Discover and consume data products in a data mesh

We recommend that you design your data mesh to support a wide variety of use cases for data consumption. The most common data consumption use cases in an organization are described in this document. The document also discusses what information data consumers must consider when determining the right data product for their use case, and how they discover and use data products. Understanding these factors can help organizations to ensure that they have the right guidance and tooling in place to support data consumers.

This document is part of a series which describes how to implement a data mesh on Google Cloud. It assumes that you have read and are familiar with the concepts described in Architecture and functions in a data mesh (/architecture/data-mesh) and Build a modern, distributed Data Mesh with Google Cloud (https://services.google.com/fh/files/misc/build-a-modern-distributed-datamesh-with-google-cloud-whitepaper.pdf)
.

The series has the following parts:

- Architecture and functions in a data mesh (/architecture/data-mesh)
- Design a self-service data platform for a data mesh (/architecture/design-self-service-data-platform-data-mesh)
- Describe and organize data products and resources in a data mesh (/architecture/describe-organize-data-products-resources-data-mesh)
- Build data products in a data mesh (/architecture/build-data-products-data-mesh)
- Discover and consume data products in a data mesh (this document)

The design of a data consumption layer, specifically, how the data domain-based consumers use data products, depends on the data consumer requirements. As a prerequisite, it's assumed that consumers have a use case in mind. It's assumed that they have identified the data that they require, and can search the central data product catalog to find it. If that data is not in the catalog or is not in the desired state (for example, if the interface is not appropriate, or the SLAs are insufficient), the consumer must contact the data producer.

Alternatively, the consumer can contact the center of excellence (COE) for the data mesh for advice on which domain is the best suited to produce that data product. The data

consumers can also ask how to make their request. If your organization is large, there should be a process to surface data product requests in a self-service manner.

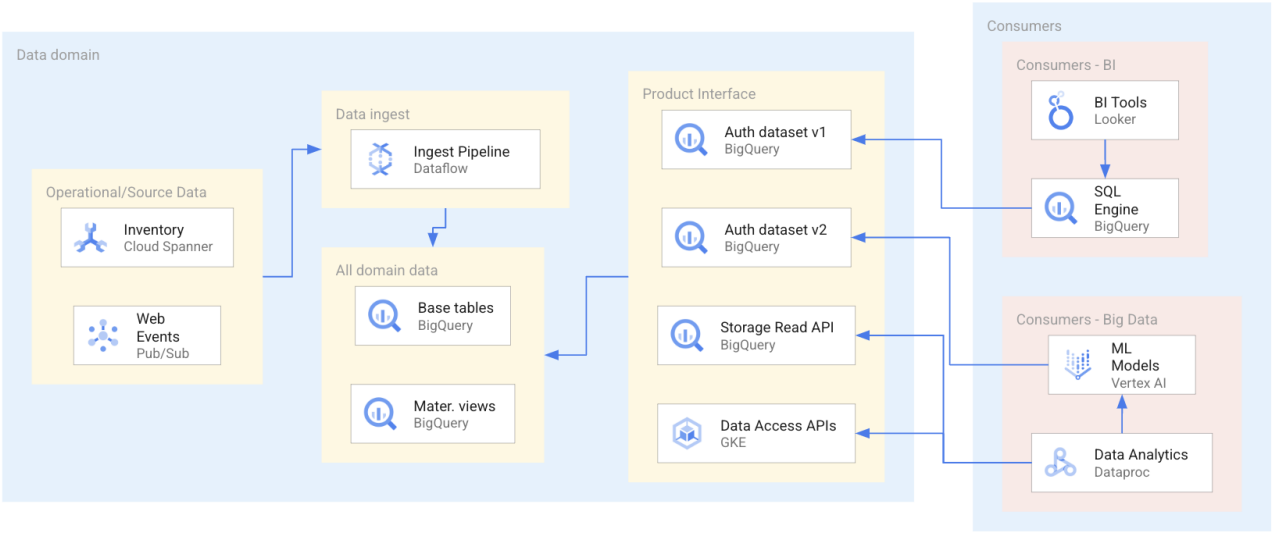
Data consumers use data products through the applications that they run. The type of insights required drives the choice of design of the data-consuming application. When they develop the design of the application, the data consumer also identifies their preferred use of data products in the application. They establish the confidence that they need to have in the trustworthiness and reliability of that data. The data consumers can then establish a view on the data product interfaces and SLAs that the application requires.

Data consumption use cases

For data consumers to create data applications, sources could be one or more data products and, perhaps, the data from the data consumer's own domain. As described in Build data products in a data mesh (/architecture/build-data-products-data-mesh), analytical data products could be made from data products which are based on various physical data repositories.

Although data consumption can happen within the same domain, the most common consumption patterns are those that search for the right data product, regardless of domain, as the source for the application. When the right data product exists in another domain, the consumption pattern requires you to set up the subsequent mechanism for access and usage of the data across domains. The consumption of data products created in domains other than the consuming domain is discussed in Data consumption steps (#data_consumption_steps).

The following diagram shows an example scenario in which consumers use data products through a range of interfaces, including authorized datasets and APIs.



As shown in the preceding diagram, the data producer has exposed four data product interfaces: two BigQuery authorized datasets, a BigQuery dataset exposed by the BigQuery storage read API, and data access APIs hosted on Google Kubernetes Engine. In using the data products, data consumers use a range of applications that query or directly access the data resources within the data products. For this scenario, data consumers access data resources in one of two different ways based on their specific data access requirements. In the first way, Looker uses BigQuery SQL to query an authorized dataset. In the second way, Dataproc directly accesses a dataset through the BigQuery API and then processes that ingested data to train a machine learning (ML) model.

The use of a data consumption application might not always result in a business intelligence (BI) report or a BI dashboard. Consumption of data from a domain can also result in ML models that further enrich analytical products, are used in data analysis, or are a part of operational processes, for example, fraud detection.

Some typical data product consumption use cases are as follows:

- **BI reporting and data analysis:** In this case, data applications are built to consume data from multiple data products. For example, data consumers from the customer relationship management (CRM) team need access to data from multiple domains such as sales, customers, and finance. The CRM application that is developed by these data consumers might need to query both a BigQuery authorized view in one domain and extract data from a Cloud Storage Read API in another domain. For data consumers, the optimizing factors that influence their preferred consumption interface are computing costs and any additional data processing that is required after they query the data product. In BI and data analysis use cases, BigQuery authorized views are likely to be most commonly used.
- **Data science use cases and model training**: In this case, the data consuming team is using the data products from other domains to enrich their own analytical data product such as an ML model. By using Dataproc Serverless for Spark, Google Cloud provides data pre-processing and feature engineering capabilities to enable data enrichment before running ML tasks. The key considerations are availability of sufficient amounts of training data at a reasonable cost, and confidence that the training data is the appropriate data. To keep costs down, the preferred consumption interfaces are likely to be direct read APIs. It's possible for a data consuming team to build an ML model as a data product, and in turn, that data consuming team also becomes a new data producing team.
- **Operator processes:** Consumption is a part of the operational process within the data consuming domain. For example, a data consumer in a team that deals with fraud might be using transaction data coming from operational data sources in the

merchant domain. By using a data integration method like change data capture, this transaction data is intercepted at near real time. You can then use Pub/Sub to define a schema for this data and expose that information as events. In this case, the appropriate interfaces would be data exposed as Pub/Sub topics.

Data consumption steps

Data producers document their data product in the central catalog, including guidance on how to consume the data. For an organization with multiple domains, this documentation approach creates an architecture that's different from the traditional centrally built ELT/ETL pipeline, where processors create outputs without the boundary of business domains. Data consumers in a data mesh must have a well-designed discovery and consumption layer to create a data consumption lifecycle. The layer should include the following:

Step 1: Discover data products through declarative search and exploration of data product specifications: Data consumers are free to search for any data product that data producers have registered in the central catalog. For all data products, the data product tag specifies how to make data access requests and the mode to consume data from the required data product interface. The fields in the data product tags are searchable using a search application. Data product interfaces implement data URIs, which means data does not need to be moved to a separate consumption zone to service consumers. In situations when real-time data isn't needed, consumers query data products and create reports with the results that are generated.

Step 2: Exploring data through interactive data access and prototyping: Data consumers use interactive tools like BigQuery SQL workspace and Jupyter Notebooks to interpret and experiment with the data to refine the queries that they need for production use. Interactive querying enables data consumers to explore newer dimensions of data and improve the correctness of insights generated in production scenarios.

Step 3: Consuming data product through an application, with programmatic access and production:

- **BI reports.** Batch and near-real time reports and dashboards are the most common group of analytic use cases required by data consumers. Reports might require cross-data product access to help facilitate decision making. For example, a customer data platform requires programmatically querying both orders and CRM data products in a scheduled fashion. The results from such an approach provide a holistic customer view to the business users who consume the data.

- **AI/ML model for batch and real-time prediction.** Data scientists use common MLOps principles to build and service ML models that consume data products made available by the data product teams. ML models provide real-time inference capabilities for transactional use-cases like fraud detection. Similarly, with exploratory data analysis, data consumers can enrich source data. For example, exploratory data analysis on sales and marketing campaigns data shows demographic customer segments where sales are expected to be highest and hence where campaigns should be run.

What's next

- See a reference implementation of the data mesh architecture (https://github.com/GoogleCloudPlatform/data-mesh-demo).
- Learn more about BigQuery (/bigquery).
- Read more about Vertex AI (/vertex-ai).
- Learn about data science on Dataproc (/dataproc#section-8).
- For more reference architectures, diagrams, tutorials, and best practices, explore the Cloud Architecture Center (/architecture).

Last updated 2022-10-06 UTC.