Oracle® Cloud

Using the Oracle Database Cloud Service Adapter with Oracle Integration

E95854-14
October 2019
4  Implement Common Patterns Using the Oracle Database Cloud Service Adapter

Define a Select Operation on Database Tables 4-1
Define Fault Mapping in Orchestrated Integrations 4-4

5  Troubleshoot the Oracle Database Cloud Service Adapter

Set Null to Collections 5-1
Wrappers Require Regeneration After Objects Change 5-1
Recover from a CLOUD-0005: Unable to Establish Connection Error 5-1
Special Characters are Not Supported in Schema Names 5-2
Resolve Message Time Out Errors 5-2
Preface

This guide describes how to configure the Oracle Database Cloud Service Adapter as a connection in an integration in Oracle Integration.

Note:

The information in this guide applies to all of your Oracle Integration instances. It doesn’t matter which edition you’re using, what features you have, or who manages your cloud environment. You’ll find what you need here, including notes about any differences between the various flavors of Oracle Integration when necessary.

Topics

• Audience
• Documentation Accessibility
• Related Resources
• Conventions

Audience

This guide is intended for developers who want to use the Oracle Database Cloud Service Adapter in integrations in Oracle Integration.

Documentation Accessibility

For information about Oracle’s commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Resources

See these Oracle resources:
Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td><strong>monospace</strong></td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
1

Understand the Oracle Database Cloud Service Adapter

Review the following conceptual topics to learn about the Oracle Database Cloud Service 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:

• Oracle Database Cloud Service Adapter Capabilities
• Oracle Database Cloud Service Adapter Restrictions
• What Application Version Is Supported?
• Workflow to Create and Add an Oracle Database Cloud Service Adapter Connection to an Integration

Oracle Database Cloud Service Adapter Capabilities

The Oracle Database Cloud Service Adapter enables you to integrate Oracle Database Cloud Service with Oracle Integration through use of the connectivity agent. The Oracle Database Cloud Service Adapter provides support for integrating with one-node database systems on either bare metal or virtual machines, and two-node real application cluster (RAC) database systems on virtual machines offered by Oracle Cloud Infrastructure using the connectivity agent. Use the Oracle Database Cloud Service Adapter to poll for new and updated records for processing in Oracle Integration. For example, any new record added to the Employee table in your Oracle Database Cloud Service can be synchronized with Oracle HCM Cloud using Oracle Integration. In addition, use the Oracle Database Cloud Service Adapter to execute SQL queries or stored procedures in Oracle Database Cloud Service. For example, quotes in Oracle CPQ Cloud can be created as Orders in Oracle Database Cloud Service by sending SQL statements or stored procedures using the Oracle Database Cloud Service Adapter.

Note:

Apart from using the Oracle Database Cloud Service Adapter to integrate with Oracle Database Cloud Service, it can only be used for existing Oracle Database Cloud Service integrations to work with Oracle Autonomous Transaction Processing Cloud Service. Use the Oracle Autonomous Transaction Processing Adapter for any new integrations built to connect with an Oracle Autonomous Transaction Processing instance database. Oracle Autonomous Transaction Processing integration-related capabilities/enhancements are only introduced in the Oracle Autonomous Transaction Processing Adapter.
The Oracle Database Cloud Service Adapter provides the following capabilities:

- Support for execution of DML statements and SQL queries: Select, Insert, Update, and Delete.

Select the **Run a SQL Statement** option on the Basic Info page of the Adapter Endpoint Configuration Wizard to execute simple SQL queries. For complex SQL queries, use stored procedures by selecting the **Invoke a Stored Procedure** option on the Basic Info page of the Adapter Endpoint Configuration Wizard. Stored procedures can reduce the complexity of a SQL query.

- Support for a comprehensive database workbench for configuring SQL statements to perform **SELECT**, **INSERT**, and **UPDATE** operations on up to four tables.

- Support for non-JDBC (PL/SQL) datatypes in outbound invocations of stored procedures.

- Support for invocation of stored procedures in the Oracle database.

- Support for automatically generating and presenting the request and response data definitions in the mapper based on the SQL Query/DML statements/stored procedure configured through the adapter. This feature generates an XSD from a PureSQL statement provided by dynamically querying the table.

- Support for polling new and updated records for processing in the Oracle database. The Oracle Database Cloud Service Adapter supports distributed polling and multithreading. Distributed polling helps eliminate duplicate polling of the same records while multithreading provides optimum performance.

- Support for a logical delete polling strategy. This strategy involves updating a special field on each row once it is processed.

- Support for updating or inserting multiple records in a single request.

- Support for database fault mapping.

- Support for processing message payloads up to 10 MB in size. In the case of polling, you must set the **Rejected Value** property to **REJECTED** on the Polling Strategy and Options page. If the incoming message is greater than the 10 MB threshold size, that particular record is updated to **REJECTED** instead of **READ**. If the outbound operation returns a response greater than the 10 MB threshold size, the response message is ignored and a fault response is sent to the calling client.

**Note:**

In Java, Unicode characters are represented as 2 bytes.

- Support for connecting to the Oracle Database Cloud Service instance with SSL, thus avoiding the need to install and configure the connectivity agent. You select and configure Oracle Wallet as the security policy on the Connections page. For these scenarios, the Oracle Database Cloud Service Adapter can only be used as an invoke connection, and not as a trigger connection. See [Prerequisites for Creating a Connection](#).
The Oracle Wallet security policy can only be used when the database is SSL-enabled.

The Oracle Database Cloud Service Adapter is one of many predefined adapters included with Oracle Integration. You can configure the Oracle Database Cloud Service Adapter as a target or invoke connection in an integration in Oracle Integration.

Oracle Database Cloud Service Adapter Restrictions

Note the following Oracle Database Cloud Service Adapter restrictions in Oracle Integration.

- The PL/SQL boolean type is not supported as an IN/OUT parameter in a stored procedure if the target database is version 18c or higher. However, you can create a wrapper stored procedure that converts a PL/SQL boolean type to an integer and uses that wrapper stored procedure in Oracle Integration.

- The nested PL/SQL type (for example, a RECORD type inside a TABLE type) is not supported as an IN/OUT parameter in a stored procedure if the target database is version 18c or higher. However, you can define OBJECT types inside TABLE types.

What Application Version Is Supported?

For information about which application version is supported by this adapter, see the Oracle Integration Adapters Certification Matrix under section Oracle Integration Adapters Certification at the top of the page:

Oracle Integration Adapters Certification Matrix

Workflow to Create and Add an Oracle Database Cloud Service Adapter Connection to 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.

This table lists the workflow steps for both adapter tasks and overall integration tasks, and provides links to instructions for each step.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Access Oracle Integration.</td>
<td>Go to <a href="https://hostname:port_number/ic">https://hostname:port_number/ic</a>. Prerequisites for Creating a Connection</td>
</tr>
<tr>
<td>2</td>
<td>Perform prerequisites for creating an adapter connection.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>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.</td>
<td>Create an Oracle Database Cloud Service Adapter Connection Note: The adapter name to select in the Create Connection -Select Adapter dialog is Oracle DBaaS.</td>
</tr>
</tbody>
</table>
## Workflow to Create and Add an Oracle Database Cloud Service Adapter Connection to an Integration

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Create the integration. When you do this, you add trigger (source) and invoke (target) connections to the integration.</td>
<td>Create Integrations of Using Integrations in Oracle Integration and Add the Oracle Database Cloud Service Adapter Connection to an Integration</td>
</tr>
<tr>
<td>5</td>
<td>Map data between the trigger connection data structure and the invoke connection data structure.</td>
<td>Map Data of Using Integrations in Oracle Integration</td>
</tr>
<tr>
<td>6</td>
<td>(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).</td>
<td>Manage Lookups of Using Integrations in Oracle Integration</td>
</tr>
<tr>
<td>7</td>
<td>Activate the integration.</td>
<td>Activate Integrations of Using Integrations in Oracle Integration</td>
</tr>
<tr>
<td>8</td>
<td>Monitor the integration on the dashboard.</td>
<td>Monitor Integrations of Using Integrations in Oracle Integration</td>
</tr>
<tr>
<td>9</td>
<td>Track payload fields in messages during runtime.</td>
<td>Assign Business Identifiers for Tracking Fields in Messages and Manage Business Identifiers for Tracking Fields in Messages of Using Integrations in Oracle Integration</td>
</tr>
<tr>
<td>10</td>
<td>Manage errors at the integration level, connection level, or specific integration instance level.</td>
<td>Manage Errors of Using Integrations in Oracle Integration</td>
</tr>
</tbody>
</table>
Create an Oracle Database Cloud Service 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

Prerequisites for Creating a Connection

You must satisfy the following prerequisites to create a connection with the Oracle Database Cloud Service Adapter.

- Note details such as Public IP, SID, SQL*Net Port, PDB Name, and Connect String from the instance details page for the Oracle Database Cloud Service instance in the My Services Console. This information is required when configuring an Oracle Database Cloud Service Adapter connection on the Connections page.
- If using connectivity with SSL, note the following requirements:
  
  Note: Connectivity agent installation and agent group creation are not required when using direct connectivity.

  - The Oracle Database Cloud Service instance is accessible through a public IP address.
  - Oracle Database Cloud Service should be SSL-enabled along with the necessary wallet configuration. See the Oracle Database documentation for configuring secure sockets layer authentication for your respective database version. For example, for Release 18, see Configuring Secure Sockets Layer Authentication of the Security Guide.
  - Once SSL is configured, the following files are available under the wallet directory:

    * cwallet.sso
    * ewallet.p12

    Follow these steps to create a wallet archive file:
1. Generate the truststore and keystore using `orapki`:

   ```
   orapki wallet pkcs12_to_jks -wallet wallet_directory/ewallet.p12
   -pwd password_provided_during_wallet_creation
   -jksKeyStoreLoc wallet_directory/keystore.jks
   -jksKeyStorepwd password
   -jksTrustStoreLoc wallet_directory/truststore.jks
   -jksTrustStorepwd password
   ```

2. Create a ZIP file containing `cwallet.sso`, `ewallet.p12`, `keystore.jks`, and `truststore.jks`, including the `tnsnames.ora` file found in the `$ORACLE_HOME/network/admin/` directory.

- If using the Username Password Token security policy, note the following requirements:
  - Create an agent group. See Creating an Agent Group of Using Integrations in Oracle Integration. Note the agent group name and agent group identifier that you specify. The agent group name is required when creating an Oracle Database Cloud Service Adapter connection. The agent group identifier is required when installing the connectivity agent.
  - Download and install the connectivity agent on the same host as the Oracle Database Cloud Service or a compute node from which the database is accessible. See Downloading and Running the On-Premises Agent Installer of Using Integrations in Oracle Integration.

- If using Oracle Cloud Infrastructure Database version 18c, you must perform the following steps before using stored procedures in Oracle Cloud Infrastructure (that is, before generating wrapper packages). Some changes to the 18c database must be bypassed to get table/record type metadata information.

Run the first command to enable events to capture metadata:

   ```
   alter session set events '10946 trace name context forever, level 65536';
   ```

Run the second command to recompile and populate that metadata:

   ```
   alter package package_name compile;
   ```

Migrate from an Oracle Database Cloud Service Database Instance to an Oracle Autonomous Transaction Processing Database Instance

Perform the following steps if you want to migrate from an Oracle Database Cloud Service database instance to an Oracle Autonomous Transaction Processing database instance.

1. Migrate all the required database objects, stored procedures, wrapper procedures, and tables to the destination Oracle Autonomous Transaction Processing database instance.

2. Change the Oracle Database Cloud Service Adapter connection details to point to an Oracle Autonomous Transaction Processing database instance.
a. Go to the Connection page for the Oracle Database Cloud Service Adapter.
b. Click **Configure Connectivity**.
c. Specify the new host name.
d. Specify the new service name and click **OK**.
e. Click **Configure Security**.
f. Select the **Oracle Wallet** security policy.
g. Upload the wallet.
h. Specify the wallet password and reconfirm it.
i. Specify the database service username.
j. Specify the database service password, reconfirm it, and click **OK**.
k. Delete the agent from the connection.

3. Test the connection.
4. Once the test is successful, click **Save** to save the connection details.
5. Reactivate the integrations.

**Create 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. Click **Create**.

   ![Note:](image)

   > You can also create a connection in the integration canvas of:
   > * An orchestrated integration (See Define Inbound Triggers and Outbound Invokes.)
   > * A basic routing integration (See Add a Trigger (Source) Connection.)

   The Create Connection — Select Adapter dialog is displayed.

2. 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 Create New Connection dialog is displayed.

3. 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, Sales Opportunity).
   - 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 Service Cloud (RightNow) Adapter as only an invoke, but drag the adapter to the trigger section).

- Enter an optional description of the connection.

4. **Click Create.**

   Your connection is created and you are now ready to configure connection details, such as email contact, connection properties, security policies, connection login credentials, and (for certain connections) agent group.

**Add a Contact Email**

You can add an optional contact email address for notifications.

1. In the **Email Address** field, enter an optional email address. You do not receive automatic notifications at this address.

2. In the upper right corner, click **Save**.

**Configure Connection Properties**

Enter connection information so your application can process requests.

1. **Click Configure Connectivity.**

   The Connection Properties dialog is displayed. Specify the following Oracle Database Cloud Service instance details that you obtained in Prerequisites for Creating a Connection.

2.
3. In the **Host** field, specify the host.
4. In the **Port** field, specify the SQL*Net port.
5. In the **SID** field, specify the database SID.
6. In the **Service Name** field, specify the database service name.

**Note:**
You must specify *either* a SID or service name value. *Do not* specify values for both fields.

7. Click **OK**.
8. Configure connection security.

**Configure Connection Security**

Configure security for your Oracle Database Cloud Service Adapter connection by selecting the security policy.

1. Click **Configure Credentials**.
2. Select the security policy.
3. If you select **Username Password Token**:
   a. Enter the database username and password to connect to Oracle Database Cloud Service.
   b. Reenter the password a second time.
4. If you select **Oracle Wallet**:

**Note:**
The Oracle Database Cloud Service Adapter cannot be used as a trigger connection if configured with **Oracle Wallet**. However, all operations that you select on the Basic Info page such as **Run a SQL Statement**, **Invoke a Stored Procedure**, and **Perform an Operation On a table** are supported when configuring the adapter as an invoke connection.

   a. In the **Wallet** field, select the check box, then click **Upload** to upload the wallet file.
   b. Enter the wallet password, then re-enter it a second time to confirm.
   c. Enter the database username and password to connect to Oracle Database Cloud Service.
   d. Enter the database password a second time to confirm.
5. Click **OK**.
Configure an Agent Group

Configure an agent group for accessing the service hosted on your premises behind the fire wall.

1. Click Configure Agents.
   
The Select an Agent Group page appears.

2. Click the name of the agent group.

3. Click Use.

To configure an agent group, you must download and install the on-premises connectivity agent. See Download and Run the On-Premises Agent Installer and About Agents and Integrations Between On-Premises Applications and Oracle Integration in Using Integrations in Oracle Integration.

Test the Connection

Test your connection to ensure that it is successfully configured.

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

2. If your adapter connection uses a WSDL, you are prompted 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.

   If successful, the following message is displayed and the progress indicator shows 100%.
   
   Connection *connection_name* was tested successfully.

3. If your connection was unsuccessful, an error message is displayed with details. Verify that the configuration details you entered are correct.

4. When complete, click Save, then click Close.
Add the Oracle Database Cloud Service Adapter Connection to an Integration

When you drag the Oracle Database Cloud Service Adapter into the trigger or invoke area of an integration, the Adapter Endpoint Configuration Wizard appears. This wizard guides you through the configuration of the Oracle Database Cloud Service Adapter endpoint properties.

These topics describe the wizard pages that guide you through configuration of the Oracle Database Cloud Service Adapter as a trigger or invoke in an integration.

Topics:

- Basic Information Page
- Trigger Polling Page
- Invoke Store Procedure Page
- Invoke SQL Statement Page
- Table Operation Page
- Summary Page

Note:

Apart from using the Oracle Database Cloud Service Adapter to integrate with Oracle Database Cloud Service, it can only be used for existing Oracle Database Cloud Service integrations to work with Oracle Autonomous Transaction Processing Cloud Service. Use the Oracle Autonomous Transaction Processing Adapter for any new integrations built to connect with an Oracle Autonomous Transaction Processing instance database. Oracle Autonomous Transaction Processing integration-related capabilities/enhancements are only introduced in the Oracle Autonomous Transaction Processing Adapter.
Basic Information Page

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

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>What do you want to call your endpoint?</strong></td>
<td>Identifies the connection with a meaningful name that defines the purpose of connection. For example, CreateEmployeeInDB for a database connection that adds new employee data. The name can include English alphabetic characters, numbers, underscores, and dashes. The name cannot include:</td>
</tr>
<tr>
<td></td>
<td>• Blank spaces (for example, My DB Connection)</td>
</tr>
<tr>
<td></td>
<td>• Special characters (for example, #;83&amp; or righ(t)now4)</td>
</tr>
<tr>
<td></td>
<td>• Multibyte characters</td>
</tr>
<tr>
<td><strong>What does this endpoint do?</strong></td>
<td>Provide a description of what this connection does.</td>
</tr>
<tr>
<td><strong>What operation do you want to perform?</strong></td>
<td>If configuring the connection in the invoke direction, select the operation to perform.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Invoke a Stored Procedure</strong> — Select to run a stored procedure on the database.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Run a SQL Statement</strong> — Select to run a SQL query on the database.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Perform an Operation On a Table</strong> — Select to perform one of the following operations on a table. You can update or insert multiple records in a single request.</td>
</tr>
<tr>
<td></td>
<td>– <strong>Insert</strong></td>
</tr>
<tr>
<td></td>
<td>– <strong>Update</strong></td>
</tr>
<tr>
<td></td>
<td>– <strong>Insert or Update (Merge)</strong></td>
</tr>
<tr>
<td><strong>Notes</strong></td>
<td>• When operations in a SQL statement such as Update, Concat, and Merge accept values for the inbound invocation of an integration, they do not work. For example, the following query does not work:</td>
</tr>
<tr>
<td></td>
<td>select concat(empname, 'ss') from DB_AQ</td>
</tr>
<tr>
<td></td>
<td>where empno=#empno</td>
</tr>
<tr>
<td></td>
<td>select empno from DB_AQ where</td>
</tr>
<tr>
<td></td>
<td>empname=concat(#empname, 'YY')</td>
</tr>
<tr>
<td></td>
<td>As a workaround, handle these scenarios during payload mapping. For example, perform a concatenation during mapping of the payload. The final output can then be passed as input to the SQL query.</td>
</tr>
<tr>
<td></td>
<td>• <strong>IN/BETWEEN</strong> operators are not supported with bind parameters. Use greater than (&gt;) and less than (&lt;) operators instead.</td>
</tr>
</tbody>
</table>
Trigger Polling Page

You can poll for new or changed records.

Note:
No order is maintained while polling records.

- Polling Page
- Search and Import Tables Page
- Primary Keys Page
- Manage Table Relationships Page
- Attribute Filtering Page
- Polling Strategy and Options Page

Polling Page

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

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import Tables</td>
<td>Import the tables and select the root database table for the service query.</td>
</tr>
<tr>
<td>Remove Tables</td>
<td>Remove the selected table from the service query tables list.</td>
</tr>
<tr>
<td>Review and manage root database</td>
<td>Appears after importing tables. Select Edit to open the Manage Table Relationships page where you can view, create, and remove relationships between tables.</td>
</tr>
<tr>
<td>table relationships</td>
<td></td>
</tr>
<tr>
<td>Review and verify table and relationship attributes</td>
<td>Appears after importing tables. Select Edit to open the Attributes Filtering page where you can review, verify, select or deselect the attributes in the object model created from the imported tables and the defined relationships.</td>
</tr>
<tr>
<td>Review the polling strategy and specify polling options</td>
<td>Appears after importing tables. Select Edit to open the Polling Strategy and Options page where you can define the polling strategy and specify polling options.</td>
</tr>
</tbody>
</table>
Search and Import Tables Page

The following table describes the key information on the Search and Import Tables page. The Search and Import Tables page appears when you select **Import Tables** on the Polling page.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Schema</strong></td>
<td>Select the schema for the tables and views you are importing. Special characters (for example, #) are not supported in schema names.</td>
</tr>
</tbody>
</table>
| **Table Name** | The name of the table to which the schema or view is applied. The **Table Type** list next to the **Table Name** field allows these selections:  
  - **All** — Select all available tables and views.  
  - **Materialized View** — Select materialized views.  
  - **Materialized View Log** — Select materialized view logs.  
  - **Synonym** — Select the alias for the schema object.  
  - **Table** — Select tables.  
  - **View** — Select views.  |
| **Search**  | Click to search for the specified table type. |
| **Available** | Lists the elements of the table type available for selection. |
| **Selected** | Lists the selected table type. |

Primary Keys Page

The following table describes the key information on the Define Primary Keys page.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Primary Keys** | This page appears when you select tables without a primary key defined. Select the virtual primary keys for the table type.  
**Note:** Having the primary key at the database level is the best practice. |
Manage Table Relationships Page

The following table describes the key information on the Manage Table Relationships page. The Manage Table Relationships page appears when you select Edit for the Review and manage root database table relationships option on the Polling page.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
</table>
| Create New | Opens the Create Relation page with these options:  
- **Parent Table** — Select the parent table for the relationship between tables.  
- **Child Table** — Select the child table for the relationship between tables.  
- **Relationship** — Defines the relationship between the parent and child tables.  
- **Attribute Name** — Apply attributes to the table relationship.  
- **Mapping** — Provide the mapping for the table relationship. |
| Detach | Opens the Relationships list in a new window. |

Attribute Filtering Page

The following table describes the key information on the Attribute Filtering page. This page appears when you click Edit for the Review and verify table and relationship attributes option on the Polling page.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes View</td>
<td>Review the attributes in the object model created from the imported tables and the defined relationships and uncheck any attributes to exclude from the database queries. Primary key attributes cannot be excluded.</td>
</tr>
</tbody>
</table>

Polling Strategy and Options Page

The following table describes the key information on the Polling Strategy and Options page. The Polling Strategy and Options page appears when you select Edit for Review the polling strategy and specify polling options on the Polling page.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical Delete Field</td>
<td>Selects a field in the root database table. To allow the selection, polling must be enabled in the Status column.</td>
</tr>
<tr>
<td>Read Value</td>
<td>Identifies the value that is used to indicate a row has been read. For example, PROCESSED. Surrounding quotes are not required.</td>
</tr>
<tr>
<td>Unread Value</td>
<td>Indicates the rows to process. Only rows with Logical Delete Field and column values that match the Unread Value are read.</td>
</tr>
</tbody>
</table>
Rejected Value

Set to **REJECTED**. If the incoming message is greater than the 10 MB threshold size, that particular record is updated to **REJECTED** instead of **READ**. If the outbound operation returns a response greater than the 10 MB threshold size, the response message is ignored and a fault response is sent to the calling client.

Polling Frequency (Sec)

Specifies the polling frequency (in seconds) for new records or events.

---

**Invoke Store Procedure Page**

Enter the stored procedure properties.

---

**Note:**

- Stored procedures return binary large objects (for example, BLOB database data types) as base64Binary types in XML. Depending upon the use cases, these can be decoded during transformation using inbuilt functions such as decodeBase64 or can be passed as-is for downstream processing.
- Adapter input/output parameters are defined based on the stored procedure IN/OUT parameters. The IN parameter corresponds to the request and the OUT parameter is translated as the response.

---

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select Schema</td>
<td>Select a database schema from the list. This action refreshes the page to display fields for selecting a package or procedure to invoke. Special characters (for example, #) are not supported in schema names.</td>
</tr>
<tr>
<td>Select Package</td>
<td>Select the database package. This action refreshes the page to display the procedures available for the package. When importing a predefined integration package containing PLS or SQL stored procedures, the wrapper package is not recreated in the target database. To add the wrapper package, confirm JPublisher is installed on the target database and define the original stored procedure. After confirming JPublisher is installed and the stored procedure is defined, open the PL/SQL Wrapper utility and execute the add scripts command to add the scripts included in the exported inventory archives (IAR) file.</td>
</tr>
<tr>
<td>Select Procedure</td>
<td>Displays the in (inbound), out (outbound), and in/out (inbound/outbound) parameters for the selected package.</td>
</tr>
<tr>
<td>Arguments</td>
<td>Display the in, out, and in/out parameters that are passed with this procedure.</td>
</tr>
</tbody>
</table>
Invoke SQL Statement Page

Enter the SQL statement properties.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQL Query</td>
<td>Enter a SQL query.</td>
</tr>
<tr>
<td>Status</td>
<td>After you click Validate SQL Query, ensure the Status field shows a Success message.</td>
</tr>
</tbody>
</table>

Table Operation Page

You can update or insert multiple records in a single request.

Topics:
- Import Tables Page
- Relationships Page
- Create Relationship Page
- Attribute Filtering Page
- Advanced Options Page
- Operations on Table Page

Import Tables Page

Filter and select the tables to import based on the selected schema. These tables are used to generate a SQL statement based on the operation selected.

You can import the following number of tables:
- A maximum of three tables for insert, update, and insert or update actions
- A maximum of five tables for the select - operation on table feature
- A maximum of five tables for the polling feature

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schema</td>
<td>Select the schema to use. The page is refreshed to display the tables available for selection.</td>
</tr>
<tr>
<td>Name Filter</td>
<td>Filter the display of tables.</td>
</tr>
<tr>
<td>Available</td>
<td>Select the tables on which to insert or update records.</td>
</tr>
<tr>
<td>Selected</td>
<td>Displays the selected tables.</td>
</tr>
</tbody>
</table>
Relationships Page

Review the relationships between the selected tables and optionally create, remove, or rename relationships. These relationships are used in the insert or update SQL statements.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationships Table</td>
<td>Displays the relationships defined on the root database table and any related tables (one-to-one or one-to-many).</td>
</tr>
<tr>
<td>Create</td>
<td>Click to create new relationships.</td>
</tr>
<tr>
<td>Remove</td>
<td>Click to remove a selected relationship.</td>
</tr>
<tr>
<td>Rename</td>
<td>Click to rename a selected relationship.</td>
</tr>
</tbody>
</table>

Create Relationship Page

Specify the parent and child relationships to use in the SQL statement.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Table</td>
<td>Select the parent table.</td>
</tr>
<tr>
<td>Child Table</td>
<td>Select the child table.</td>
</tr>
<tr>
<td>Mapping Type</td>
<td>Select the mapping type (one-to-many, one-to-one, or one-to-one with the foreign key on the child table). For example, if you selected Employees as the parent table and Departments as the child table, the following options are displayed: • Employees has a 1:1 Relationship with Departments • Employees has a 1:1 Relationship with Departments (Foreign Key on Child table) • Employees has a 1:M Relationship with Departments</td>
</tr>
<tr>
<td>Parent and Child Table</td>
<td>Associate the foreign key fields to the primary key fields.</td>
</tr>
<tr>
<td>Relationship Name</td>
<td>Optionally name the relationship (a default name is generated).</td>
</tr>
</tbody>
</table>

Attribute Filtering Page

Filter out the attributes to exclude.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes Tree</td>
<td>Deselect any attributes to exclude from the database query. You cannot exclude primary key attributes.</td>
</tr>
</tbody>
</table>
Advanced Options Page

Provide additional advanced options such as sequencing. This is only valid for the insert and merge operations.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>Displays the selected table.</td>
</tr>
<tr>
<td>Sequence</td>
<td>Specify that the primary key is assigned from a sequence on any insert. Click <strong>Search</strong> and select a sequence from the list.</td>
</tr>
</tbody>
</table>

Operations on Table Page

Select the database tables. To use the bulk extract feature, you must choose the `SELECT` operation from the **Perform an Operation On a Table** list on the Basic Info page.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schema</td>
<td>Select the database schema that includes the tables to process.</td>
</tr>
<tr>
<td>Table Name</td>
<td>Enter a filter with which to search the schema (for example, %TAB to search for tables with TAB in the name).</td>
</tr>
</tbody>
</table>
| Table Type       | Specify the table type filter to get a subset of the appropriate database objects, then click **Search**.  
|                  | • **ALL**  
|                  | • **MATERIALIZED VIEW**  
|                  | • **MATERIALIZED VIEW LOG**  
|                  | • **SYNONYM**  
|                  | • **TABLE**  
|                  | • **VIEW**  
| Filter By        | Enter the initial letters to filter the display of table names.              |
| Table Names      | Select the tables to import.                                               |
|                  | **Note**: It is recommended that you import the tables together for the adapter to automatically recognize the relationship. If you import the tables separately, you must explicitly create the table relationship. |
| Import Tables    | Click to import the tables. The page is refreshed for you to select the parent database table. |
| Select the parent database table | Select the parent (root) table from the list. If using multiple related tables, this is the top-level parent table in the relationship. After making your selection, the page is refreshed for you to view and edit the table relationships. |
| Add || Remove Tables | Click to add more tables or remove tables no longer in use. |
Review and manage parent database table relationships

Click **Edit** to view and edit the table relationships. The relationships automatically identified by the adapter are displayed. See Review and manage parent database table relationships Option.

Review and filter columns from selected database tables

Click **Edit** to view and edit the table attributes. You can deselect any attributes to exclude from the database queries. Primary key attributes cannot be excluded. See Review and filter columns from selected database tables Option.

Review and edit SQL query

Click **Edit** to view and edit the default SQL query. See Review and edit SQL query Option.

**Note:** This field is available for a **Select** operation on the table.

### Review and manage parent database table relationships Option

**Table 3-1** - Review and manage parent database table relationships Option

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create New Relations</td>
<td>Click to create a new relationship.</td>
</tr>
<tr>
<td></td>
<td>View the existing parent and child table relations automatically created by the adapter.</td>
</tr>
</tbody>
</table>

### Review and filter columns from selected database tables Option

**Table 3-2** - Review and filter columns from selected database tables Option

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes Tree</td>
<td>View and deselect attributes automatically created by the adapter.</td>
</tr>
</tbody>
</table>

### Review and edit SQL query Option

**Note:**

This is only applicable for a **Select** operation on a table.

**Table 3-3** - Review and edit SQL query Option

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQL Edit</td>
<td>Click to manually edit the query in the SQL Query field.</td>
</tr>
</tbody>
</table>
Table 3-3  (Cont.) - Review and edit SQL query Option

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit using Expression Builder</td>
<td>Click to edit the query in the Expression Builder.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Add New</strong>: Click to add new criteria to the SQL query.</td>
</tr>
<tr>
<td></td>
<td>1. Click <strong>Add New</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. In the <strong>First Argument</strong> field, click <strong>Edit</strong>, and select the argument to add (for example, <code>deptno</code>).</td>
</tr>
<tr>
<td></td>
<td>3. In the <strong>Operator</strong> field, select the operator to use for the comparison from the dropdown list (for example, <code>=</code>).</td>
</tr>
<tr>
<td></td>
<td>4. In the <strong>Second Argument</strong> field, select the option to use:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Literal</strong>: Click to specify a value. If selected, you are prompted to select the data type (for example, integer) and specify the value.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Parameter</strong>: Click to specify a bind parameter.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Query Key</strong>: Click to run the comparison against another column in the table.</td>
</tr>
<tr>
<td></td>
<td>New criteria is appended to the SQL query with a <strong>WHERE</strong> clause. If you add subsequent SQL queries, they are appended to the SQL query with an <strong>AND</strong> clause.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Add Nested</strong>: Click to add nested criteria to the SQL query.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Edit</strong>: Click the edit the SQL criteria you specified.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Remove</strong>: Click the edit the SQL criteria you specified.</td>
</tr>
<tr>
<td></td>
<td>Click to edit the query with the Expression Builder.</td>
</tr>
<tr>
<td><strong>Maximum Number of Records to be fetched</strong></td>
<td>Select the number of records to fetch with this SQL query.</td>
</tr>
</tbody>
</table>
You can review the specified adapter configuration values on the Summary page.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary</td>
<td>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. Click Cancel to cancel your configuration details.</td>
</tr>
</tbody>
</table>
Implement Common Patterns Using the Oracle Database Cloud Service Adapter

You can use the Oracle Database Cloud Service Adapter to implement the following common patterns.

Topics:

• Define a Select Operation on Database Tables
• Define Fault Mapping in Orchestrated Integrations

Define a Select Operation on Database Tables

You can define a `SELECT` operation to perform against database tables. This section provides a high level overview of creating an integration in which an Oracle Database Cloud Service Adapter is configured as an invoke connection to retrieve table records from the Oracle Database.

To define a `SELECT` operation on database tables:

1. Configure SOAP Adapter and Oracle Database Cloud Service Adapter connections.
2. Select **App Driven Orchestration** in the Create Integration - Select a Style dialog.
3. Add and configure the SOAP Adapter as a trigger connection in the integration. The SOAP Adapter is configured to accept an input and return the response received from the invoke connection.
4. Add the Oracle Database Cloud Service Adapter as an invoke connection in the integration. This invokes the Adapter Endpoint Configuration Wizard.
5. On the Basic Info page, select **Perform an Operation On a Table** as the type of operation to perform and **Select** as the operation to perform on the table.
6. On the Operate On Table page, specify the schema and tables to import, and click **Import Tables**. For this example, the following values are specified:
   • **Schema**: HR
   • **Table Type**: TABLE
   • **Table Name**: %TAB
   • **Selected Tables**: EMPLOYEE_TAB and DEPARTMENT_TAB. The tables are imported together for the Oracle Database Cloud Service Adapter to recognize the relation between the tables.
The page is refreshed for you to select the parent (root) database table.

7. Select the parent table (for this example, **DEPARTMENT_TAB** is selected).

   This page enables you to:
   - View the automatically created table relationships and create new ones.
   - View and deselect attributes.
   - View and edit the automatically created SQL query.

8. If you want to edit the automatically created SQL query, click **Edit** to the right of **Review and edit SQL Query**.
   a. Click **Edit using Expression Builder**. You can also manually edit the SQL query by clicking **SQL Edit**.
b. Click **Add New** to add new criteria to the SQL query. The automatically created SQL query is displayed below the link.

```
SELECT DEPTNO, DEPTNAME, LOC FROM DEPTAMENT_TAB
```

c. Specify values for the following fields, and click **OK**.
   - **First Argument**
   - **Operator**
   - **Second Argument**

For example:

The criteria you specify are appended to the existing SQL query as part of a **WHERE** clause. Any additional SQL query criteria you specify are appended as part of an **AND** clause. For example:

```
SELECT DISTINCT t0.DEPTNO, t0.DEPTNAME, t0.LOC FROM DEPTAMENT_TAB t0, EMPLOYEE_TAB t1
WHERE ((t0.DEPTNO = #deptno) AND (t1.EMPID > 999)) AND (t1.DEPTNO = t0.DEPTNO))
```

d. Click **OK**.

9. Click **Next**.

10. View your selections on the Summary page. Links to the tables you selected to import and SQL query you specified are provided.

11. Click **Done** to exit the Adapter Endpoint Configuration Wizard.
12. Complete the integration by performing mapping and tracking tasks.
13. Activate the integration.
14. Copy the link to invoke the integration from under the How to Run link.
15. Invoke the integration from a tool such as the SOAP UI.
16. Review the values returned by the Oracle Database Cloud Service Adapter.

Define Fault Mapping in Orchestrated Integrations

You can define fault mappings in integrations. This mapping transforms a Database Adapter fault when used as a target into the source format defined in its WSDL. You add the Database Adapter to a scope action in an orchestrated integration and select this fault in the Fault Handler part of the scope action.

A serviceInvocationError fault mapping is defined in the WSDL.

In the mapper, the elements of serviceInvocationError provide details about the runtime fault:

- **type**: The type of fault.
- **title**: The title of the fault.
- **detail**: Information about the fault cause.
- **errorCode**: Information about the fault code.
- **remedialAction**: How to fix the fault.

This fault structure is populated during runtime when any exception occurs in an outbound invocation (for example, a primary key violation).

If using the adapter in a map data integration, only reason, detail, and errorCode are available in the mapper.

Assume an exception (for example, NumberFormatException) occurs in an invoke (outbound) adapter. Exceptions are mapped in fault mappings and returned to the source format as defined in its WSDL contract. In this use case, a stored procedure is used that accepts only an integer type. If you invoke the adapter by passing a noninteger value, Oracle Integration reports the fault back to you.

To define fault mapping:

1. Create connections for the SOAP Adapter and the Database adapter.
2. Create an orchestrated integration.
3. Drag the SOAP Adapter into the integration canvas as a trigger.
   The Adapter Endpoint Configuration wizard is displayed.
4. Configure the SOAP Adapter (for this example, named s1).
5. From the Actions palette, drag a Scope action below the SOAP Adapter.
6. From the Invokes palette, drag the Database Adapter inside the scope.
The Adapter Endpoint Configuration wizard is displayed.

7. Select an operation to invoke any stored procedure that accepts only an integer as the input parameter (for this example, the adapter is named \texttt{db1}).


9. In the integration canvas, click \textbf{Reposition} and move the \texttt{s1} map inside the scope.

10. Define mappings for \texttt{s1}.
11. Click the **Fault Handler** part and select **Oracle Database** : `servicelnvocationError db1`.

12. From the **Actions** palette, drag a **Fault Return** action inside the **Fault Handler** part.

The root element for the fault is `serviceInvocationError`. The fault includes other elements that carry the fault details: `type`, `title`, `detail`, `errorCode`, and `remedialAction`. The `detail` element carries information about the fault cause. The `remedialAction` element suggests the action to fix the fault.

**14.** From the menu, select **Tracking** and define the tracking field.

**15.** Activate and invoke the integration by passing a string value (that is, a noninteger value) from the SOAP UI.

```xml
<typ:getOrganization>
  <typ:partyId>test</typ:partyId>
</typ:getOrganization>
```

The fault response returns information similar to the following:

```xml
<nstrgmpr:code>XSD object conversion error</nstrgmpr:code>
<nstrgmpr:message>An error occurred while parsing XML representing a Java object.</nstrgmpr:message>
<nstrgmpr:severity>Unable to convert the XSD element DATA_IN whose SQL type is INTEGER and JDBC type is INTEGER. Cause: java.lang.NumberFormatException: For input string: "test"
<nstrgmpr:severity>
<nstrgmpr:detail>
<nstrgmpr:code>serviceInvocationError</nstrgmpr:code>
<nstrgmpr:message>Check to ensure that the XML data describing the object matches the definition of the element in the XSD.</nstrgmpr:message>
<nstrgmpr:detail/>
<nstrgmpr:code/>
<nstrgmpr:severity/>
<nstrgmpr:detail/>
<nstrgmpr:code/>
</nstrgmpr:detail>
```

Chapter 4
Define Fault Mapping in Orchestrated Integrations
4-7
Troubleshoot the Oracle Database Cloud Service Adapter

Review the following topics to learn about troubleshooting issues with the Oracle Database Cloud Service Adapter.

Topics:
• Set Null to Collections
• Wrappers Require Regeneration After Objects Change
• Recover from a CLOUD-0005: Unable to Establish Connection Error
• Special Characters are Not Supported in Schema Names
• Resolve Message Time Out Errors

Additional integration troubleshooting information is provided. See Troubleshoot Oracle Integration in Using Integrations in Oracle Integration.

Set Null to Collections

You may sometimes want to pass null to the adapter while mapping collections. If you do not map those collections, an ORA-06550 pl/sql statement ignored error can occur. To avoid this error, map the collections using the mapping component attribute name='xsi:nil'. This action ensures that a null collection is propagated to the adapter.

Wrappers Require Regeneration After Objects Change

The adapter automatically generates the wrapper packages and objects for stored procedures used in an integration when PL/SQL boolean, table, and record types are involved. If the underlying objects (that is, the IN/OUT parameters) are changed, the wrappers must be regenerated after you delete the existing wrapper's packages and objects. During design time or activation, the wrappers are regenerated automatically with the latest object definitions available in the database.

Recover from a CLOUD-0005: Unable to Establish Connection Error

If you receive the following error:

CLOUD-0005: Unable to establish connection.
Please check connection parameters · IO Error: Invalid connection string
Perform the following steps:

1. Check if the service name can be modified to remove the hyphen (-).
2. If you cannot remove the hyphen, prefix the host name in the database connection with // (for example, //host.test.com).

Special Characters are Not Supported in Schema Names

If you use schema names with special characters such as #, integration activation fails. For stored procedures, the schema derives the names of the types in the XSD. If the type name contains #, the XSD has problems with the name. Use a schema name that does not contain any special characters.

Resolve Message Time Out Errors

The following error can occur during both design time (in both the inbound and outbound directions) and runtime.

Message not received within X seconds of wait interval

**Note:**

When using the adapter to connect to an Oracle E-Business Suite database instance and this error continuously occurs, review the SQL query plans and other SQL tuning aspects. The adapter relies on JDBC driver APIs to fetch metadata such as table details, stored procedure details, and so on. This involves execution of certain SQL queries by the JDBC driver involving SYS tables such as the ALL_TYPES table. Since Oracle E-Business Suite has a large data dictionary, these metadata queries requires tuning consideration to improve overall performance of the adapter.