Google Cloud

Provider Managed BigQuery Projects

Using BigQuery to provide Data Warehouse as a Service

Last Updated: Jun 3, 2025

Author: Balazs Pinter, Deepinder Dhuria, Tom Cannon

Overview

Many Independent Software Vendors (ISVs) and data providers aim to share data with their customers using BigQuery. While BigQuery offers a robust data sharing capability through **BigQuery Sharing** (BQS, formerly known as Analytics Hub), this feature inherently requires data consumers to have an existing BigQuery environment provisioned within Google Cloud. This poses a challenge for customers who do not have a Google Cloud footprint or prefer not to manage one.

This document outlines how data providers can leverage the concept of **provider-managed projects** - a "do-it-yourself" approach to set up and manages dedicated BigQuery projects on behalf of their customers, allowing customers to access and consume the shared data without needing to provision or manage their own Google Cloud environments. This approach is often referred to as providing a Data Warehouse as a Service.

The Appendix includes specific guidelines for a 'Solution Accelerator' that provides automation scripts and other assets for some of the common scenarios (e.g. read-only access via API).

Table of Contents

Terminology

- **Provider**: The party, usually an ISV or Data Provider, that owns and manages the BQ project on behalf of the customer. They are being billed by Google Cloud for the resources consumed in the managed project.
- **Consumer**: The party that uses the provided BQ project. Consumers may query data that has been shared with them by the provider, or in some cases bring additional data into the project to execute aggregate queries.
- **Provider Managed BigQuery Project (PMBQP)**: A BigQuery project that resides in the Provider's tenant, is owned and managed by them but is provided to be used by another party, the Customer.
    - **Data Warehouse as a Service (DWaaS):** This is another term for PMBQP, often used when talking to providers who tend to think in terms of the service they are providing, rather than the underlying technology.

Before you begin - key decisions

The critical first step is to define the type of PMBQP that will be provided. There are a number of fundamental decisions to be made and thinking these through at the outset will enable greater focus in the steps that follow. The key decisions are summarised below and the Appendix contains a Questionnaire and Decision Tree that enables the decisions to be worked through systematically.

Are they a Google Cloud / BigQuery customer?

- If so, it is likely that the best solution would be to share the data to their existing BigQuery environment using BigQuery Sharing.

What type of data is being shared?

- Provider sharing their own proprietary data with the consumers: For example, a provider of financial markets or company data.
- Provider sharing the consumers own data back to them: For example a SaaS company, e.g. an eCommerce or Marketing ISV that captures data on behalf of customers and is sharing it back to the customer's data warehouse.
- Consumer brings their own data: Consumers may wish to bring their own data so that they can broaden the analysis they can do, including joining it with the data from the provider.

How will the provider share the data from the source to the PMBQP?

- BQ Sharing: This enables zero-copy sharing of data using a publish and subscribe model. It also includes Usage Metrics including number of rows and bytes processed and subscriber ID logging.
- BQ Sharing with Data Clean Room controls: Same as BQS, but also includes Analysis Rules including Aggregation Rules, List Overlap Rules, Differential Privacy and (coming soon) Query Templates, that enable the provider to tightly control how the data can be interacted with.
- IAM Sharing / Authorized Views: This is also a zero-copy approach, using the IAM permissions and Authorized Views.
- Copying data from source to PMBQP: Results in a duplication of data and ongoing updates, but may be appropriate when the source data is not in BQ and a copy will need to be made and maintained in BQ - it may be more efficient to simply copy the data into the PMBQP, rather than copy it into one project in order to share it to another.

How will consumers access the data?

(These options are not mutually exclusive - multiple options may be combined)

- **BigQuery Studio in Cloud Console:** Advanced users may be comfortable with accessing the data directly from BigQuery studio (within Cloud Console).

    - In PMBQP, Cloud Console can NOT be configured to show only the BigQuery Studio UI, but the functionalities of the non-BigQuery features can be severely restricted.
    - Cloud Console can only be accessed using end-customer identities, including federated identities.
    - During the PMBQP provisioning workflow the provider needs to ensure that customers only get access to BigQuery, and customers can't create additional resources which would incur costs and / or may pose a security risk.
- **BigQuery API / BigQuery ODBC / JDBC Connector:** BigQuery API can be accessed either using end-user identities or service accounts. Command Line tools and 3rd party integrations usually access BigQuery API using downloaded service account keys or through service account impersonation via Workload Identity Federation.
    - ***3rd Party Application:*** APIs and ODBC/JDBC Connectors can also enable access via 3rd party applications, like Tableau or PowerBI.
- **Looker Studio: Looker Studio offers interactive, collaborative, and ad hoc reports and dashboards.**
    - **BigQuery connector**: Looker Studio has a built-in BigQuery connector that can easily connect to BigQuery datasets authenticating as the currently logged in user.
    - **Custom connector:** Building community connectors is another possibility in Looker Studio. In this case a custom SaaS API wrapper would sit in between BigQuery and the Looker Studio connector, developed by the data provider. This would give more control over the queries and make it impossible to access BigQuery directly.
      Note: This solution can be implemented without provider managed projects as well, given appropriate access control is in place
      on the provider side.
- **Looker:** Looker offers enterprise dashboards that are built on a Looker's semantic layer. While providing a richer experience than Looker Studio, the semantic layer means that it is likely not suited to use cases where customers can bring their own data, as this would also need to be modelled in the semantic layer.
- **Custom UI**: Providers can build a custom UI that abstracts the underlying tech away from the consumer. (***Not covered in this guide***)

What accounts/identities will the consumers use?

- **What type of account?** (They are not mutually exclusive - different consumers may have different needs)
    - **Google Accounts**: Google Workspace accounts (@customerorg.com), Personal Google Accounts (gmail.com), Service Accounts (Does not support Cloud Console)
    - **External (non-Google) Accounts / Federated Identities** from an external Identity Provider (Microsoft Entra ID, Okta, etc)
- ***Who manages the accounts?***

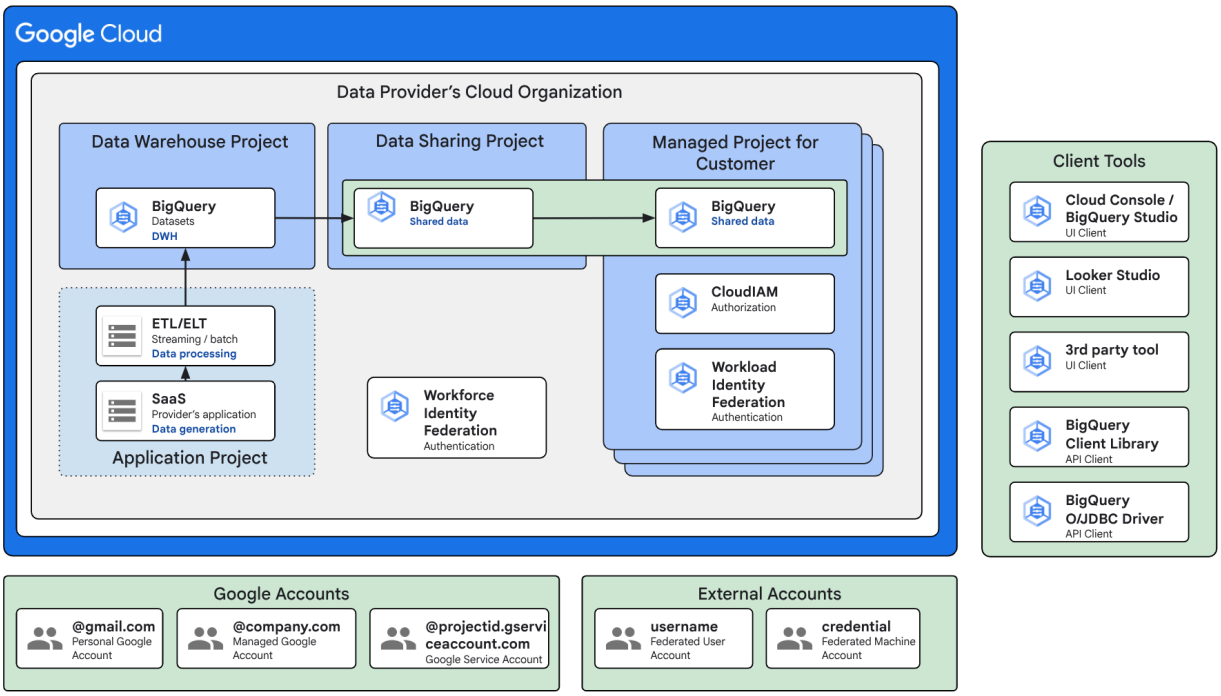
- Provider managed accounts
- Consumer managed accounts

Does the consumer have privacy concerns?

- *Is the consumer comfortable with the provider seeing their data and the logs of their query activity?*
  - Note: The provider can gain access to data and logs, as ultimately they own the project and can override settings. If the consumer is not comfortable with this, they can be given access to the immutable Audit Logs, so they can be notified immediately, which may be sufficient protection for some consumers.

Solution Architecture

The solution architecture depends on the choices the provider makes, the below diagram shows a high level overview of the options involved.



Detailed Guidelines

Consumer Identity Management

There are many different options available to provide the consumers with an identity to access the managed project, or alternatively, the consumers can also use their own identities through Workforce and / or Workload Identity Federation. The key questions to to consider for identity management are:

1. **What type of identities are going to be used?**
   The choice of identities will determine how the customers may access BigQuery, as they limit which tools can authenticate to Google Cloud. There are two main categories of identities that can access Google Cloud resources:

Identity type	Details

Google Accounts	• Google User Accounts
	  ○ Gmail Account (@gmail.com address)
	  ○ Google Workspace / Cloud Identity Account (@a-domain.com)
	• Google Cloud Service Accounts (@PROJECTID.iam.gserviceaccount.com address)

External (non-Google; federated accounts)	• End-user accounts (using ***Workforce*** Identity Federation)
	  ○ Microsoft Entra ID
	  ○ Okta
	  ○ Any OIDC Identity Provider
	  ○ Any SAML 2.0 Identity Provider
	• Machine accounts / workloads (using ***Workload*** Identity Federation)
	  ○ AWS
	  ○ Azure / Microsoft Entra ID
	  ○ Active Directory (ADFS)
	  ○ Kubernetes
	  ○ Any OIDC Identity Provider
	  ○ Any SAML 2.0 Identity Provider

Provider native accounts (non-Google; federated accounts)	This is a special case of the **External (non-Google; federated accounts)** case. Many providers are SaaS companies who are managing the identities of their end-customers already in their own systems.

	As Google Cloud's Workforce Identity federation supports standard compliant OIDC and SAML 2.0 Identity Providers, providers can implement an OIDC interface to act as an Identity Provider to Google Cloud and allow customers to reuse their existing login credentials.

The comparative identity type and supported consumer-experience (as evaluated in before you begin section) is as below:

	Cloud Console / BigQuery Studio	**BigQuery API / Google Cloud CLI / SDK**	**BigQuery O/JDBC connector**	**Looker Studio**	**3rd party tool (Tableau, PowerBI, ...)**
Google Account	Supported	Supported	Supported	Supported	Supported (likely; need to be checked for the specific tool)
Service Account (authenticated using Service Account Key)	Not supported	Supported	Supported	Supported	Supported
Service Account (impersonated)	Not supported	Supported	Supported	Supported (Only if the currently authenticated user is managed by Cloud Identity / Google Workspace)	Supported (maybe; need to be checked for the specific tool)
External: Workload Identity (Federated)	Not supported	Supported	Supported	Not supported	Supported (maybe; need to be checked for the specific tool)
External: Workforce Identity (Federated)	Supported	Supported	Supported	Not supported	Supported (maybe; need to be checked for the specific tool) * PowerBI supports identities federated from EntraID

2. Who is going to own / provision / manage the identities?

The identities can be owned by the provider or consumer and each has its pros and cons.

a. Provider owned identities

Provider provisions the identity(ies) for the consumer to use, and they manage the full lifecycle of the identities. Providers additionally need to give access to a list of identities for the consumer to be able to use PMBQP. In case of the provider managed identities, the following operational aspects have to be taken into consideration:

- Account provisioning / deprovisioning
- Account recovery (password reset, etc)
- Abuse protection
- Access / permission management
- Data handling (consent): Accepting Terms & Conditions (for example on first login attempt)
- Account management fees (if using an identity provider with per-account charges)
- Account lockdown (lateral movement, SSO integrations, etc)

Providers can use either Google Workspace / Cloud Identity or integrate with an external identity provider within their control to provision end user accounts

	Provider managed Google Workspace / Cloud Identity	**Provider integrates with an external identity provider**
Key Options	• Provider managed account provisioned in the provider's primary Cloud Identity / Google Workspace account (@dataprovider.com address) • Provider managed account provisioned in the provider's secondary domain (@manageusers.dataprovider.com address). • Domains can be added as alias domains in the primary Google Workspace / Cloud Identity Organization, or created as a separate Cloud Identity / Google Workspace organization. Having a	Providers may integrate with an external identity provider within their control (like Microsoft Entra ID, Okta, Keycloak) via Workload Identity Federation or Workforce Identity Federation and manage the lifecycle of the managed accounts there. This gives following options • Work*force* Identity Federation provides Federated Cloud Console access (including BigQuery Studio) meant for human users, in addition to API access. • Work*load* Identity Federation provides API access directly or

	separate organization increases isolation, but also operational complexity, because it requires separate billing and organization management. • Provider managed service account using downloaded Service Account key • Provider managed service account using customer-provided / uploaded Service Account key • Provider managed service account using Service Account impersonation through Workload Identity Federation	via Service Account Impersonation (as the preferred alternative to Downloaded Service Account keys)
Additional Considerations	• Customer identities managed in Cloud Identity / Google Workspace take up seats in the related subscription plan, and may require providers to pay for the seat. • Providers need to ensure that the identities can only access allowed Google Cloud APIs and Google Services, for example they should not be used for Google Docs or Google Drive, only Google Cloud Console. • These accounts should not be used as a Single-Sign-On (SSO) identity or any other services unrelated to the provider managed projects. • Google identities are managed in the Google Workspace Admin Console (admin.google.com). Additional steps should be taken to further restrict these accounts so they can only use BigQuery related features. Allowed Google Services can be enabled / disabled on a Global / Organization Unit (OU) or User level. Managed Identities	Additional considerations for that provider need to consider include: • The provider may integrate with an external identity management solution under their control using Workload or Workforce Identity Federation. In this model customer organizations may be represented by a Workforce Identity Pool in the provider's organization, or the different customer organizations may be represented as different attributes / claims on the external identities. Attributes of the external identities can be used to match a given set of external customers in Cloud IAM. • By default 100 workforce identity pools can be created in a Google Cloud organization, but it is possible to increase this limit through Google Cloud Support channels. • Provider owns:

	should be located under a dedicated OU, and all Google Services except Google Cloud should stay disabled. Cloud Shell access and Project Creation should be disabled under “Apps / Additional Google Services”. ● In order to limit the usage of the provider managed accounts to the PMBQP (Google Cloud) context, It’s also recommended to disable using provider-managed accounts for SSO in 3rd party applications: this is also configured in the Google Workspace Admin Console, under Security / Access and data control / API Controls / Settings / Unconfigured third-party apps / Don’t allow users to access any third-party apps ● In order to further limit the usage of the provider managed accounts to the resources the provider intends to share with them (likely residing in the provider’s Cloud Organization) Principal Access Boundary Policies can be applied. ● Provider owns: ○ Google Account (User account, Service account) ○ Access Management (Project level) ● Customer owns: ○ Nothing / they access the data using the provided credentials	○ Identity management in the 3rd party system (Keycloak, Okta, Entra ID, ...) ○ Workforce Identity Pool Configuration (Organizational level) ○ Workload Identity Pool Configuration (Project level) ○ Access Management (Project level) ● Consumer owns: ○ Nothing / they access the data using the provided credentials

b.

Consumer owned identities

Consumers will be responsible for the provisioning of the identities and the management of the full account lifecycle. Providers need to give access to a list of identities for the customers to be able to use the provider managed projects.

Consumers can use either Google Workspace / Cloud Identity or integrate with an external identity provider within their control to provision end user accounts

	Consumer managed Google Workspace / Cloud Identity	**Consumer managed external identities**
Key Options	Consumers can use Google Workspace / Cloud Identity to provision end-user accounts and Google Cloud APIs to provision Service Accounts in a Google Cloud project. This gives the following options: ● Customer managed account provisioned in the customer's Cloud Identity / Google Workspace account (@customerdomain.com address) ● Personal Gmail accounts (@gmail.com) ● Customer managed service account (using downloaded Service Account key; customer-provided / uploaded Service Account key; Service Account impersonation through Workload Identity Federation) - The authentication method is less relevant as it's owned by the customer.	Providers may integrate with the customer's external identity provider (like Microsoft Entra ID, Okta, Keycloak) via Workload Identity Federation or Workforce Identity Federation. ○ Workforce Identity Federation provides Federated Cloud Console access (including BigQuery Studio) meant for human users, in addition to API access. ○ Example: a user authenticated with their Entra ID account can access Google Cloud BigQuery Studio and query datasets there ○ Workload Identity Federation provides API access directly or via Service Account Impersonation (as the preferred alternative to Downloaded Service Account keys) ■ Example: an automation running on an AWS instance can authenticate to BigQuery and pull data from a dataset
Additional Considerations	Customers need to share the list of identities with the provider, or they may	The provider may integrate with the customer's identity management solution using Workload or Workforce

	use Google Group membership to manage access. Depending on the constraints on the provider side, they may or may not be allowed to have external identities added to their IAM policy bindings. The Domain Restricted Sharing Cloud Organization Resource Constraint can restrict users from which Cloud Identity / Google Workspace domains can be added. Customer owns: • Google Account (User account, Service account) Provider owns: • Access Management (Project level)	Identity Federation. In this model every customer organization would be represented by a Workforce Identity Pool in the provider's organization. By default 100 workforce identity pools can be created in a Google Cloud organization, but it is possible to increase this limit through Google Cloud Support channels. Customer owns: • Identity management in the 3rd party system (Keycloak, Okta, Entra ID, ...) Provider owns: • Workforce Identity Pool Configuration (Organizational level) • Workload Identity Pool Configuration (Project level) • Access Management (Project level)

Identity management matrix:

	Provider managed identity	Customer managed Google identity	Customer managed external identity
Pros	- Full end-to-end control over the identity lifecycle - Only known and managed identities appear in IAM policies - Customers do not need to have any relationship with Google	- Low operational complexity: only the addition / removal from IAM policies is required - Lower security risk as lateral movement is unlikely even in the case of less restrictive security configurations	- Mid operational complexity: the addition / removal from IAM policies is required, organization level workforce identity pool configuration management is also required - Lower security risk as lateral movement is unlikely even in the case of less restrictive security configurations
Cons	- Higher operational complexity due to the additional workflows needed (account recovery, provisioning, deprovisioning, etc) - Higher security risk in case of security misconfiguration (lateral movement) - In case of workforce identity federation: Looker studio and some other services are not available with the Workforce Identity Federation feature	- Customers need to provision / have a Google Account at least, all the rest can be managed by the provider	- Looker studio and some other services are not compatible with the Workforce Identity Federation feature

3. What IT security/governance restrictions apply to the provider?

Additionally providers need to assess their internal security controls and work closely with IT / Security to make sure the customer identities can be used to access the managed projects. Providers may spin up a secondary Cloud Organization dedicated to managed projects with different security controls. See the following (non-exhaustive) list:

- Can external identities get Cloud IAM permissions in the provider's Cloud Organization? (accounts with @customer.com domains, accounts with @gmail.com domains, external service accounts)
- Can Service Account Keys be uploaded and/or downloaded?
- Can Workload / Workforce Identity Federation be used?
    - Workload Identity Federation is configured on the project level, but Organization Policies may limit which workload identity pools can be used in a given project
    - Workforce Identity Federation is configured on the organization level
- What level of isolation is required / enforced across projects, when it comes to data sharing?
    - Is VPC Service Controls (VPC-SC) mandatory for all projects?

Public documentation:

- https://cloud.google.com/iam/docs/workload-identity-federation
- https://cloud.google.com/iam/docs/workforce-identity-federation
- https://cloud.google.com/iam/docs/migrate-from-service-account-keys
- https://cloud.google.com/identity/docs
- https://support.google.com/a/answer/7281227?hl=en#trustorlimit
- https://support.google.com/a/answer/9275380?hl=en
- https://cloud.google.com/iam/docs/principal-access-boundary-policies

Consumer Access Management

In order to access Google Cloud resources and call different Google Cloud APIs, including BigQuery, the requestor needs to have permissions. Permissions are granted on resources in Cloud IAM policies. In the managed BigQuery context, users need to read the data shared with them. The following roles are relevant:

- BigQuery Data Viewer (roles/bigquery.dataViewer)
    - When applied to a dataset, this role provides permissions to list all of the resources in the dataset (such as tables, views, snapshots, models, and routines) and to read their data and metadata with applicable APIs and in queries.
    - When applied at the project or organization level, this role can also enumerate all datasets in the project. Additional roles, however, are necessary to allow the running of jobs.
- BigQuery Job User (roles/bigquery.jobUser)
    - Provides permissions to run jobs, including queries, within the project.

In case the users are allowed to bring in their data, the following role grants the required permissions:

- BigQuery User (roles/bigquery.user)
    - When applied to a project, this role also provides the ability to run jobs, including queries, within the project. A principal with this role can enumerate their own jobs, cancel their own jobs, and enumerate datasets within a project.

    - Additionally, allows the creation of new datasets within the project; the creator is granted the BigQuery Data Owner role (roles/bigquery.dataOwner) on these new datasets.

In case the users are allowed to subscribe to 3rd party listings through Analytics Hub, they need:

- BigQuery User (roles/bigquery.user)
    - When applied to a project, this role also provides the ability to run jobs, including queries, within the project. A principal with this role can enumerate their own jobs, cancel their own jobs, and enumerate datasets within a project. Additionally, allows the creation of new datasets within the project; the creator is granted the BigQuery Data Owner role (roles/bigquery.dataOwner) on these new datasets.
- Analytics Hub Subscriber (roles/analyticshub.subscriber)
    - Lets users view and subscribe to listings. Subscription creates a Linked Dataset in the provider managed project.

As part of the onboarding process the provider needs to collect the user account information from the customers. The automation pipeline handling the resource creation needs to add the user account to the relevant IAM policies. Additionally providers need to ensure that no additional roles are granted to the users.

Note: Google Cloud Console today cannot be restricted to show only enabled Cloud services, like BigQuery Studio, so in all cases where customers are using their end-user credentials and not Service Account credentials, the users will be able to see all menu items, but will only be able to interact with BigQuery studio to query their own data.

Public documentation:

- https://cloud.google.com/bigquery/docs/access-control
- https://cloud.google.com/bigquery/docs/control-access-to-resources-iam
- https://cloud.google.com/bigquery/docs/analytics-hub-grant-roles

Consumer Data Sharing Options

In most cases, the reason for the PMBQP is to enable the consumers to access data. There are a number of ways for the Provider to share the data with the PMBQP. Let's imagine a scenario where a B2B Software-as-a-Service (SaaS) company wants to make end-customer data available to select consumers as a value added service (Data-as-a-product). Consumers who subscribe will get access to the data in a provider managed BQ project. The key consideration to evaluate sharing options include

- Data duplications vs. zero copy: replication speed, storage cost
- Share table vs. authorized view: query plan exposure: problem or not? Authorized views / views expose the query plan and show table names, etc, from the source tables
- Data locality: which BQ region / multi-region to use

	Data replication from BigQuery or another database to BigQuery	**Share customer-specific dataset using Analytics Hub**	**Authorized view to access the customer specific dataset in the source project**
Description	Data for each customer of the provider can be ingested into a customer-specific BigQuery Dataset in a provider managed project.	Similar to the previous solution, but instead of directly sharing the customer specific dataset, the provider opts to share data using Analytics Hub Exchanges / Listings.	Data providers can create a view in the managed project to query data of the customer from the central data warehouse. This view can be authorized to access the central data warehouse.
Pros	• Clear separation of shared data	• Zero-copying of data (lower storage and network costs) • Out of the box usage monitoring through Analytics Hub • Better separation of concerns, easier permission management	• Zero-copying of data (lower storage and network costs) • Query plan is revealed to the user and the provider has to be careful not to expose unintended internal details

		• No end-customer identities in the source projects • Query plan is not revealed to the user if a table is queried through the linked dataset • Better control over queries (in the case of DCR templated queries)	
Cons	• If the provider maintains a central data warehouse for all of their customers, this solution leads to data duplication. • Multiple data pipelines (source -> central DwH, central DwH -> customer specific dataset)	• More complex automation to manage environments	• More complex automation required to create per-customer views and authorizations.

Public documentation:

- https://cloud.google.com/bigquery/docs/authorized-views
- https://cloud.google.com/bigquery/docs/analytics-hub-introduction
- https://cloud.google.com/bigquery/docs/reference/analytics-hub/rest

PMBQP Monitoring

BigQuery usage monitoring in Cloud Monitoring

The usage monitoring of BigQuery can be done using a monitoring dashboard. The public documentation is as below:
https://cloud.google.com/monitoring/api/metrics_gcp#gcp-bigquery
https://cloud.google.com/bigquery/docs/monitoring-dashboard

BigQuery Sharing (formerly AnalyticsHub) usage metrics

If BigQuery Sharing is selected for sharing, the metrics can be monitored as per below documentation:
https://cloud.google.com/bigquery/docs/analytics-hub-monitor-listings#use-analytics-hub

Audit Logging and INFORMATION_SCHEMA

INFORMATION_SCHEMA

INFORMATION_SCHEMA views in BigQuery are another source of insights that you can use along with metrics and logs. These views contain metadata about jobs, datasets, tables, and other BigQuery entities. For example, you can get real-time metadata about which BigQuery jobs ran during a specified time. Then, you can group or filter the results by project, user, tables referenced, and other dimensions.

INFORMATION_SCHEMA views provide you information to perform a more detailed analysis about your BigQuery workloads, such as the following:

- What is the average slot utilization for all queries over the past seven days for a given project?
- What streaming errors occurred in the past 30 minutes, grouped by error code?

BigQuery Audit Logs

BigQuery audit logs contain log entries for API calls, but they don't describe the impact of the API calls. A subset of API calls creates jobs (such as query and load) whose information is captured by INFORMATION_SCHEMA views.

For example, you can find information about the time and slots that are utilized by a specific query in INFORMATION_SCHEMA views but not in the audit logs. BigQuery audit logs can include information that users might consider sensitive, such as SQL text, schema definitions, and identifiers for resources such as tables and datasets. If given access, Cloud Audit Logging

and BigQuery Audit Logs can give customers of the managed projects visibility into provider actions.

a. **Audit Logging for providers**

   Providers can see customer administrative actions and BigQuery data access by default. In a provider managed BigQuery project customers should only have access to BigQuery, where audit logging provides the following insights:
   - If a query job is created in the managed project: job creation including the query and additional details
   - If a query job is created in a different project, querying data from BigQuery datasets in the managed project: the fact of job creation is recorded, without the exact query, but including the tables and fields queried.
   - If the user of the managed project has write access to BigQuery (they are allowed to bring their data in), all administrative actions are logged as well

b. **Audit Logging for consumer users of the managed projects**

   Users of the managed projects can see all administrative actions and BigQuery data access by default, including provider's access. In a provider managed BigQuery project providers usually have (can have) full administrative and data access to BigQuery, where audit logging provides the following insights:
   - If a query job is created in the managed project: job creation including the query and additional details
   - If a query job is created in a different project, querying data from BigQuery datasets in the managed project: the fact of job creation is recorded, without the exact query, but including the tables and fields queried.
   - All administrative actions across all Google Cloud services are logged by default

The ability for users (i.e. consumers) to see audit logs may be a valuable way of addressing their privacy concerns. It may be that organizations are willing to accept the technical possibility that the provider could access their data or queries (presumably in breach of contract) as long as they are able to verify that they have not done so.

Public documentation:
- https://cloud.google.com/monitoring/api/metrics_gcp#gcp-bigquery
- https://cloud.google.com/bigquery/docs/monitoring-dashboard
- https://cloud.google.com/bigquery/docs/analytics-hub-monitor-listings#use-analytics-hub
- https://cloud.google.com/bigquery/docs/information-schema-intro
- https://cloud.google.com/bigquery/docs/reference/auditlogs
- https://cloud.google.com/bigquery/docs/monitoring
- https://cloud.google.com/bigquery/docs/monitoring-dashboard

Consumer bringing in their own data / Bring Your Own Data / BYOD

In addition to analyzing / visualizing the data shared with the consumer using BigQuery managed projects, they may want to bring in their own data to join with the data shared by the provider. This is also possible using Provider Managed BigQuery Projects by enabling the required services and granting additional (write) permissions to additional services.

Below are the services (non exhaustive) you can consider to load data into BigQuery.

1. **Services to consider**

    a. Cloud Storage
    Cloud Storage can store the data files used by BigQuery external tables or BigLake tables. Supported formats: Comma-separated values, JSON NL, Avro, ORC, Parquet, Delta Lake, Iceberg, Datastore exports, Firestore exports.
    b. BigQuery Data Transfer Service
    The BigQuery Data Transfer Service automates data movement into BigQuery on a scheduled, managed basis. It supports many sources, including Cloud Storage, Amazon S3, Amazon Redshift, Azure Blob Storage.
    c. Dataflow / Dataproc
    Managed services for streaming and batch processing.
    d. Cloud Pub/Sub
    Google Cloud Pub/Sub is a fully-managed, scalable, global and secure messaging service that allows you to send and receive messages among applications and services. BigQuery can ingest data directly from Pub/Sub
    e. Datastream
    Datastream is a fully managed serverless service that enables Change Data Capture loads for bigquery

2. **Cost factors to consider (non-exhaustive list)**

    a. Storage costs: data storage is charged differently by the used services (BigQuery, Cloud Storage)
    b. Network ingress: network ingress is free of charge in general, but network egress fees may apply
    c. External costs: costs can be incurred outside of Google Cloud by using the BigQuery Data Transfer Service, such as AWS or Azure data transfer charges.
    d. BigQuery ingestion
        i. Free: Batch Loading (via Data Transfer Service or Batch load jobs)
        ii. Charged: Streaming inserts (tabledata.insertAll), BigQuery Storage Write API

e. CloudStorage charges for operations and transfers that are not within the same region

3. **Monitoring aspects to consider**

   a. Carefully monitor and alert on costs across all services that allow write access using the Google Cloud Observability tooling (Cloud Monitoring and Alerting)
   b. Analyze usage patterns to detect potential abuse

4. **Access control aspects to consider**

   a. Ensure that public access is disabled where applicable, for example
      i. Public access prevention is enforced on GCS buckets
      ii. Signed URLs are disabled on Google Cloud Storage
   b. Consumers don't have permissions to change IAM policies anywhere in the managed project
   c. Consumers don't have permissions to enable / disable Google Cloud APIs

Giving any level of write access to the consumers needs careful consideration from many aspects, including costs, monitoring and abuse potential. We recommend tailoring the "BYOD" solution to the individual user needs and building targeted solutions with adequate monitoring rather than giving generic, broad access. Additionally if the consumers need complex data ingestion pipelines, the recommendation is to steer the customers towards becoming a Google Cloud customer and continuing the work in their own environment.

For example if the consumers are okay bringing their own data as CSV, it is possible to grant very limited roles that allow the users only to upload CSV data to a Cloud Storage bucket, and importing that data using a BigQuery Batch ingestion via a Load Job.

Public documentation:

- https://cloud.google.com/bigquery/docs/loading-data
- https://cloud.google.com/bigquery/docs/dts-introduction
- https://cloud.google.com/storage/docs/public-access-prevention
- https://cloud.google.com/storage/docs/access-control/signed-urls

Billing / cost control

1. Cost factors to consider

BigQuery pricing has two main components:

- Compute pricing is the cost to process queries, including SQL queries, user-defined functions, scripts, and certain data manipulation language (DML) and data definition language (DDL) statements.
    - On-demand: By default, queries are billed using the on-demand (per TiB) pricing model, where you pay for the data scanned by your queries.
    - Capacity: BigQuery offers a capacity-based compute pricing model for customers who need additional capacity or prefer a predictable cost for query workloads rather than the on-demand price (per TiB of data processed).
- Storage pricing is the cost to store data that you load into BigQuery.
    - Active storage includes any table or table partition that has been modified in the last 90 days.
    - Long-term storage includes any table or table partition that has not been modified for 90 consecutive days. The price of storage for that table automatically drops by approximately 50%. There is no difference in performance, durability, or availability between active and long-term storage.
    - Metadata storage includes storage for logical and physical metadata for datasets, tables, partitions, models and functions stored in the BigQuery metastore.

Additionally, network ingress and egress costs may apply:

- Data ingestion - BigQuery offers two modes of data ingestion:
    - Batch loading. Load source files into one or more BigQuery tables in a single batch operation.
    - Streaming. Stream data one record at a time or in small batches using the BigQuery Storage Write API or the legacy streaming API.
- Data extraction pricing - BigQuery offers the following modes of data extraction. You are charged for data transfer when you export data in batch from BigQuery to a Cloud Storage bucket or Bigtable table in another region:
    - Batch export. Use an extract job to export table data to Cloud Storage. There is no processing charge for exporting data from a BigQuery table using an extract job.
    - Export query results. Use the EXPORT DATA statement to export query results to Cloud Storage, BigTable, or Spanner. You are billed for the compute cost to process the query statement.

- Streaming reads. Use the Storage Read API to perform high-throughput reads of table data. You are billed for the amount of data read.

You are not charged for data extraction or data transfer when accessing query results in the Google Cloud console, BigQuery API, or any other clients, such as Looker.

2. Customer price plan (model)

Naturally a top concern for the providers is the cost model (how they charge their customers for using the provider managed projects) and the cost control (how they block their users from generating unexpected costs).

Providers need to define pricing plans for their users and implement the appropriate cost controls and visibility for the customers of the managed projects.

a. Prepaid / tiered model

In this model the customers can pick a tier based on their query volume. This may be difficult to predict, and the provider may need to analyze user behavior to come up with sensible pricing.
Example tiers

- Free: only allowed to use the free usage limits
- Small (S): X GiB scanned in BigQuery + no 3rd party tool usage to avoid network egress costs
- Medium (M): Y GiB scanned in BigQuery + 10 GiB network egress
- Large (L): Z GiB scanned in BigQuery + 20 GiB network egress

b. Postpaid / pay-for-use model

In this model the customers are charged for their usage. It is crucial to still prevent runaway costs and proactively notify customers reaching a certain threshold.

c. Mix of two: baseline usage included, pay-for-extra

In this model customers are paying a flat fee + overages based on their usage.

3. Cost attribution to customers

The provider needs visibility into the costs of the provider managed projects, and potentially needs to proactively act in case of runaway costs.

- Billing exports can provide granular visibility into costs.
- Per-project billing budgets provide a single cost value for the projects involved.
- Looker dashboards on billing data exported into BigQuery provide granular visibility into costs.

4. Cost control / quotas

There are three key decisions to make in order to monitor costs and keep it under control.

a. BigQuery pricing model model

The first important decision is to decide which pricing model to select for query processing. Each pricing model has different means of cost control. As this topic goes beyond the scope of the document we only provide a short overview here and recommend reading the BigQuery cost control whitepaper.

The following pricing models are available:

- On-demand pricing: You pay for the data scanned by your queries. You have a fixed, per-project query processing capacity, and your cost is based on the number of bytes processed.
- Flat-rate pricing: You purchase dedicated query processing capacity.

Some providers start with on-demand pricing and gather data about the usage patterns to establish a baseline capacity estimate. Based on the data then it's possible to switch to a capacity based pricing model with reserved capacity.

b. Combined / layered cost control approach

The second important decision is to decide how to implement cost control. There are two possible levels of controlling and tracking costs, BigQuery Level (depends on the pricing model) and Project Level through the Billing Account associated with the project.

i. Custom Quotas (BigQuery level on-demand pricing)

The provider can limit costs across multiple projects by setting a custom quota that specifies a limit on the amount of data processed per day. Daily quotas are reset at midnight Pacific Time.

Custom quota is proactive, so it is not possible to run an 11 TB query if there is a 10 TB quota specified. Custom quota can be applied on the project level and limits the amount of data processed by BigQuery in the context of the given project. It can be enforced across all users of BigQuery in the project (QueryUsagePerDay), or enforced for individual users (QueryUsagePerUserPerDay).

- QueryUsagePerDay: Project-level custom quotas limit the aggregate usage of all users in that project.

- QueryUsagePerUserPerDay: User-level custom quota is separately applied to all users and service accounts within a project.

This approach has the limiting effect of putting a “hard stop” on usage once the limit has been reached, which may be jarring for the end-user.

ii. Slots and Reservations (BigQuery level capacity based pricing)

With the capacity-based model, the provider pays for the slot capacity allocated for queries over time. This model gives the provider explicit control over total slot capacity, whereas the on-demand model does not. The provider explicitly chooses the amount of slots to use through a reservation, which can be allocated to a project. The provider can specify the amount of slots in a reservation as a baseline amount which is always allocated, or as an autoscaled amount, which is allocated when needed.

When query demands exceed slots you committed to, you are not charged for additional slots, and you are not charged for additional on-demand rates. Your individual units of work queue up.

This approach has the limiting effect of slowing down query performance when a limit is reached, which may be less jarring than a hard stop.

iii. Billing account budgets and alerts (Project level)

Project level budgets can be used to monitor charges of a Google Cloud project. Budgets let you track your actual Google Cloud costs against your planned costs. After you've set a budget amount, you set budget alert threshold rules that are used to trigger email notifications or automation through Pub/Sub. Budget alert emails help you stay informed about how your spend is tracking against your budget. You can also use budgets to automate cost control responses.

This approach enables soft limits via notifications and hard limits via automation

We recommend a multi layered approach that includes both Project and BigQuery level controls:

- **Layer 1: Budgets & Alerts (Soft-limit):** Track costs on the project level using billing account budgets and alerts and configure alerting at a certain threshold, for example at 50% of the estimated cost based on the anticipated usage
- **Layer 2: Custom quotas, Slots & Reservations (Hard-limit)**: Implement cost control on BigQuery level (on-demand or capacity based)

- **Layer 3: Budgets with Automation (Hard-limit)**: Implement cost control on the project level using billing account budgets and alerts integrated with an automation that takes the desired action (revoke access, shutdown project, notify, etc)

c. Actions to take in case of runaway costs

The third important decision is to decide what action(s) to take in case of runaway costs or unexpected forecasted costs.

A mix of the following actions can be implemented:

- Send early warning based on cost forecast (internal) and investigate the root cause
- Send early warning based on cost forecast (external, to the user)
- Remove user access
- Disable billing (detach Billing Account from the project)
- Shutdown the project

Relevant public documentation:

- https://cloud.google.com/resources/bigquery-pricing-whitepaper
- https://cloud.google.com/bigquery/pricing
- https://cloud.google.com/bigquery/docs/reservations-intro
- https://cloud.google.com/billing/docs/how-to/budgets-programmatic-notifications
- https://github.com/Cyclenerd/poweroff-google-cloud-cap-billing
- https://github.com/GoogleCloudPlatform/terraform-google-billing-dashboard
- https://cloud.google.com/bigquery/pricing#data_extraction_pricing
- https://cloud.google.com/bigquery/pricing#data-extraction-pricing-details

Appendices

Appendix A: Assessment Questionnaire & Decision Tree

Are your data-consumer customers already Google Cloud customers?

- ☐ Yes - Use BigQuery Data Sharing to share data
- ☐ No - Create a provider managed project to share data with your customers using BigQuery Data Sharing
- ☐ Both - BigQuery Data Sharing can be used on the provider side in both cases to create a customer-specific shared dataset that can be subscribed to either by the customer (if already a Google Cloud customer) - or by the provider in the per-customer managed project (if not a Google Cloud customer yet)

Are your customers bringing their own data into a provider managed BigQuery project, requiring write access?

- ☐ Yes - Write access can be granted in Cloud IAM, although special care must be taken to control costs of BigQuery storage (or Google Cloud Storage). Data shared through Analytics Hub is always read-only, but can be joined with additional data brought in.
- ☐ No - Data shared through Analytics Hub is always read-only, in addition Cloud IAM can be used to enforce read-only access (don't allow the creation of new datasets).

What tools are your customers using to access the data shared with them using BigQuery? What is the end-user experience?

- ☐ BigQuery Studio in Cloud Console - Power users may use BigQuery studio directly to query the data shared with them. BigQuery studio works with Google Accounts and federated identities (Workforce Identity Federation)
- ☐ BigQuery API directly (BQ CLI, BQ J/ODBC drivers, direct API calls through API libraries etc..) - Power users may use BigQuery CLI or API directly to query data shared with them and / or develop custom integration to their own systems.
- ☐ Looker Studio - Customers may already be familiar with the free Looker Studio that provides an easy to use interface to create dashboards and reports. Looker studio requires Google Accounts to access the tool, and does not work with federated identities.
- ☐ 3rd party tools (e.g. Microsoft PowerBI, Tableau) - Customers may be using a 3rd party tool to access data. These applications often use downloaded service account keys for authentication and it's unlikely that they support federated (Workload / Workforce) identities. The supported authentication methods need to be evaluated on a tool-by-tool basis.
- ☐ Custom tooling / UI (SaaS) developed and provided by the provider (for example a CLI tool to fetch the data to BigQuery using the API and custom authentication, custom Looker Studio Connector)

What type of identities / accounts are your customers using to access the data shared with them via BigQuery?

- ☐ Google Account / Personal Google Accounts (gmail.com) - Supports using Cloud Console / BigQuery Studio, Looker Studio, etc.
- ☐ Google Account / Google Workspace account that belongs to an organization (@customerorg.com) - Supports using Cloud Console / BigQuery Studio, Looker Studio, etc.
- ☐ Google Account / Service Account - Does not support Cloud Console / BigQuery Studio but most 3rd party BI tools support authentication using Service Account Keys.
- ☐ External Account / Federated Identity from an external Identity Provider (Microsoft Entra ID, Okta, misc SAML/OIDC provider) - Supports using Cloud Console / BigQuery Studio / BigQuery API but does not support Looker Studio. 3rd party BI tools may or may not support federated identities.
- ☐ Provider managed accounts: see the next question for details.

Does the provider need to provide / manage identities for the customers?

- ☐ Yes / federated end-user accounts: the provider can provision user accounts in a 3rd party identity management system they own / control, and use Workforce Identity federation to provide access to BigQuery Studio and BigQuery CLI / APIs. Supports using Cloud Console / BigQuery Studio / BigQuery API but does not support Looker Studio. 3rd party BI tools may or may not support federated identities.
- ☐ Yes / provider managed Google Workspace / Cloud Identity accounts (e.g. users@subdomain.providerdomain.com): the provider can provision user accounts in Google Workspace / Cloud Identity to be used by the end-customers. Accounts provisioned this way take up seats in the relevant subscription and are paid accounts, except for 50 accounts in Cloud Identity Free Tier. Supports using Cloud Console / BigQuery Studio, Looker Studio, etc.
- ☐ Yes / provider managed Service Accounts: the provider can create Service Accounts in the provider managed projects and share the service account keys with the users. Alternatively the users can provide the key which the provider uploads to the service account.
- ☐ No: customer provides the Identity, provider assigns IAM roles to give access (see the previous question for details)

Are the customers concerned about the provider:

1. Accessing their data, in the case where they bring their own data in to be combined with the data shared by the provider, or
2. Having visibility into their query activity.

- ☐ Yes - Providers should give read access to the Audit Logging in the provider managed projects to the customers, so they can check the activities related to their project and

data - the provider can always grant access to themselves, as they own the project. Cloud Audit Logging is immutable, append-only, and has a long (400d) retention period.

Is the provider concerned about specific queries the customers are executing against the data?

- ☐ Yes - Provider has access to the audit logs in the provider managed project, but they can also implement centralized logging to streamline anomaly detection and alerting.

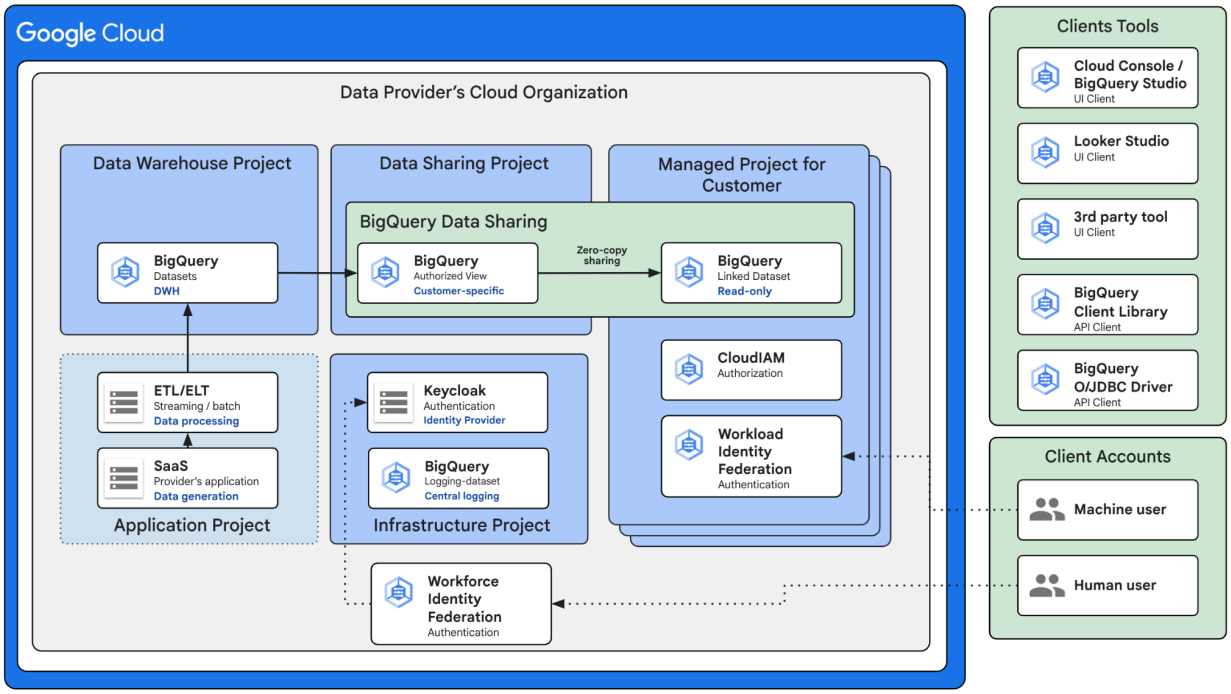
Appendix B: Solution Accelerator for the read-only data sharing use cases

The PMBQP solution accelerator published on GitHub implements one specific scenario of deploying Provider Managed BigQuery Project, which gives read-only access to the data shared to the external users through BigQuery Studio.

Feature highlights of the solution accelerator:

- Create one managed project per customer with only BigQuery being enabled
- Use KeyCloak as an external identity provider to manage customer identities
- Use Authorized Views in customer specific datasets to provide customer-specific views of the customers data
- Use BigQuery Data Sharing (formerly Analytics Hub) to share customer-specific datasets
- Provide only read-only access to the shared data
- Use BigQuery Studio in Cloud Console for human access through a web based UI
- Allow AWS EC2 instances to access the data programmatically through BigQuery API

Architecture diagram



The solution accelerator implements the infrastructure automation aspects only, broken down into different stages.

Solution Accelerator Overview

Stage	High level overview	Manages infrastructure
Stage 0 (init) Shell script	Creates and bootstraps the seed project that stores the Terraform state	Seed project State bucket API enablement on the seed project
Stage 0 (org setup) Terraform	Configures the Cloud Organization	Configure Org Policies - Domain Restricted sharing - Allowed Workload Identity Pools
Stage 1 (bootstrap) Terraform	Creates and configures the folders and projects	Provision folder structure Provision provider projects Configure IAM
Stage 2 (provider infra) Terraform	Provision the infrastructure required for the provider side	Provision VPC network and configure firewall Provision GKE cluster Provision CloudDNS zone Configure Private Google Access Provision LoadBalancer Provision CloudSQL database for keycloak Provision Secret Manager secrets Configure log routing for centralized audit logging from client project's folder Build and publish Keycloak (IdP) image
Stage 3 (identity provider) Shell script	Deploy Keycloak and create an administrative root account using a password saved to Cloud Secret Manager.	Deploy Keycloak to the Kubernetes cluster Wait for Keycloak to be healthy Configure root admin account and store access credentials locally for the terraform keycloak provisioner
Stage 4 (managed identities) Terraform (Keycloak provisioner)	Provision the managed identities within Keycloak	Create the managed identities in Keycloak and store the generated passwords locally (for testing)
Stage 5 (identity federation)	Configure identity federation	Configure Workforce Identity Federation on the Organization level Configure Workload Identity Federation for AWS EC2 on the Project level.
Stage 6 (provider data sharing)	Provision data sharing	Create customer-specific shared dataset

	resources on the provider side.	within the provider's data sharing project Create customer-specific authorized view within the dataset Create customer-specific BigQuery Listing within the provider's data sharing project
Stage 7 (provider managed customer projects)	Provision tenant project for each customer and configure them to be used by the provisioned identities	Create the provider managed projects in the customer projects folder Enable BigQuery APIs only Create the linked dataset by subscribing to the customer-specific listing Grant read-only permissions on bigQuery to the federated identities - Workforce Identity Pool (External human users from Keycloak) - Workload Identity Pool (EC2 instances from AWS)

Additional resources

The detailed documentation is available on GitHub: https://github.com/GoogleCloudPlatform/cpe-partner-solution-accelerators/tree/main/bq-ah-provider-projects

Appendix C: Recommendations for an end-to-send solution in production

The solution accelerator implements one specific scenario (read-only data sharing use cases) and one aspect (Infrastructure Provisioning). In order to build a full end-to-end solution that scales well there are multiple additional steps to consider.

1. **Customer signup and approval process**

   Customers are usually stored in a database along with the features / capabilities enabled for them. As Provider Managed BigQuery Projects may be an upsell to the customer, service enablement and approval status should be tracked in the database as well. In addition to approval status the provisioning status should be tracked (waiting for approval, approved, provisioning, provisioned, suspended, shutdown, etc)

   Once the provisioning is complete the customer should receive a notification (email) with their account details and access information, like the federated Cloud Console login URL.

2. **Provisioning pipeline schedule automation**

   In case a customer's status changes (new customer approved, customer details updated, account shut down because of overspending, etc) the provisioning pipeline should be automatically triggered in order to apply the required changes.

3. **Pipeline modularization for parallel execution**

   In the case of hundreds of projects Terraform may be slow to apply changes across the whole user base, it may be necessary to break up the pipeline into multiple parts that can be executed in parallel. For example there may be a pipeline for each region.

4. **Self-service control panel for the customers**

   Customers should be able to manage their own details, which are not possible for them via Cloud Console. Some examples:

   - Generate new Service Account key
   - Manage users
   - Initiate the loading of external data (in case they can bring their own data)
   - See their current usage vs. their allocated limits

5. **Customer facing notification in case of over usage**

   Billing Account based notifications are primarily meant for providers and they are not necessarily directly consumable by consumers. Consumers should get an actionable notification with clear next steps, for example what happens if they reach their limit, how can they increase their usage limits, who can they contact (support).

6. **Preventive measures in case of over usage**

   Depending on the provider's design decisions automation may be needed to act in case of significant over usage.