What is Cloud Data Fusion?

Cloud Data Fusion is a fully managed, cloud-native, enterprise data integration service for quickly building and managing data pipelines.

The Cloud Data Fusion web UI lets you to build scalable data integration solutions to clean, prepare, blend, transfer, and transform data, without having to manage the infrastructure.

Cloud Data Fusion is powered by the open source project CDAP (https://cdap.io/). Throughout this page, there are links to the CDAP documentation site, where you can find more detailed information.

Interfaces

To use Cloud Data Fusion, you can use the visual web UI or command-line tools.

Using the code-free web UI

When using Cloud Data Fusion, you use both the Google Cloud console and the separate Cloud Data Fusion web UI.

- In the Google Cloud Console, you create a Google Cloud project, create and delete Cloud Data Fusion instances (unique deployments of Cloud Data Fusion), and view Cloud Data Fusion instance details.
- In the Cloud Data Fusion UI, you use the various pages, such as **Pipeline Studio** or **Wrangler**, to visually design data pipelines and use Cloud Data Fusion functionality.

At a high level, you do the following:

1. Create a Cloud Data Fusion instance (/data-fusion/docs/how-to/create-instance) in the Google Cloud Console.
2. Find your Cloud Data Fusion instance in the Google Cloud console Instances page (https://console.cloud.google.com/data-fusion/locations/-/instances), and click the **View instance** link in the **Action** column. This opens the Cloud Data Fusion UI in a new browser tab.
3. Use the various pages in the Cloud Data Fusion web UI to visually design your pipelines and manage metadata.

Using command-line tools

Alternatively to the web UI, you can use command-line tools to create and manage your Cloud Data Fusion instances and pipelines.

- The REST reference (/data-fusion/docs/reference/rest) describes the API for creating and managing your Cloud Data Fusion instances on Google Cloud.
- The CDAP reference (/data-fusion/docs/reference/cdap-reference) describes the REST API for creating and managing pipelines and datasets.

Core concepts

This section provides an introduction to some of the core concepts of Cloud Data Fusion. Some sections provide links to the CDAP documentation, where you can learn more about each concept and in more detail.

Cloud Data Fusion instance

A Cloud Data Fusion *instance* is a unique deployment of Cloud Data Fusion. To get started with Cloud Data Fusion, you create a Cloud Data Fusion instance through the Google Cloud console.

You can create multiple instances in a single Google Cloud console project and can specify the Google Cloud region to create your Cloud Data Fusion instances in.

Based on your requirements and cost constraints, you can create a Developer, Basic, or Enterprise (/data-fusion/pricing#comparison_of_basic_enterprise_and_developer_editions) instance.

Each Cloud Data Fusion instance contains a unique, independent Cloud Data Fusion deployment that contains a set of services, that handle pipeline lifecycle management, orchestration, coordination, and metadata management. These services run using long-running resources in a tenant project (/service-infrastructure/docs/glossary#tenant).

Execution environment

Cloud Data Fusion creates ephemeral execution environments to run pipelines when you manually run your pipelines or when pipelines run through a time schedule or a pipeline state trigger. Cloud Data Fusion supports Dataproc (/dataproc) as an execution environment, in which you can choose to run pipelines as MapReduce, Spark, or Spark Streaming programs. Cloud Data Fusion provisions an ephemeral Dataproc cluster in your customer project at the beginning of a pipeline run, executes the pipeline using MapReduce or Spark in the cluster, and then deletes the cluster after the pipeline execution is complete.

Alternatively, if you manage your Dataproc clusters in controlled environments, through technologies like Terraform, you can also configure Cloud Data Fusion not to provision clusters. In such environments, you can run pipelines against existing Dataproc clusters.

Autoscaling

You can use the predefined Cloud Data Fusion autoscaling policy or your own policy to automate cluster resource management for processing.

For information about creating your own policy to increase cluster workers to meet workload demands, see Autoscaling clusters (/dataproc/docs/concepts/configuring-clusters/autoscaling).

For information about using the predefined autoscaling policy for pipelines running in Cloud Data Fusion 6.6 and later, see When to use autoscaling (/data-fusion/docs/concepts/configure-clusters#autoscaling).

Pipeline

A *pipeline* is a way to visually design data and control flows to extract, transform, blend, aggregate, and load data from various on-premises and cloud data sources. Building pipelines lets you to create complex data processing workflows that can help you solve data ingestion, integration, and migration problems. You can use Cloud Data Fusion to build both batch and real-time pipelines, depending on your needs.

Pipelines enable you to express your data processing workflows using the logical flow of data, while Cloud Data Fusion handles all the functionality that is required to physically run in an execution environment. The Cloud Data Fusion planner transforms the logical flow into parallel computations, using Apache Spark and Apache Hadoop MapReduce on Dataproc.

Pipeline node

In the **Studio** page of the Cloud Data Fusion UI, pipelines are represented as a series of *nodes* arranged in a directed acyclic graph (DAG), forming a one-way flow. Nodes represent the various actions that you can take with your pipelines, such as reading from sources, performing data transformations, and writing output to sinks. You can develop data pipelines in the Cloud Data Fusion UI by connecting together sources, transformations, sinks, and other nodes.

By providing access to logs and metrics, pipelines offer a simple way for administrators to operationalize their data processing workflows without the need for custom tooling.

Learn more about pipelines (https://cdap.atlassian.net/wiki/spaces/DOCS/pages/517406816) on the CDAP documentation site.

Replication job

Replication lets you to replicate your data continuously and in real time from Operational Datastores, such as SQL Server and MySQL into BigQuery.

For more information, see the Replication job (/data-fusion/docs/concepts/replication) page.

Triggering

You can create a *trigger* on a data pipeline (called the upstream pipeline), to have it run at the completion of one or more different pipelines (called downstream pipelines). You choose when the downstream pipeline runs - upon the success, failure, stop, or any combination thereof, of the upstream pipeline run.

Triggers are useful for:

- Cleansing your data once and making it available to multiple downstream pipelines for consumption.
- Sharing information, such as runtime arguments and plugin configurations, between pipelines. This is called *payload configuration*.
- Having a set of dynamic pipelines that can run using the data of the hour/day/week/month, as opposed to a static pipeline that must be updated for every run.

Plugin

A *plugin* is a customizable module that can be used to extend the capabilities of Cloud Data Fusion. Cloud Data Fusion provides plugins for sources, transforms, aggregates, sinks, error collectors, alert publishers, actions, and post-run actions.

A plugin is sometimes referred to as a *node*, usually in the context of the Cloud Data Fusion web UI.

The following table describes the various categories of plugins available in Cloud Data Fusion.

Category	Description
Sources	**Sources** are connectors to databases, files, or real-time streams from which you obtain your data. They enable you to ingest data, using a simple UI, so you don't have to worry about coding low-level connections.
Transforms	**Transforms** let you to manipulate data after the data is ingested. For example, you can clone a record, format JSON, and even create custom transforms using the JavaScript plugin.
Analytics	**Analytics** plugins are used to perform aggregations such as grouping and joining data from different sources, as well as running analytics and machine learning operations. Cloud Data Fusion provides built-in plugins for various such use cases.
Actions	**Action** plugins define custom actions that are scheduled to take place during a workflow but don't directly manipulate data in the workflow. For example, using the Database custom action, you can run an arbitrary database command at the end of your pipeline. Alternatively, you can trigger an action to move files within Cloud Storage.
Sinks	Data must be written to a *sink*. Cloud Data Fusion contains various sinks, such as Cloud Storage, BigQuery, Spanner, relational databases, file systems, mainframes.
Error collectors	When nodes encounter null values, logical errors, or other sources of errors, you can use an *error collector* plugin to catch errors. You can connect this plugin to the output of any transform or analytics plugin, and it will catch errors that match a condition you define. You can then process these errors in a separate error processing flow in your pipeline.
Alert publishers	**Alert Publisher** plugins let you to publish notifications when uncommon events occur. Downstream processes can then subscribe to these notifications to trigger custom processing for these alerts.
Conditionals	Pipelines offer control flow plugins in the form of *conditionals*. Conditional plugins let you to branch your pipeline into two separate paths, depending on whether the specified condition predicate evaluates to true or false.

If you need a plugin that isn't provided, you can develop a custom plugin (https://cdap.atlassian.net/wiki/spaces/DOCS/pages/480412201) yourself.

WARNING: Installing an untrusted plugin is not recommended, as it might present a security risk.

Compute profile

A *compute profile* specifies how and where a pipeline is executed. A profile encapsulates any information required to set up and delete the physical execution environment of a pipeline. For example, a profile includes the type of cloud provider (such as Google Cloud), the service to use on the cloud provider (such as Dataproc), credentials, resources (memory and CPU), image, minimum and maximum node count, and other values.

A profile is identified by name and must be assigned a provisioner and its related configuration. A profile can exist either at the Cloud Data Fusion instance level or at the namespace level.

Learn more about profiles (https://cdap.atlassian.net/wiki/spaces/DOCS/pages/480314003) on the CDAP documentation site.

Features

Category	Features
Development	• Graphical pipeline designer • 100+ plugins - connectors, transforms & actions • Code-free visual transformations • 1000+ built-in transforms • Data quality libraries • Developer SDK
Testing	• Visual pipeline debugging • Testing framework
Execution	• Dataproc - batch (Apache Spark, Apache Hadoop MapReduce) and real time (Spark Streaming) • Control flow and data flow in pipelines
Operations	• REST API • Schedules and triggers • Monitoring dashboards

Category	Features
Integration metadata	• Automatic technical and operational metadata capture • Business metadata annotations • Search datasets by keywords and schema • Dataset and field-level lineage for traceability
Extensibility	• Custom plugins • Configurable plugin UI widgets • Custom provisioners • Custom compute profiles
Reusability	• Pipeline and plugin templates • Runtime arguments and preferences • Hub for distributing reusable plugins, pipelines, and solutions
Google Cloud Integrations	• GKE - instance deployment • Dataproc - pipeline execution (batch and real time) • Cloud KMS - secure data storage • Cloud SQL and Cloud Storage - entity and artifact metadata storage • Persistent Disk - Logs and metrics storage • Google Cloud console - instance lifecycle management • Google Cloud's operations suite (audit logs only)
Connectors (Google Cloud)	• Cloud Storage • BigQuery • Cloud SQL • Pub/Sub • Spanner • Bigtable • Datastore

Category	Features
Connectors (non-Google Cloud)	• Public cloud services • File systems • Relational DBs • NoSQL stores • Mainframes and other legacy systems
Transformations	• Code-free transformations for cleansing, blending, harmonizing, and mapping • Interactive transformations with feedback • Code based (in-browser) transforms - Scala (Apache Spark), Python, and JavaScript • Existing Spark and MapReduce jobs
Analytics	• Aggregations • Joins • Group By

What's next

- See Cloud Data Fusion common use cases (/data-fusion#use-cases).
- See Cloud Data Fusion pricing (/data-fusion/pricing).
- Create a Cloud Data Fusion instance (/data-fusion/docs/how-to/create-instance).
- Work through a tutorial (/data-fusion/docs/tutorials).

Last updated 2022-09-28 UTC.