Oracle® Cloud

Using the File Adapter with Oracle Integration 3

F45553-05
October 2024

Oracle Cloud Using the File Adapter with Oracle Integration 3,

F45553-05

Primary Author: Oracle Corporation

Contents

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

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 File Adapter

Review the following conceptual topics to learn about the File 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:

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

File Adapter Capabilities

Use the File Adapter in integrations in Oracle Integration to exchange (write) files on directories that are local to the connectivity agent.

> **Note:**
>
> Starting with release 24.10, it is recommended that you enable the **Use a Single Agent at a Time** option when creating the agent group that you associate with the File Adapter during connection configuration. This option lets you install and use two connectivity agents, which function as follows:
>
> - The primary agent is active and handles all incoming requests and polling responsibilities.
> - The second agent is passive and runs in standby mode, ready to take over if the primary agent fails.
>
> See Create an Agent Group in *Using Integrations in Oracle Integration 3*.

The File Adapter provides the following benefits:

- Transfers (writes) files to any publicly accessible server in either binary or ASCII format.
- Supports a synchronous one-way request message exchange pattern. There is no response from the server.
- Allows the creation of binary files (opaque) and schema-based files such as XML schema and comma-separated value.

> **Note:**
>
> The File Adapter supports Microsoft Excel files. Either send the file as an opaque (pass through) or first save it as a comma-separated value file before using Oracle Integration.

- Supports processing of message sizes described in the service limits documentation. See Service Limits in *Provisioning and Administering Oracle Integration 3*.

  Note that the size of CSV files increases when translated into a message. Therefore, the message size must be less than the size described in the service limits documentation so that after translation, the message size does not exceed the limit.

- Supports the upload of an XSD file without a target namespace. In these cases, a surrogate namespace is added to the XSD file that all messages then use:

  ```
  http://xmlns.oracle.com/cloud/adapter/nxsd/surrogate
  ```

- Supports complex XSDs that can import and include other XSDs. The included XSDs in the ZIP file can import the XSD from an HTTP location. All XSD files must be added to a ZIP file and uploaded when configuring the File Adapter for read and write operations in the Adapter Endpoint Configuration Wizard.

  In the following example, the hierarchy of the ZIP file to upload is as follows:

  ```
  zipxsd.zip
    first.xsd
    second (folder)
      second.xsd
  ```

  `first.xsd` imports `second.xsd`.

  ```
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://xmlns.oracle.com/first"
  targetNamespace="http://xmlns.oracle.com/first"
        xmlns:tns1="http://xmlns.oracle.com/second">
  <xs:import schemaLocation="./second/second.xsd"
  targetNamespace="http://xmlns.oracle.com/second"/>
  <xs:import schemaLocation="https://example.com/fscmService/ItemServiceV2?
  XSD=/xml/datagraph.xsd" targetNamespace="commonj.sdo"/>
  <xs:element name="book">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="isbn" type="xs:string"/>
        <xs:element name="title" type="xs:string"/>
        <xs:element name="author" type="tns1:author"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  </xs:schema>
  ```

  The contents of `second.xsd` are as follows.

  ```
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://xmlns.oracle.com/second"
  ```

```
targetNamespace="http://xmlns.oracle.com/second">
<xs:import schemaLocation="https://example.com/fscmService/ItemServiceV2?
XSD=/mycompany/apps/scm/productModel/items/itemServiceV2/
ItemAttachment.xsd"
targetNamespace="http://xmlns.oracle.com/apps/scm/productModel/items/
itemServiceV2/"/>
<xs:complexType name="author">
    <xs:sequence>
       <xs:element name="name" type="xs:string"/>
       <xs:element name="address" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="Admin">
    <xs:complexType>
          <xs:sequence>
             <xs:element name="AdminName" type="xs:string"/>
             <xs:element name="AdminAdd" type="xs:string"/>
          </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>
```

> **Note:**
>
> If you are importing from HTTPS locations, ensure that you import the SSL certificates into Oracle Integration.

- Supports specifying sample XML documents for the payload when configuring the File Adapter for read and write operations.
- Supports specifying JSON documents for the payload when configuring the File Adapter for read and write operations.

The File Adapter is a predefined adapter included with Oracle Integration . You can configure File Adapter as a connection in an integration in Oracle Integration.

File Adapter Restrictions

Note the following File Adapter restrictions.

- Oracle Integration does not support reading and writing the BOM mark in a payload.
- The File Adapter only supports homogeneous arrays in JSON. Heterogeneous arrays in JSON samples and payloads are not supported. An example of a heterogeneous array is as follows:

```
{
    "example": [
         "kumar",
        {
            "firstName": ["John"],
        },
        {
            "length": 100,
            "width": 60,
```

```
                "height": 30
            }
    ]
    }
```

- Uploading a schema without a target namespace in the File Adapter is supported only for an XML schema and not supported for comma separated values or a native schema.

Workflow to Create and Add a File Adapter Connection to an Integration

Follow a 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 a File Adapter Connection
2	Create the integration. When you do this, you add trigger and invoke connections to the integration.	Create Integrations and Add the File 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*

2

Create a File 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

These are the prerequisites for creating a connection with the File Adapter.

> **Note:**
>
> To create a connection, a trusted public certificate may be required. Typically, the certificate is included with Oracle Integration. If you cannot locate the public certificate, contact your administrator. Rename the public certificate file extension to `.crt`. To upload the certificate, see Upload a Certificate to Connect with External Services.

- Make sure that the target server is publicly accessible and not behind a firewall. If the target server is behind a firewall, make sure it has been opened for outside client access.
- Make sure that you have write permissions on the server directory to which the files are transferring.

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 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 ...
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.

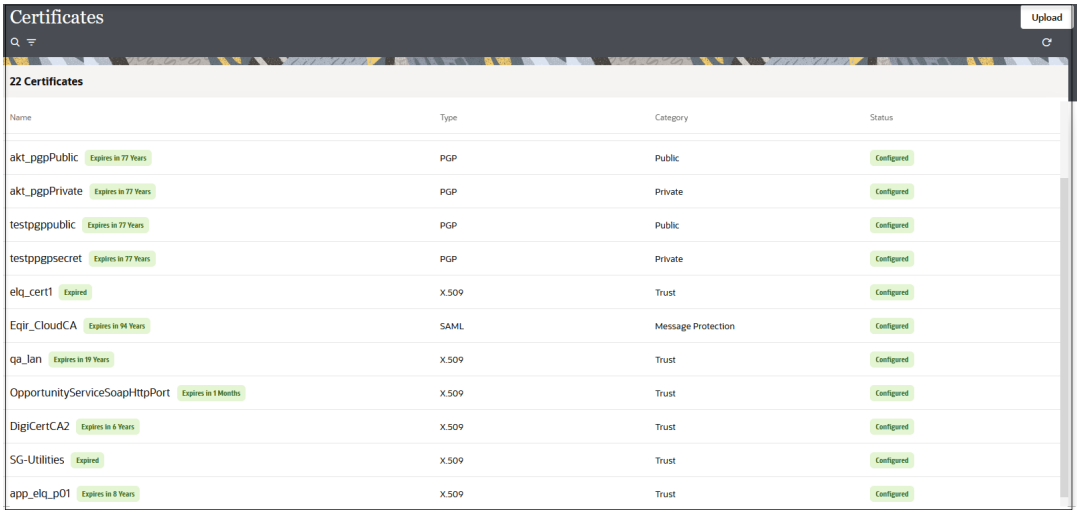
2. Wait for a message about the results of the connection test.
   - If the test was successful, then the connection is configured properly.
   - If the test failed, then edit the configuration details you entered. Check for typos and verify URLs and credentials. Continue to test until the connection 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 File Adapter Connection to an Integration

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

These topics describe the wizard pages that guide you through configuration of the File Adapter as a trigger and invoke in an integration.

Topics:

Trigger Basic Information Page

You can enter a name and description on the Basic Info page of each trigger and invoke 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 dashes in the name. You cannot include the following: • Blank spaces (for example, `My Inbound Connection`) • Special characters (for example, `#;83&` or `righ(t)now4`) • Multibyte characters
What does this endpoint do?	Enter an optional description of the connection's responsibilities. For example: `This connection receives an inbound request to synchronize account information with the cloud application.`

Element	Description
Do you want to specify the structure for the contents of the file?	Select **Yes** to define a schema format for the transfer files. Select **No** if a schema is not required and you want to send opaque files (for example, a GIF or PNG file).
Which of the following choices would be used to describe the structure of the file contents?	Create a new schema with a CSV file or select an existing schema from your local system. • **Sample delimited document (e.g. CSV)** • **XML schema (XSD) document** • **Sample XML document (Single on No Namespace)** • **Sample JSON document**

Trigger Configure File Read Page

Enter the File Adapter trigger file read configuration values for your integration.

Element	Description
Specify an Input Directory	Specifies the directory path for file reading.
Specify a File Name Pattern	Specifies the input file naming pattern.
Specify the Rejection Directory	Specify the directory in which to place rejected files.
Read Files Recursively	Select to read subdirectory files recursively.
Maximum Files	Specifies the number of files to be processed in a single poll operation.
Polling Frequency	Specifies the polling operation frequency.
Processing Delay	Specifies the polling operation frequency delay.
Delete Files After Successful Reading	When selected, files are deleted after they are successfully read.
Ignore File Not Found Error in Delete	When selected, any `file not found` errors are ignored during file polling, and polling continues.

Trigger or Invoke File Contents - Definition Page

Enter the format definition parameters.

The fields that display on the Format Contents - Definition page are determined by your selection on the Schema page:

- Creating a New Schema from a CSV File
- Select an existing XML schema or schema archive from the file system
- Provide a sample XML document from the file system
- Provide a sample JSON document from the file system

> **Note:**
>
> - If configuring the adapter in the trigger (inbound) direction, schema selection is *not* supported. If you select **Yes** to define a schema for the endpoint on the Basic Info page, nothing prevents you from uploading a schema on the Format Contents - Definition page. However, this schema is not used. You must select **No** on the Basic Info page to transfer files as an attachment.
> - The adapter can only be configured as an invoke connection in an orchestrated integration.

Creating a New Schema from a CSV File

Element	Description
Select the Delimited Data File	Select the delimited comma-separated value (CSV) file from which to create the schema file. The content of the file is then displayed at the bottom of the page.
Enter the Record Name	Enter the record name. This becomes the parent element in the created schema file for the record names selected as column headers from the CSV file.
Enter the Recordset Name	Enter the recordset name. This becomes the root element of the created schema file.
Select the Field Delimiter	Select one of the following supported file delimiter options: • Single space • Comma • Semicolon • Tab • Pipe (for example, Name|City|Country)
Character Set	Select a character set. The selected value will be used as the encoding format while reading the sample data file. This field is used for character encoding during file transfer. If data sent to the adapter is in a specific encoding format, then select that same encoding format in the adapter. Otherwise, there may be some character loss in the final written file. This also corrupts the file. For example, if a REST Adapter is using UTF-8 encoding and the adapter is configured with the ASCII character set, this may corrupt the file. If you select the same UTF-8 encoding in the adapter, the problem is resolved.

Element	Description
Optionally Enclosed By	This value causes occurrences of the selected delimiter to be ignored during processing. For example, when processing the following record: `Fred,"2 Old Street, Old Town,Manchester",20-08-1954,0161-499-1718` If the selected **Field Delimiter** is "," and the **Optionally Enclosed By** value is `quot;` ("), then the value `2 Old Street, Old Town,Manchester` is treated as a single record column. If **Optionally Enclosed By** is used, the character must not be part of any field. Therefore, the following is invalid: • `a,b",c` (In this scenario, " is part of the second field) • `a,"b,c",d` (In this scenario, the second field is created using **Optionally Enclosed By** (`'"'`). However, it has `','` as content for the second field value.)
Terminated By	Displays by default the first row of the selected CSV file as the column headers. Select the option to terminate the end of a line. • **${eol}** • **\n** • **\r\n** • **\r**
Detach	Select to edit the CSV file in a separate window.
Use First Row as Column Headers	Select to use the first row as the column headers.
Mark All As Optional	Select to mark elements as optional in the schema file. By default, all elements are mandatory. You can also select the data type (for example, string, byte, integer, and so on) to use for each column in the table and mark specific elements as optional. While this option enables you to select all elements as optional, you must have at least one mandatory element to validate this page. This checkbox provides a convenient method to select the majority of elements as optional.

Select an existing XML schema or schema archive from the file system

The schema archive can have a single top level schema with nested input and include elements containing absolute or relative paths.

Element	Description
Select File	Select the existing schema file to use.
Selected File Name	Displays the selected schema file name.
Select the Schema Element	Select the schema element. This field is displayed after the schema file is selected. The element name is treated as the root element in the uploaded schema file.

Provide a sample XML document from the file system

The XML document should contain no namespaces or only a single namespace.

Element	Description
Select File	Select the existing XML document to use.
Selected File Name	Displays the selected schema file name.
Select the Schema Element	Select the schema element. This field is displayed after the schema file is selected. The element name is treated as the root element in the uploaded schema file.

Provide a sample JSON document from the file system

Element	Description
Select File	Select the existing JSON file to use.
Selected File Name	Displays the selected file name.
Select the Schema Element	Select the schema element. This field is displayed after the schema file is selected. The element name is treated as the root element in the uploaded schema file.

Invoke Basic Information Page

Enter the basic information parameters.

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

Invoke Operations Page

Enter the File Adapter operation properties.

Choose one of the following operations to perform on files. Depending on which operation you choose, you are presented with different options and parameters that apply specifically to that operation. All operations support dynamic file name and directory through mapping.

- Read File: Reads a file from the file location. This operation reads a single file. If you have not selected any schema, a file size of up to 1 GB is supported. If you have selected the schema, a file size of up to 50 MB is supported.
- Write File: Writes a single file onto the target file server.
- List Files: Lists the names of the files in the input directory. This operation returns the list of files without any data. It is similar to running the `ls` command in Linux.
- Move a File: Moves a file from one location to another on the same target file server.
- Delete a File: Deletes a single file on the file server.
- Download File: Specifies a directory to be used for staged activity. Use this directory to stage large files for processing. This operation downloads a single file in the logical directory you specify. File size support is up to 1 GB.

> **Note:**
>
> If you want to read/download multiple files, use the following integration pattern. File List File (Operation) > For Each Loop > File Read File (Operation) > End For Each Loop

The following tables describe the key information on the File Adapter invoke Operations page. Depending on which operation you choose, you are presented with different options and parameters that apply specifically to that operation. Each table corresponds to a specific operation.

Each operation can be performed against ZIP and GZIP file formats.

> **Note:**
>
> Elements marked with “+” can be mapped in the mapper and can be passed dynamically at runtime.

Read File

When using the connectivity agent, the **Read File** operation when used without a schema supports files of up to 1 GB in size. This limit is the same as the **Download File** operation. If you are using a schema for transformation, the file size limit is 50 MB. You can specify sample XML and JSON documents for the payload when configuring the File Adapter for read operations.

Element	Description
Input Directory	The name of the directory that contains the file to be read. For example, `/tmp/Oracle/input`.
File Name	The name of the file to be read. The name can also be provided by the mapper. File sizes of up to 50 MB are supported.

Write File

When using the connectivity agent, the **Write File** operation when used without a schema supports files of up to 1 GB in size. This limit is the same as the **Download File** operation. If

you are using a schema for transformation, the file size limit is 50 MB. You can specify sample XML and JSON documents for the payload when configuring the File Adapter for write operations.

Element	Description
Specify an Output Directory	The directory path to which to write the file. For example, `/tmp/Oracle/output`.
File Name Pattern	The pattern of file names to transfer to the output directory. Use the pattern inside `%%`. For example, `Oracle%SEQ%ICS.txt` creates files in sequence, such as `Oracle1ICS.txt`, `Oracle2ICS.txt`, and so on. For a list of supported file patterns, click the information icon. **Note:** Use of these patterns leads to message loss if the messages are written to the file system at the same time or files are written to a separate node. To ensure that files are not overwritten, use the mapper to assign the file names explicitly. You can use XPath functions to ensure that file names are unique.
Append to Existing File	If selected, the file content is appended to the existing file content and is not overwritten. **Notes**: • Writing record by record using the **Append to Existing File** option creates too many network calls and eventually slows down the process. Instead, use a stage file action and write the records to a stage directory. Once all records are written, use the **List File** option in the stage file action and the File Adapter option to transfer the file to a file location. This approach reduces processing time and prevents too many calls to the file server. • You may be performing successive appends by putting multiple write operations or using a write operation in a for-each loop. In this scenario, if the file server is running on multiple nodes or on a cloud service, it should synchronize the data immediately among all nodes. Otherwise, if the next request goes to a different node, there is inconsistency in the final output file. As an alternative, you can also place a wait activity before or after every append to give the data time to synchronize.

List Files

Element	Description
Input Directory	The name of the directory that contains the file to be read. For example, `/tmp/Oracle/input`.
File Name Pattern	Specify a wildcard pattern to be used for listing files from the input directory. For example, `order*.csv`.

Element	Description
Max Files	The maximum number of file names that should be listed. The maximum value is 1000. **Note**: The **List File** operation returns the file list in a sorted order according to the last modified time. If you selected **10** as the maximum number of files and the last modified time of the eleventh file is the same as the tenth file, then the eleventh file is also added. This continues until you get a file with a different timestamp. For example, assume the directory has 15 files and you select **10** as the maximum number of files. If the tenth, eleventh, twelfth, and thirteenth files have the same time stamp, then the list file returns thirteen files.
Minimum Age	The minimum age in seconds for files to be displayed. For example, if the last time a file was modified is 02:28:45 AM and the minimum age is defined as 80 seconds, the file will not be listed at 02:29:00 AM and 02:30:00 AM. It will be listed after 02:30:05.
Process Files Recursively	List files within directories recursively beneath the input directory.

Move a File

Element	Description
Directory Path	The name of the directory that contains the file to be moved. For example, `/tmp/Oracle/input`.
File Name	The name of the file to be moved. There is no limit to the size of the file to move.
Target Directory Path	The name of the directory to which the file will be moved.
Target File Name	The name of the file as it will be written in the target directory.
Overwrite File	Select to overwrite the file in the target directory.

Delete a File

Element	Description
Directory Path	The directory path to the file you wish to delete.
File Name	The name of the file to delete.

Download File

The **Download File** operation enables you to map the download directory dynamically. This works fine for new integrations. You can also edit an existing **Download File** operation in the Adapter Endpoint Configuration Wizard to add more file download operations, then click **Done** to regenerate the adapter artifacts. You can download file sizes of up to 1 GB.

Using the **Download File** operation with the connectivity agent enables you to upload the file from the connectivity agent server to Oracle Integration. This operation enables you to

download and upload files of up to 1 GB in size. This operation may take more time depending on the network conditions between the file server and the connectivity agent and the connectivity agent and Oracle Integration.

Element	Description
Input Directory	Specify the name of the directory that contains the file to be read. For example, `/tmp/Oracle/input`.
File Name	The name of the file to download. This can be overridden using the mapper.
Download Directory	The name of the directory to be used for stage file action. See Processing Files in Scheduled Integrations with a Stage File Action. Do not enter the directory path in double quotes. The quotes are mistakenly included as part of the directory name.

Invoke Schema Page

Enter the schema properties.

Element	Description
Do you want to specify the structure for the contents of the file?	• **Yes**: Select to define a schema format to use for the file to transfer. This option enables you to read and write files. Understand the file size supported. See Service Limits in *Provisioning and Administering Oracle Integration 3*. • **No**: Select if a schema is not required and you want to send the file as an attachment. You typically select **No** if you want to stream large files to or from the servers. This option enables you to read and write files up to 1 GB in size.
Which one of the following choices would be used to describe the structure of the file contents?	Select an option: • **Sample delimited document (e.g. CSV)**: Select to create a new schema file from a comma-separated value (CSV) file. On a subsequent page of this wizard, you are prompted to select the CSV file from which to create the schema. • **XML schema (XSD) document**: Select an existing schema file. On a subsequent page of this wizard, you are prompted to select the existing schema (XSD) file from the file system. You can also upload non-native schemas that are created outside of Oracle Integration. See XSD File Capabilities. • **Sample XML document (Single or No Namespace)**: Select to provide sample XML content for the payload. • **Sample JSON document**: Select to provide sample JSON content for the payload.

XSD File Capabilities

- The adapter supports the upload of an XSD file without a target namespace. In these cases, a surrogate namespace is added to the XSD file that all messages then use:

  ```
  http://xmlns.oracle.com/cloud/adapter/nxsd/surrogate
  ```

- The adapter supports complex XSDs that can import and include other XSDs. The included XSDs in the ZIP file can import the XSD from an HTTP location. All XSD files must be added to a ZIP file and uploaded when configuring the adapter for read and write operations in the Adapter Endpoint Configuration Wizard.

  In the following example, the hierarchy of the ZIP file to upload is as follows:

  ```
  zipxsd.zip
    first.xsd
    second (folder)
      second.xsd
  ```

  `first.xsd` imports `second.xsd`.

  ```
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://xmlns.oracle.com/first"
  targetNamespace="http://xmlns.oracle.com/first"
        xmlns:tns1="http://xmlns.oracle.com/second">
  <xs:import schemaLocation="./second/second.xsd"
  targetNamespace="http://xmlns.oracle.com/second"/>
  <xs:import schemaLocation="https://example.com/fscmService/ItemServiceV2?
  XSD=/xml/datagraph.xsd" targetNamespace="commonj.sdo"/>
  <xs:element name="book">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="isbn" type="xs:string"/>
        <xs:element name="title" type="xs:string"/>
        <xs:element name="author" type="tns1:author"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  </xs:schema>
  ```

  The contents of `second.xsd` are as follows.

  ```
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://xmlns.oracle.com/second"
  targetNamespace="http://xmlns.oracle.com/second">
  <xs:import schemaLocation="https://example.com/fscmService/ItemServiceV2?
  XSD=/mycompany/apps/scm/productModel/items/itemServiceV2/
  ItemAttachment.xsd"
  targetNamespace="http://xmlns.oracle.com/apps/scm/productModel/items/
  itemServiceV2/"/>
  <xs:complexType name="author">
      <xs:sequence>
         <xs:element name="name" type="xs:string"/>
         <xs:element name="address" type="xs:string"/>
  ```

```
    </xs:sequence>
</xs:complexType>
<xs:element name="Admin">
    <xs:complexType>
          <xs:sequence>
             <xs:element name="AdminName" type="xs:string"/>
             <xs:element name="AdminAdd" type="xs:string"/>
          </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>
```

> **Note:**
>
> If you are importing from HTTPS locations, ensure that you import the SSL certificates into Oracle Integration.

Trigger or Invoke File Contents - Definition Page

Enter the format definition parameters.

The fields that display on the Format Contents - Definition page are determined by your selection on the Schema page:

- Creating a New Schema from a CSV File
- Select an existing XML schema or schema archive from the file system
- Provide a sample XML document from the file system
- Provide a sample JSON document from the file system

> **Note:**
>
> - If configuring the adapter in the trigger (inbound) direction, schema selection is *not* supported. If you select **Yes** to define a schema for the endpoint on the Basic Info page, nothing prevents you from uploading a schema on the Format Contents - Definition page. However, this schema is not used. You must select **No** on the Basic Info page to transfer files as an attachment.
> - The adapter can only be configured as an invoke connection in an orchestrated integration.

Creating a New Schema from a CSV File

Element	Description
Select the Delimited Data File	Select the delimited comma-separated value (CSV) file from which to create the schema file. The content of the file is then displayed at the bottom of the page.
Enter the Record Name	Enter the record name. This becomes the parent element in the created schema file for the record names selected as column headers from the CSV file.

Element	Description
Enter the Recordset Name	Enter the recordset name. This becomes the root element of the created schema file.
Select the Field Delimiter	Select one of the following supported file delimiter options: • Single space • Comma • Semicolon • Tab • Pipe (for example, Name|City|Country)
Character Set	Select a character set. The selected value will be used as the encoding format while reading the sample data file. This field is used for character encoding during file transfer. If data sent to the adapter is in a specific encoding format, then select that same encoding format in the adapter. Otherwise, there may be some character loss in the final written file. This also corrupts the file. For example, if a REST Adapter is using UTF-8 encoding and the adapter is configured with the ASCII character set, this may corrupt the file. If you select the same UTF-8 encoding in the adapter, the problem is resolved.
Optionally Enclosed By	This value causes occurrences of the selected delimiter to be ignored during processing. For example, when processing the following record: `Fred,"2 Old Street, Old Town,Manchester",20-08-1954,0161-499-1718` If the selected **Field Delimiter** is "," and the **Optionally Enclosed By** value is `quot;` ("), then the value `2 Old Street, Old Town,Manchester` is treated as a single record column. If **Optionally Enclosed By** is used, the character must not be part of any field. Therefore, the following is invalid: • `a,b",c` (In this scenario, " is part of the second field) • `a,"b,c",d` (In this scenario, the second field is created using **Optionally Enclosed By** (`'"'`). However, it has `','` as content for the second field value.)
Terminated By	Displays by default the first row of the selected CSV file as the column headers. Select the option to terminate the end of a line. • **${eol}** • **\n** • **\r\n** • **\r**
Detach	Select to edit the CSV file in a separate window.
Use First Row as Column Headers	Select to use the first row as the column headers.

Element	Description
Mark All As Optional	Select to mark elements as optional in the schema file. By default, all elements are mandatory. You can also select the data type (for example, string, byte, integer, and so on) to use for each column in the table and mark specific elements as optional. While this option enables you to select all elements as optional, you must have at least one mandatory element to validate this page. This checkbox provides a convenient method to select the majority of elements as optional.

Select an existing XML schema or schema archive from the file system

The schema archive can have a single top level schema with nested input and include elements containing absolute or relative paths.

Element	Description
Select File	Select the existing schema file to use.
Selected File Name	Displays the selected schema file name.
Select the Schema Element	Select the schema element. This field is displayed after the schema file is selected. The element name is treated as the root element in the uploaded schema file.

Provide a sample XML document from the file system

The XML document should contain no namespaces or only a single namespace.

Element	Description
Select File	Select the existing XML document to use.
Selected File Name	Displays the selected schema file name.
Select the Schema Element	Select the schema element. This field is displayed after the schema file is selected. The element name is treated as the root element in the uploaded schema file.

Provide a sample JSON document from the file system

Element	Description
Select File	Select the existing JSON file to use.
Selected File Name	Displays the selected file name.
Select the Schema Element	Select the schema element. This field is displayed after the schema file is selected. The element name is treated as the root element in the uploaded schema file.

Summary Page

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**.