Oracle® Cloud

Sync Data between Azure Storage and PostgreSQL Upon Receiving Notification on Azure Event Grid





Oracle Cloud Sync Data between Azure Storage and PostgreSQL Upon Receiving Notification on Azure Event Grid, G24762-02

Copyright © 2025, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

Contents

1	About This Recipe		
	Overview	1-:	
	System and Access Requirements	1-3	
2	Before You Install the Recipe		
	Access Your FTP Server	2-2	
	Configure Azure Event Grid	2-3	
	Configure Azure Storage (Get Blob)	2-3	
	Configure PostgreSQL	2-3	
3	Install and Configure the Recipe		
	Configure the Azure Event Grid Connection	3-2	
	Configure the Azure Storage Invoke Connection	3-2	
	Configure the PostgreSQL Connection	3-2	
	Configure the FTP Connection	3-2	
4	Activate and Run the Recipe		



Preface

This document describes how to install, configure, and run this recipe in Oracle Integration 3.

Topics:

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

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.

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

For more information, see these Oracle resources:

- Oracle Integration documentation on the Oracle Help Center.
- Oracle Cloud at http://cloud.oracle.com.

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.



Convention	Meaning
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

About This Recipe

Use this recipe to synchronize data between Azure Storage and PostgreSQL upon receiving notification on Azure Event Grid.



Oracle provides this recipe as a sample only. The recipe is meant only for guidance, and is not warranted to be error-free. No support is provided for this recipe.

Overview

This recipe demonstarates how a file that is uploaded to Azure Storage gets reflected in PostgreSQL database when a notification is received on Azure Event Grid.

In this recipe, the Event grid schema is used from Azure Event Grid. GET Blob (file) is used to read the uploaded file by Azure Storage (Invoke) and Put Blob (file) is used to upload the file into Azure Storage (Invoke). The Data Import Operation is reflected in the PostgreSQL database (Invoke).

To use the recipe, you must install the recipe and configure the connection and other resources within the recipe. The recipe uses three adapters in two integration flows. The first integration flow (**Oracle FTP Azure Storage Create Data**) is a scheduled orchestration style integration which reads a file from FTP to upload the data in Azure Storage. The second integration flow (**Oracle AEG Azure Storage PostgreSQL Data Sync**) is an app-driven integration that syncs data from Azure Storage to PostgreSQL on receiving a notification on Azure Event Grid.

System and Access Requirements

- Oracle Integration 3
- Azure Event Grid for creating subscriptions to receive notifications, with the following roles:
 - Azure Event Grid Reader
 - Azure Event Grid Contributor
 - Azure Event Grid Sender
- Azure Storage for uploading the files, with the following roles:
 - Data Reader
 - Data Contributor
- A PostgreSQL database
- An account on PostgreSQL database with the Administrator role, for importing the data.
- A secure FTP (sFTP) server or File Server.
- An FTP client to read the data/file from the server.

Before You Install the Recipe

You must perform the following configuration tasks on your FTP, Azure Event Grid, Azure Storage and PostgreSQL instances to successfully connect to these external systems using Oracle Integration and synchronize data between Azure Storage and PostgreSQL.

Access Your FTP Server

Obtain an FTP server and ensure that you complete the **FTP Server Prerequisites** listed in Prerequisites for Creating a Connection.

Also, ensure that you have the read and write permissions on the FTP server directory to and from which you want to transfer files.

Configure Azure Event Grid

To access Azure Event Grid from Oracle Integration and successfully receive notifications on created subscriptions you must complete certain prerequisites.

See Prerequisites for Creating a Connection.

Configure Azure Storage (Get Blob)

To successfully get a file to Azure Storage using Oracle Integration, Azure Storage needs to get the blob for respective container.

See Prerequisites for Creating a Connection.

Configure PostgreSQL

To access the PostgreSQL database from Oracle Integration and synchronize data, you must perform certain configurations on your PostgreSQL instance.

For general configuration tasks, see Prerequisites for Creating a Connection.

In addition, you must create a database table. To create a database table:

- 1. Log in to your PostgreSQL Workbench instance as an **Administrator**.
- Create the database table in either of the two ways given below.
 - a. Right-click on Public Schema.
 - b. Click Query Tool.
 - Paste the query on the Query tool and then execute.
 For a sample query, refer to the following: PostgreSQLTableCreateQuery.txt

or,

- a. Under public schema, right-click **Tables**, select **Create**, and then **Tables**.
- b. In the Create-Table dialog, specify the name of the table as aeg.

- c. Right-click on the empty table, and under **Create** select **Column**.
- d. Click Save.



Install and Configure the Recipe

On your Oracle Integration instance, install the recipe to deploy and configure the integration and associated resources.

- 1. On the Oracle Integration Home page, in the Get started section, click Browse store.
- 2. Find the recipe you want to install, then click Get.

A message confirms that the recipe was successfully installed, and the recipe card shows **In use**.

3. Click **Configure** on the recipe to configure its resources.

The project workspace opens, displaying all the resources of the recipe. Configure the following resources before you activate and run the recipe.

Configure the Azure Event Grid Connection

- 1. In the Connections section, click the connection name.
- 2. In the Properties section, enter the following details:

Field	Information to Enter
Subscription ID	Enter a valid Subscription ID. See Configure Azure Event Grid.
Resource group	Enter a valid Resource Group.
API version	Enter a valid API version.

3. In the Security section, enter the following details:

Field	Information to Enter
Client Id	Enter a valid Client ID. See Configure Azure Event Grid.
Client Secret	Enter a valid Client Secret.
Tenant ID	Enter a valid tenant ID.

- 4. Click Save. If prompted, click Save again.
- Click Test to ensure that your connection is successfully configured. In the resulting dialog, click Test again.

A message confirms if your test is successful.

6. To return to the project workspace, click **Go back**

Configure the Azure Storage Invoke Connection

- 1. In the Connections section, click the connection name.
- 2. In the Properties section, enter the following details:

Field	Information to Enter
Storage Account name	Enter a valid storage name. See Configure Azure Storage (Get Blob).
Tenant Id	Enter a valid tenant ID.

3. In the Security section, enter the following details:

Field	Information to Enter
Client Id	Enter a valid Client ID. See Configure Azure Storage (Get Blob).
Client Secret	Enter a valid Client Secret.

- Click Save. If prompted, click Save again.
- Click Test to ensure that your connection is successfully configured. In the resulting dialog, click Test again.

A message confirms if your test is successful.

6. To return to the project workspace, click **Go back** .

Configure the PostgreSQL Connection

- 1. In the Connections section, click the connection name.
- 2. In the Properties section, enter the following details:

Field	Information to Enter
Server Name	Enter a valid server name. See Prerequisites for Creating a Connection.
Port	Enter a valid port number.
DBNAME	Enter a valid database name.

3. In the Security section, enter the following details:

Field	Information to Enter
Security Policy	By default, a security policy is selected.
Username	Enter the username to connect to the database. See Prerequisites for Creating a Connection.
Password	Enter the password to connect to the database.

- 4. Click **Save**. If prompted, click **Save** again.
- Click Test to ensure that your connection is successfully configured. In the resulting dialog, click Test again.

A message confirms if your test is successful.

6. To return to the project workspace, click **Go back** <

Configure the FTP Connection

- 1. In the Connections section, click the connection name.
- 2. In the Properties section, enter the following details:



Field	Information to Enter
Host	Enter a valid host address of your FTP server.
Port	Enter a valid port number.

3. In the Security section, enter the following details:

Field	Information to Enter
Security Policy	By default, a security policy is selected.
Username	Enter the username to connect to your FTP server.
Password	Enter the password to connect to your FTP server.

- 4. Click Save. If prompted, click Save again.
- 5. Click **Test** to ensure that your connection is successfully configured. In the resulting dialog, click **Test** again.

A message confirms if your test is successful.

6. To return to the project workspace, click **Go back** .



4

Activate and Run the Recipe

After you've configured the connections and other resources, you can activate and run the recipe.

- 1. In the project workspace, click **Activate**. In the Activate project panel, with the default project deployment selected, choose an appropriate tracing option, then click **Activate**.
 - A message confirms that the integrations have been activated. Refresh the page to view the updated status of the integrations.
- Update integration property values for the Oracle FTP Azure Storage Create Data and Oracle AEG Azure Storage PostgreSQL Data Sync integration flows.
 - In the Integrations section, click Actions • on the integration flow, then select Update property values.
 - In the Update property values panel, update the integration properties with appropriate values.
 - EmailTo: This integration property holds the email ID to which error notifications are to be sent.
 - retryCount: This integration property holds the number of retries when an invoke fails. The default value is 3.
 - c. Click Submit.

A message confirms that the integration properties have been updated successfully.

- 3. Run the Oracle FTP Azure Storage Create Data integration flow.
 - a. In the Integrations section of the project workspace, click **Actions** • on the integration flow, then select **Run**.
 - b. On the Configure and run page, click **Run**.

You've now successfully submitted the integration for execution. The integration now reads a file from FTP and creates the data in Azure Storage.



You can also schedule this integration to run at a date, time, and frequency of your choosing. See Define the Integration Schedule.

- 4. Run the Oracle AEG Azure Storage PostgreSQL Data Sync integration flow.
 - In the Integrations section of the project workspace, click Actions • on the integration flow, then select Run.
 - b. On the Configure and run page, click Run.

You've now successfully submitted the integration for execution. The integration now synchronizes data from Azure Storage to PostgreSQL on receiving a notification on Azure Event Grid.

Note:

To run the recipe, you must create a subscription in Azure Event Grid and a database table in PostgreSQL.

Note:

You can also schedule this integration to run at a date, time, and frequency of your choosing. See Define the Integration Schedule.

- 5. Monitor the running of the integration flows in Oracle Integration.
 - In the project workspace, click Observe. You'll see the integration flows being triggered and running successfully.
 - b. To manage errors in your project, see Manage Errors in a Project.
- 6. Log in to the Azure Event Grid instance and check for the notification received.
 - a. Click the wired message.
 - b. Click the eye icon to open the message.
 - c. Observe the Blob-created message.

Thus, you have successfully received a notification on uploading a file.

- 7. Log in to the PostgreSQL instance, and check for the data created.
 - a. On the DB visualizer, create a successful connection.
 - b. Enter the query as select * from tablename;.
 - c. Observe if the data has been imported successfully.
 - **d.** Cross-verify the file data and imported data into the database.

Thus, you have successfully imported data from Azure Storage into PostgreSQL database.

Related Documentation

- Using the Azure Event Grid Adapter with Oracle Integration 3
- Using the Azure Storage Adapter with Oracle Integration 3
- Using the PostgreSQL Adapter with Oracle Integration 3
- Using the FTP Adapter with Oracle Integration 3

