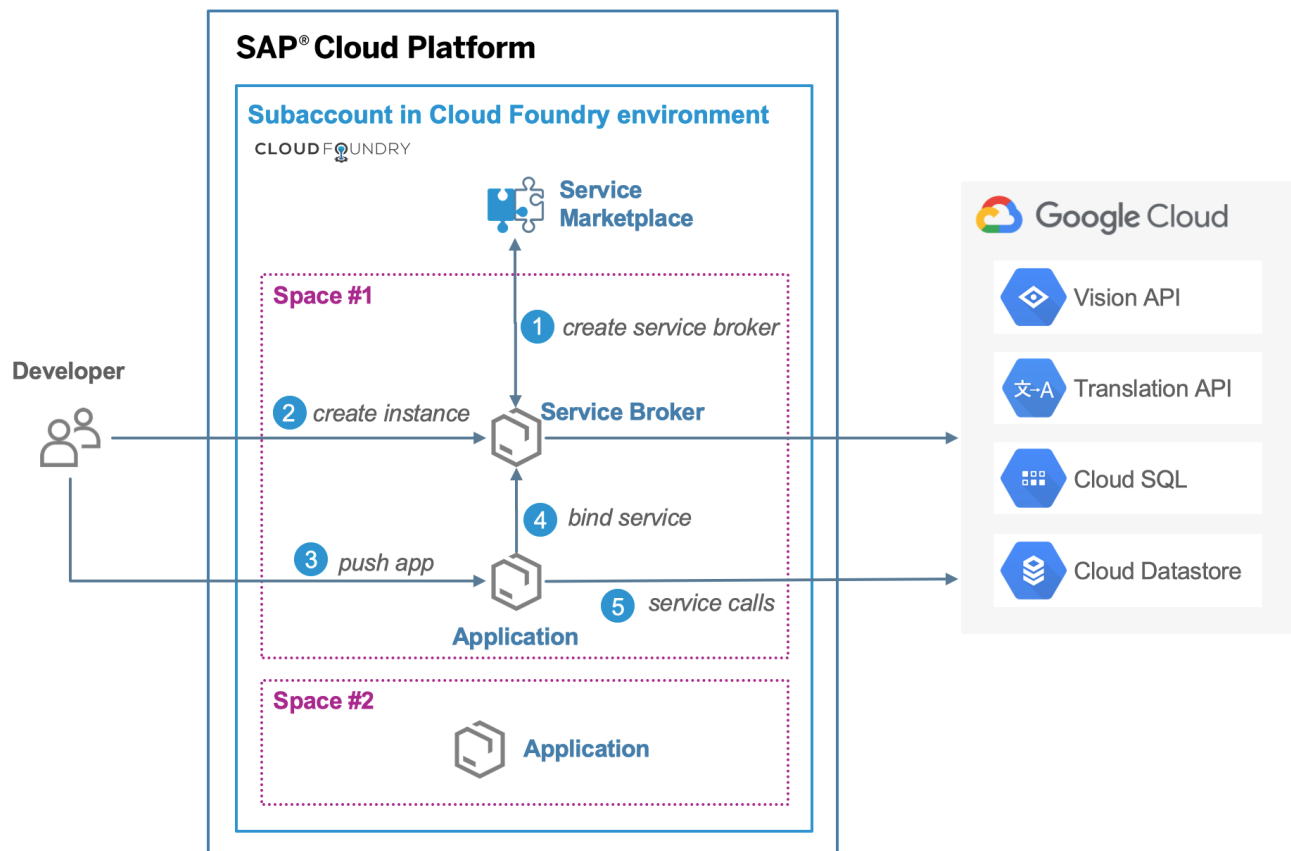


These instructions show you how to integrate Google Cloud services into a Cloud Foundry environment on SAP Cloud Platform by using the Open Service Broker for Google Cloud Platform (GCP Service Broker).

The GCP Service Broker simplifies the delivery of Google Cloud services to applications that run in a Cloud Foundry environment on SAP Cloud Platform. By creating Google Cloud resources and managing their corresponding permissions, the GCP Service Broker makes it easy to consume Google Cloud services from within an SAP Cloud Platform application environment.

The following diagram is a high-level view of the setup steps with some example Google Cloud services.



The following components are required for the set up of the GCP Service Broker on SAP Cloud Platform for Cloud Foundry.

Service	Use case
Open Service Broker for Google Cloud Platform (https://github.com/GoogleCloudPlatform/gcp-service-broker)	Integration of Google Cloud services into a SAP Cloud Platform Cloud Foundry environment
Cloud SQL for MySQL (/sql/docs/mysql/)	A database for back-end storage for GCP Service Broker
SAP Cloud Platform Account (https://account.hana.ondemand.com/#/home/welcome)	Provide the SAP Cloud Platform Cloud Foundry environment in the selected Google Cloud region
Google Cloud Project (/resource-manager/docs/creating-managing-projects)	Provide the Google Cloud services
Cloud Foundry CLI (https://docs.cloudfoundry.org/cf-cli/install-go-cli.html)	Command-line interface for configuring Cloud Foundry

Cloud SQL, which is recommended for use with GCP Service Broker, is a billable Google Cloud component.

The use of Google Cloud services through their respective APIs might also be billable.

Use the [Pricing Calculator](/products/calculator/#id=411d8ca1-210f-4f2c-babd-34c6af2b5538) (/products/calculator/#id=411d8ca1-210f-4f2c-babd-34c6af2b5538) to generate a cost estimate based on your projected usage.

Before you can set up the Cloud Foundry environment on SAP Cloud Platform you must first set up both your local development environment and create a project on Google Cloud.

1. Install the Cloud Foundry Command-Line Interface (CLI) on your development workstation as per [SAP Portal documentation](https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/4ef907afb1254e8286882a2bdef0edf4.html) (<https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/4ef907afb1254e8286882a2bdef0edf4.html>)

If you do not already have a Google Cloud project with billing enabled, you must create one.

1. [Sign in](https://accounts.google.com/Login) (<https://accounts.google.com/Login>) to your Google Account.

If you don't already have one, [sign up for a new account](https://accounts.google.com/SignUp) (<https://accounts.google.com/SignUp>).

2. In the Cloud Console, on the project selector page, select or create a Cloud project.

★ **Note:** If you don't plan to keep the resources that you create in this procedure, create a project instead of selecting an existing project. After you finish these steps, you can delete the project, removing all resources associated with the project.

[Go to the project selector page](https://console.cloud.google.com/projectselector2/home/dashboard) (<https://console.cloud.google.com/projectselector2/home/dashboard>)

3. Make sure that billing is enabled for your Google Cloud project. [Learn how to confirm billing is enabled for your project](#) (</billing/docs/how-to/modify-project>).

You need certain Google Cloud APIs enabled, a Google Cloud service account, and a database for the GCP Service Broker.

Enable the following APIs in [API Manager > Library](https://console.cloud.google.com/apis/library) (<https://console.cloud.google.com/apis/library>).

1. Enable the [Resource Manager API](#)

(<https://console.cloud.google.com/apis/api/cloudresourcemanager.googleapis.com/overview>).

2. Enable the Cloud Identity and Access Management API (<https://console.cloud.google.com/apis/api/iam.googleapis.com/overview>).
3. Enable the Cloud SQL Admin API (<https://console.cloud.google.com/apis/api/sqladmin/overview>).
4. Enable the API for any other supported Google Cloud service (<https://github.com/GoogleCloudPlatform/gcp-service-broker/blob/master/README.md>) that you need.

1. In the Cloud Console, open the **Service accounts** page.

OPEN Service accounts page (<https://console.cloud.google.com/iam-admin/serviceaccounts>)

2. Select your Google Cloud project.
3. Click **Create Service Account**.
4. Enter a name in the **Service account name** field.
5. Click **Create**.
6. In the **Select a role** list, click **Project > Owner**, which provides the permissions that are necessary for the GCP Service Broker to work with Google Cloud services.
7. Click **Continue**.
8. In the **Create key (optional)** section, click **Create Key**. Make sure the **JSON** key type is specified.
9. Click **Create**. The JSON key file is automatically downloaded to your workstation.
10. Click **Done**.
11. Move the JSON key file to a secure location.

The GCP Service Broker requires a MySQL database to store the state of provisioned resources. A Second Generation instance of Cloud SQL for MySQL, which provides a fully managed MySQL Community Edition database with automatic backups, high availability, and automatic maintenance, is recommended. However, you can use any database compatible with the MySQL protocol.

The following steps create a database by using a Second Generation Cloud SQL instance on Google Cloud, but should be applicable to any database compatible with the MySQL protocol.

Before you can create a database, you need to create a Cloud SQL for MySQL instance on Google Cloud.

1. In the Cloud Console, go to the **Cloud SQL Instances** page.

[OPEN Cloud SQL Instances page \(https://console.cloud.google.com/sql/instances\)](https://console.cloud.google.com/sql/instances)

2. Click **Create instance**.

- a. Select **MySQL** and click **Next**.
- b. Click **Choose Second Generation**.
- c. Define an **Instance ID**.
- d. Define a **Root password**.
- e. In the **Region** field, select the region in which your Cloud Foundry environment will be located.
- f. Click **Create**.

1. After the Cloud SQL instance boots, open Cloud Shell.

[OPEN Cloud Shell \(https://console.cloud.google.com/cloudshell\)](https://console.cloud.google.com/cloudshell)

2. Connect to the Cloud SQL instance.

Where ***your-cloud-sql-instance*** is the name of the Cloud SQL instance you created on Google Cloud.

3. Create a database called **servicebroker**.

4. Define a username and password for the GCP Service Broker.

Where:

- ***SB_DB_UserName*** is the username that represents GCP Service Broker as a user of the database.
- ***SB_DB_Password*** is the password that identifies the GCP Service Broker to the database.

5. Set access privileges for the GCP Service Broker.

6. Set up SSL and get the SSL certificates.

a. In the Cloud Console, open the **Cloud SQL Instances** page.

OPEN Cloud SQL Instances page (<https://console.cloud.google.com/sql/instances>)

b. Click the name of your Cloud SQL instance.

c. On the **Instance details** page, click the **Connections** tab.

d. In the **SSL** section under **SSL connections**, click **Allow only SSL Connections**. The update might take a few seconds.

e. In the **SSL** section, under **Configure SSL client certificates**, click **Create a client certificate**.

f. Enter a name and click **Create**.

g. Download the three certificate files and save them to a secure location.

7. In the **Connectivity** section of the **Connections** tab on the **Instance details** page:

- a. Confirm that the **Public IP** box is selected.
 - b. Under **Authorized networks**, click **Add network**.
 - c. In the **Network** field, enter `0.0.0.0/0`.
 - d. Click **Done**.
8. Click **Save** to save authorization changes.

1. Register for an [SAP Cloud Platform trial account](https://developers.sap.com/tutorials/hcp-create-trial-account.html) (<https://developers.sap.com/tutorials/hcp-create-trial-account.html>). You need an active SAP account to get an SAP Cloud Platform trial account.
2. Log in to the [SAP Cloud Platform Cockpit](https://account.hana.ondemand.com/cockpit) (<https://account.hana.ondemand.com/cockpit>).
3. In the SAP Cloud Platform Cockpit, choose **Home** at the top of the screen.
4. In the navigation pane on the left side of the screen, click **Regions**.
5. Under **Cloud Foundry Environment**, select a region that uses Google Cloud infrastructure, such as **US-Central (IA)**. This creates an organization and space within the SAP Cloud Platform Cloud Foundry environment.

In a terminal on your development workstation:

1. Install the latest version of GCP Service Broker for Cloud Foundry from GitHub.

2. Extract the GCP Service Broker files.

3. Switch directories.

4. Log in to the SAP Cloud Platform Cloud Foundry environment.

5. Enter the API endpoint.

Where ***techkey*** is an SAP-defined technical key that identifies the region of the API endpoint. For example, the technical key for the US Central region that includes the Google Cloud infrastructure is ***cf.us30***. For a list of all of the regions for the SAP Cloud Platform Cloud Foundry environment, see [SAP Help Portal](#)

(<https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/350356d1dc314d3199dca15bd2ab9b0e.html#loiof344a57233d34199b2123b9620d0bb41>)

.

6. Enter your username and password for your SAP Cloud Platform account.

7. Deploy the GCP Service Broker app to the Cloud Foundry environment.

Where ***APP_NAME*** is a unique name that you choose for the GCP Service Broker application. For example, ***gcp-service-broker-001*** or ***mycompany-gcp-svc-broker-app***.

You see information similar to the following example after Cloud Foundry deploys the GCP Service Broker app:

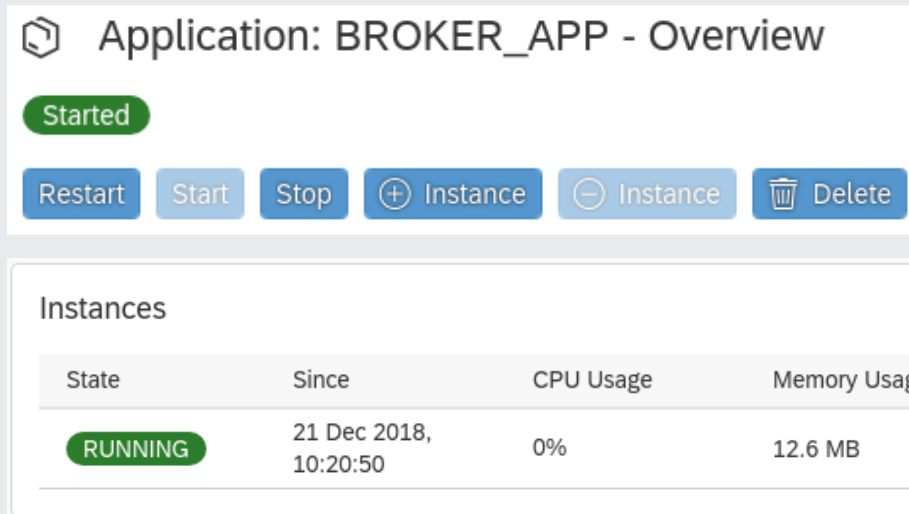
1. Log in to [SAP Cloud Platform Cockpit](https://account.hana.ondemand.com/cockpit) (<https://account.hana.ondemand.com/cockpit>).
2. In the SAP Cloud Platform Cockpit, go to **Trial Home > Regions > Your Google Cloud region > Your global account > Your subaccount > Spaces > Your space**.
3. Click the name of the GCP Service Broker app that you deployed in [the preceding section](#) (`#deploy_gcpsb`).
4. In the menu on the left hand side, click **User-provided variables**.
5. Use the **Add variable** button to define the following GCP Service Broker variables as key-value pairs:

Key	Value
	The contents of the server certificate file, <code>server-ca.pem</code> , that you downloaded when you set up SSL in Set up a backing database (<code>#set_up_a_backing_database</code>).
	The contents of the client certificate file, <code>client-cert.pem</code> , that you downloaded when you set up SSL in Set up a backing database (<code>#set_up_a_backing_database</code>).

Key	Value
	The contents of the client key file, <code>client-key.pem</code> , that you downloaded when you set up SSL in Set up a backing database (<code>#set_up_a_backing_database</code>).
	The public IP address of your Cloud SQL instance, as shown in the Connect to this instance section of the Instance details page for your Cloud SQL instance in the Cloud Console (https://console.cloud.google.com/sql/instances).
	The password that you defined for GCP Service Broker to access the database that you configured in Set up a backing database (<code>#set_up_a_backing_database</code>).
	The username that you defined for the GCP Service Broker when you created the database. For a Cloud SQL database, you can see the username under MySQL user accounts on the Users tab of the Instance details page for your Cloud SQL instance in the Cloud Console (https://console.cloud.google.com/sql/instances).
	The contents of the JSON key file that you downloaded when you created the root service account (<code>#create_service_account</code>).
	A username that you define for the service broker to use when authenticating broker requests. You specify this same username later in the <code>cf create-service-broker</code> command when you create the broker service.
	A password that you define for the service broker to use when authenticating broker requests. You specify this same password later in the <code>cf create-service-broker</code> command when you create the broker service.
Optional environment variables	For a list of optional variables that you can use to customize the GCP Service Broker, see Installation Customization (https://github.com/GoogleCloudPlatform/gcp-service-broker/blob/master/docs/customization.md)

1. Click **Overview** in the menu on the left hand side.

2. To start the broker application, click **Start**. When the application is up and running, green **Started** and **Running** indicators appear on the **Overview** page, as shown in the following images.



Application: BROKER_APP - Overview

Started

Restart Start Stop + Instance - Instance Delete

Instances

State	Since	CPU Usage	Memory Usage	Disk Usage
RUNNING	21 Dec 2018, 10:20:50	0%	12.6 MB	328.6 MB

3. If the GCP Service Broker app won't start or shows **Error** or **Crashed**, check the application logs from your development workstation and adjust the variables or configuration accordingly.

On your development workstation:

1. Connect to SAP Cloud Platform Cloud Foundry environment.
2. List the deployed applications.

You see output similar to the following example:

3. Register the broker service.

Where:

- ***BROKER_SERVICE_NAME*** is a unique name that you choose for your broker service.
- ***SECURITY_USER_NAME*** is the username that was defined to authenticate broker requests in [Configure the GCP Service Broker app for Cloud Foundry](#) (#configure_gcpsb_app).
- ***SECURITY_USER_PASSWORD*** is the password that was defined to authenticate broker requests in [Configure the GCP Service Broker app for Cloud Foundry](#) (#configure_gcpsb_app).
- ***BROKER_APPLICATION_ROUTE*** is the URL of the service broker application as displayed in the previous step. For example:

4. Verify the registration of the broker service.

You should see output similar to the following example:

5. Verify that Google Cloud services are now available in the SAP Cloud Platform Service Marketplace.

The following example, which was edited for readability and to fit the page, shows two services from a listing of Google Cloud services in the marketplace.

For a list of the available Google Cloud services, see [Open Service Broker for Google Cloud Platform](https://github.com/GoogleCloudPlatform/gcp-service-broker/blob/master/README.md) (<https://github.com/GoogleCloudPlatform/gcp-service-broker/blob/master/README.md>).

Create Google Cloud service instances and bind them to your SAP Cloud Platform applications.

For more information about creating and binding Google Cloud services, see:

- [Using Services in the Cloud Foundry Environment](https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/f22029f0e7404448ab65f71ff5b0804d.html)
(<https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/f22029f0e7404448ab65f71ff5b0804d.html>)
in the SAP Help Portal.
- [The GCP Service Broker documentation](https://github.com/GoogleCloudPlatform/gcp-service-broker/blob/master/docs/use.md)
(<https://github.com/GoogleCloudPlatform/gcp-service-broker/blob/master/docs/use.md>) in GitHub.