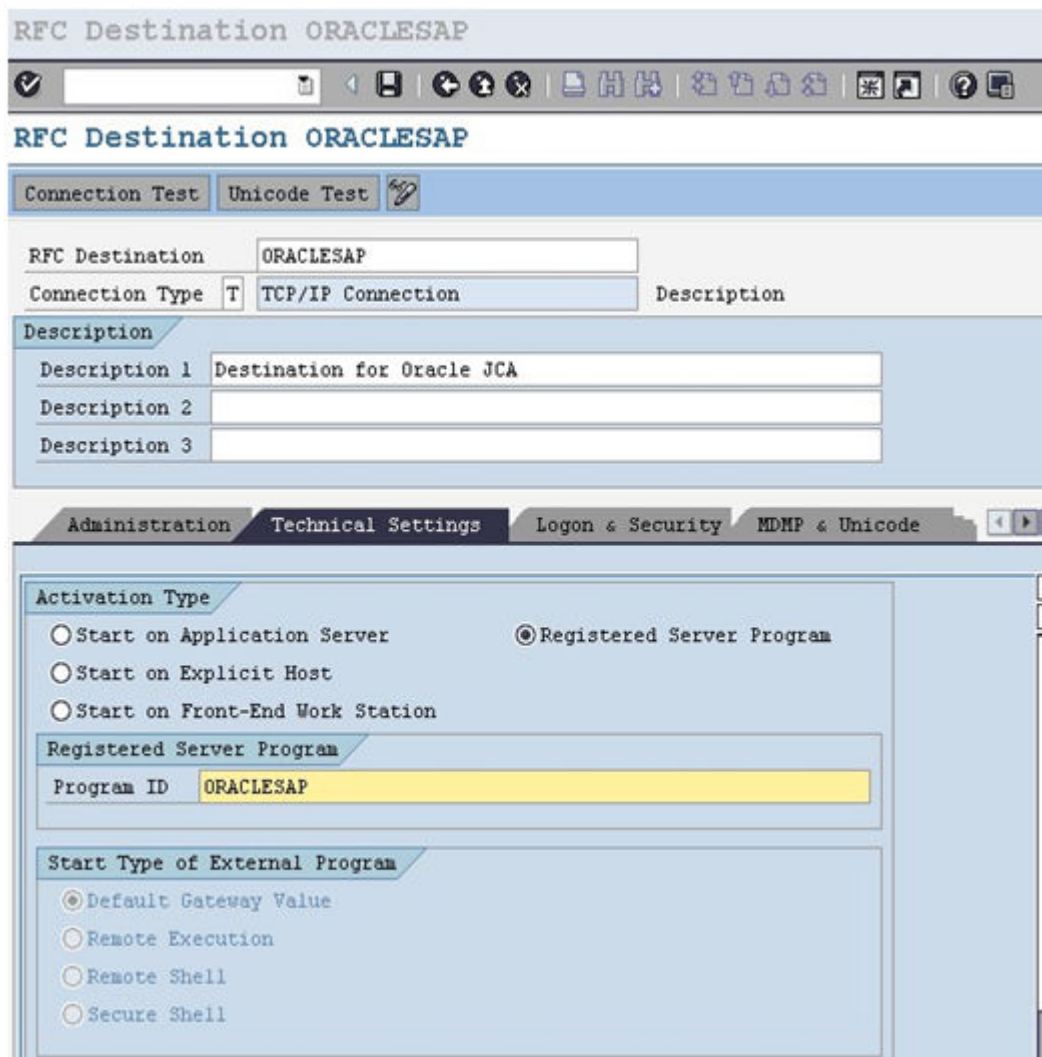
An RFC server program registers itself under the Program ID.

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

Gateway Options	
Gateway Host	bcora008
Gateway service	sapgw20

6. Click **Save**.

The RFC destination is now configured.

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

Create 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.

we21

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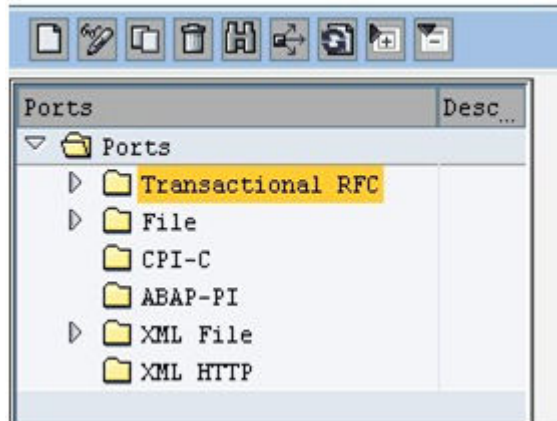
3. Click **Create**.

4. Click on the **Generate port name** radio button, or click on the **own port name** radio button and enter your own port name.
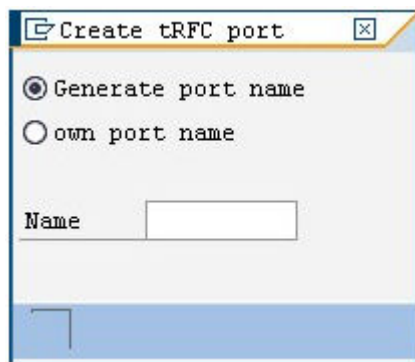
5. Enter the description in the **Description** field and the RFC destination in the **RFC destination** field.
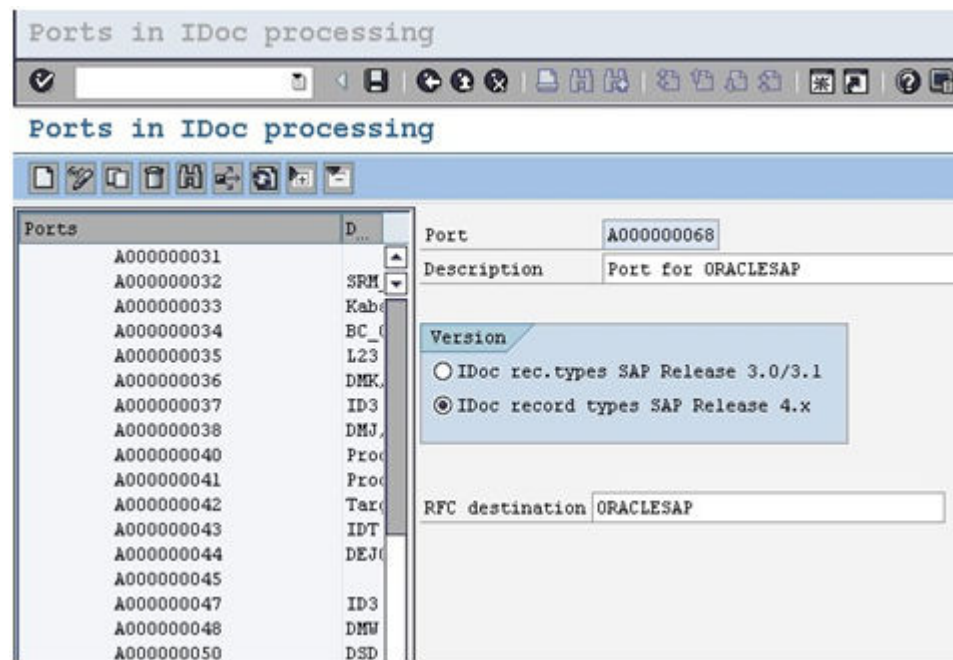
6. Click **Save**.

Configure 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 Configure a Logical System.

Configure 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 Configure a Distribution Model.

Configure 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.

we20

2. **Click Partner Type LS**.

3. Click **Create**.

4. Enter the Partner No. — the logical system name that was created earlier.
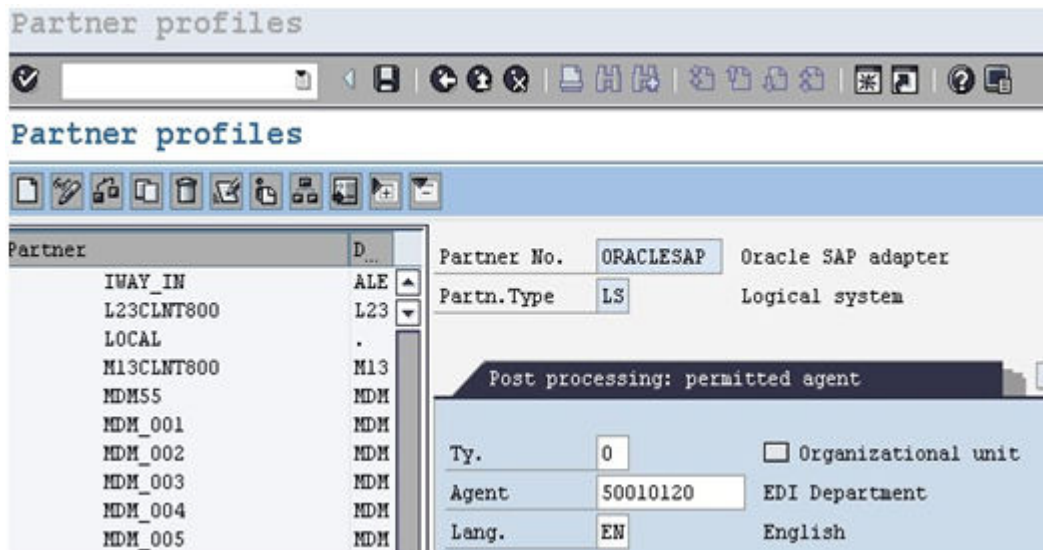
5. Click **Save**.

6. Click the **Add** icon to add the outbound parameters.
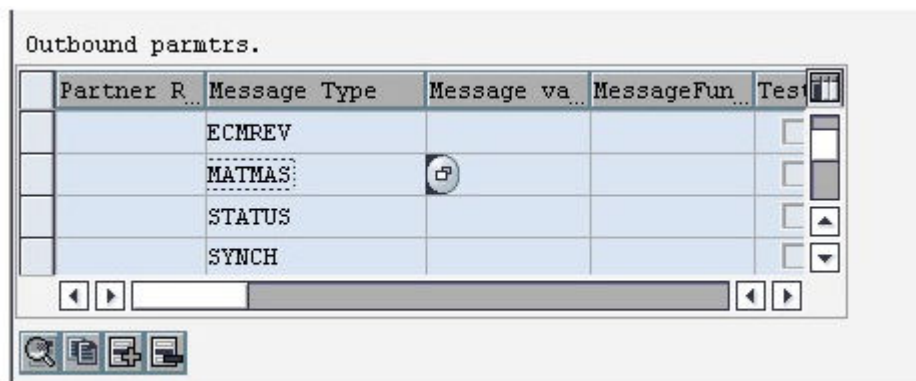
7. Enter the **Message Type**, **Port name** and the **Basic type** for the particular message type.

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

Add JAR Files to the Agent Class Path

If you use the on-premises agent with the SAP Adapter, you must 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`
- `sapjco3.dll` (for Windows)
- `libsapjco3.so` (for Linux)

These installation files are provided by the SAP administrator, or you can download the files from the SAP Support Portal. See Download SAP Java Connector 3.1 SDK.

> **✏ Note:**
>
> - It is recommended that you use the latest SAP JCo version with the SAP Adapter.
> - On Windows platforms, JCo 3.1 requires the Visual Studio 2013 C/C++ runtime libraries to be installed on your system. See Update for Visual C++ 2013 Redistributable Package.

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.

The following JCO client properties file types can be used for trigger and invoke connections, respectively:

- Adapter inbound properties file: Use this file type to configure an SAP Adapter for both trigger and invoke connections (for example, an `Adapter_inbound_Direct.properties` file).

- Adapter outbound properties file: Use this file type to configure an SAP Adapter for an invoke connection only (for example, an `Adapter_outbound_Direct.properties` file).

- Adapter_inbound_Direct.properties

- Adapter_inbound_Direct_SNC.properties

- Adapter_inbound_Loadbalanced.properties

- Adapter_inbound_Loadbalanced_SNC.properties

- Adapter_outbound_Direct.properties

- Adapter_outbound_Direct_SNC.properties

- Adapter_outbound_Loadbalanced.properties

- Adapter_outbound_Loadbalanced_SNC.properties

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

############################################################

#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.server.snc_mode          = Enable/disable secured mode. Eg: 1
to enable or 0 to disable.
        #jco.server.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.server.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.server.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.server.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.server.snc_mode         =
jco.server.snc_qop          =
jco.server.snc_myname       =
jco.server.snc_lib          =
                    =
```

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                    =

###########################################################

#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.server.snc_mode          = Enable/disable secured mode. Eg: 1
to enable or 0 to disable.
        #jco.server.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.server.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.server.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.server.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.server.snc_mode               =
jco.server.snc_qop                =
jco.server.snc_myname             =
```

```
jco.server.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                     =
```