# Using Dataflow SQL

The page explains how to use Dataflow SQL and create Dataflow SQL jobs.

To create a Dataflow SQL job, write (#writing-queries) and run (#running-queries) a Dataflow SQL query.

## Using the Dataflow SQL UI

The Dataflow SQL UI is a BigQuery web UI setting for creating Dataflow SQL jobs.

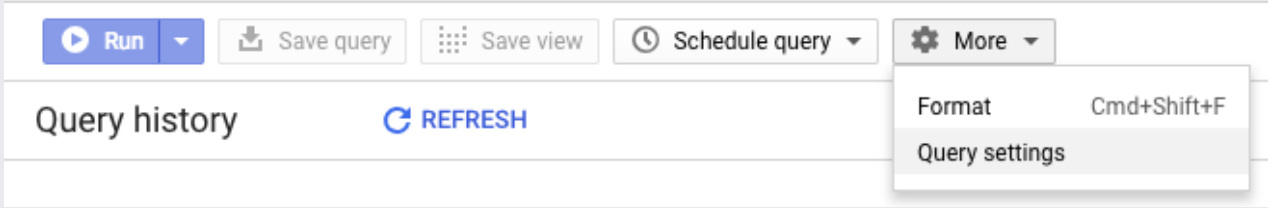
You can access the Dataflow SQL UI from the BigQuery web UI.

1. Go to the BigQuery web UI.

   Go to the BigQuery web UI (https://console.cloud.google.com/bigquery)

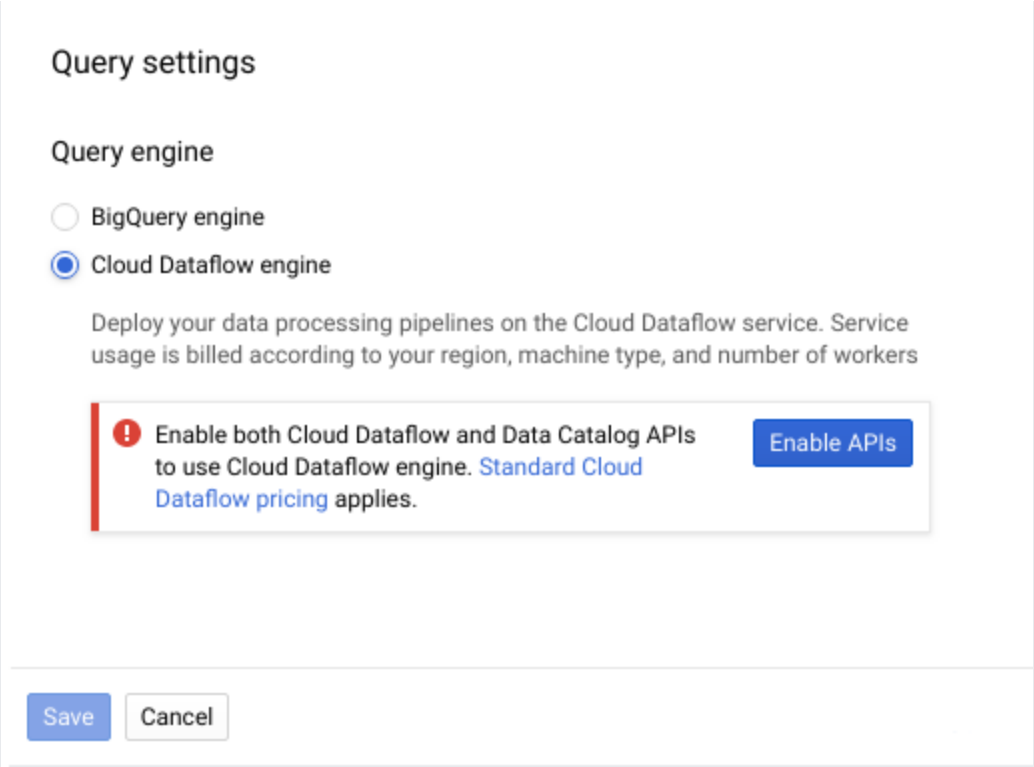
2. Switch to the Cloud Dataflow engine.

   a. Click the **More** drop-down menu and select **Query settings**.

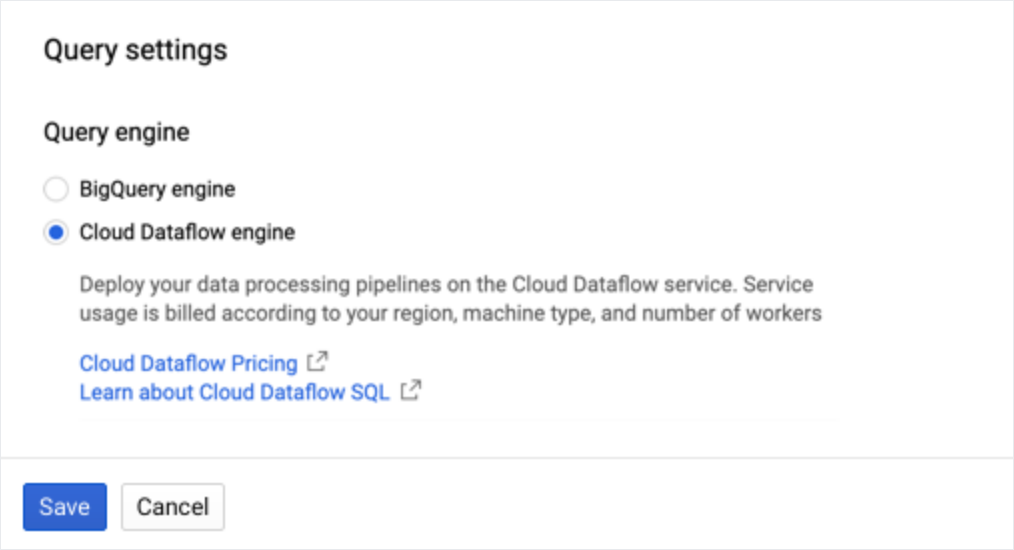



   a. In the **Query settings** menu, select **Dataflow engine**.

   a. In the prompt that appears if the Dataflow and Data Catalog APIs are not enabled, click **Enable APIs**.




a. Click **Save**.




The pricing for the Cloud Dataflow engine is different than the pricing for the BigQuery engine. For details, see g (#pricing).

You can also access the Dataflow SQL UI from the Dataflow monitoring interface (/dataflow/docs/guides/using-monitoring-intf).

1. Go to the Dataflow monitoring interface.

Go to the Dataflow monitoring interface (https://console.cloud.google.com/dataflow)

2. Click **Create job from SQL**.

## Writing Dataflow SQL queries

Dataflow SQL queries use the Dataflow SQL query syntax (/dataflow/docs/reference/sql/query-syntax). The Dataflow SQL query syntax is similar to BigQuery standard SQL (/bigquery/docs/reference/standard-sql/query-syntax).

You can use the Dataflow SQL streaming extensions (/dataflow/docs/reference/sql/streaming-extensions) to aggregate data from continuously updating Dataflow sources like Pub/Sub.

For example, the following query counts the passengers in a Pub/Sub stream of taxi rides every minute:

```
T
BLE_START('INTERVAL 1 MINUTE') as period_start,
(passenger_count) AS pickup_count
pubsub.topic.`pubsub-public-data`.`taxirides-realtime`

e_status = "pickup"
 BY
BLE(event_timestamp, 'INTERVAL 1 MINUTE')
```

## Running Dataflow SQL queries

When you run a Dataflow SQL query, Dataflow turns the query into an Apache Beam pipeline (/dataflow/docs/concepts/beam-programming-model) and executes the pipeline.

You can run a Dataflow SQL query using the Cloud Console or `gcloud` command-line tool.

Consolegcloud (#gcloud)

To run a Dataflow SQL query, use the Dataflow SQL UI.

1. Go to the Dataflow SQL UI.

   Go to the Dataflow SQL UI (https://console.cloud.google.com/bigquery?qe=df)

2. Enter the Dataflow SQL query into the query editor.
3. Click **Create Cloud Dataflow job** to open a panel of job options.
4. (Optional) Click **Show optional parameters** and set Dataflow pipeline options (#setting_pipeline_options).
5. In the **Destination** section of the panel, select an **Output type**.
6. Click **Create**.

**Note:** Starting a Dataflow SQL job might take several minutes. You cannot update a Dataflow SQL job after creating it.

For more information about querying data and writing Dataflow SQL query results, see Using data sources and destinations (/dataflow/docs/guides/sql/data-sources-destinations).

## Setting pipeline options

You can set Dataflow pipeline options for Dataflow SQL jobs. Dataflow pipeline options are execution parameters (/dataflow/docs/guides/specifying-exec-params) that configure how and where to run Dataflow SQL queries.

To set Dataflow pipeline options for Dataflow SQL jobs, specify the following parameters when you run a Dataflow SQL query (#running-query).

Consolegcloud (#gcloud)

| Parameter | Type | Description | I |
|---|---|---|---|
| **Regional endpoint** | `String` | The region to run the query in. Dataflow SQL queries can be run in regions that have a Dataflow regional endpoint. (/dataflow/docs/concepts/regional-endpoints) | I c u c |
| **Max workers** | `int` | The maximum number of Compute Engine instances available to | I |

| | | | |
|---|---|---|---|
| | | your pipeline during execution. | |
| **Worker region** | `String` | The Compute Engine region (/compute/docs/regions-zones/regions-zones) for launching worker instances to run your pipeline. The Compute Engine worker region can be in a different region than the Dataflow regional endpoint. | |
| **Worker zone** | `String` | The Compute Engine zone (/compute/docs/regions-zones/regions-zones) for launching worker instances to run your pipeline. The Compute Engine zone can be in a different region than the Dataflow regional endpoint. | |
| **Service account email** | `String` | The email address of the controller service account (/dataflow/docs/concepts/security-and-permissions#controller_service_account) with which to run the pipeline. The email address must be in the form `my-service-account-name@<project-id>.iam.gserviceaccount.com`. | |
| **Machine type** | `String` | The Compute Engine machine type | |

| | | | |
|---|---|---|---|
| | | (/compute/docs/machine-types) that Dataflow uses when starting workers. You can use any of the available Compute Engine machine type families as well as custom machine types.<br><br>For best results, use `n1` machine types. Shared core machine types, such as `f1` and `g1` series workers, are not supported under the Dataflow Service Level Agreement (/dataflow/sla).<br><br>Note that Dataflow bills by the number of vCPUs and GB of memory in workers. Billing is independent of the machine type family. | |
| **Additional experiments** | `String` | The experiments to enable. An experiment can be a value, like `enable_streaming_engine`, or a key-value pair, such as `shuffle_mode=service`. The experiments must be in a comma-separated list. | |
| **Worker IP Address Configuration** | `String` | Specifies whether Dataflow workers use public IP addresses (/dataflow/docs/guides/specifying-networks#public_ip_parameter)<br>.<br><br>If the value is set to `Private`, Dataflow workers use private IP addresses for all communication. The specified `Network` or `Subnetwork` must have Private Google Access (/vpc/docs/configure-private-google-access#configuring_access_to_google_services_from_internal_ips) enabled.<br><br>If the value is set to `Private` and the `Subnetwork` option is specified, the `Network` option is ignored. | |
| **Network** | `String` | The Compute Engine network (/vpc/docs/vpc) to which workers are assigned. | |
| **Subnetwork** | `String` | The Compute Engine subnetwork (/vpc/docs/vpc#vpc_networks_and_subnets) to which workers are assigned. The subnetwork must be in the form `regions/`***`region`***`/subnetworks/`***`subnetwork`***. | |

Dataflow SQL jobs use autoscaling and Dataflow automatically chooses the execution mode (batch or stream
annot control this behavior for Dataflow SQL jobs.

## Stopping Dataflow SQL jobs

To stop Dataflow SQL jobs, use the Cancel command (/dataflow/docs/guides/stopping-a-pipeline). Stopping a Dataflow SQL job with Drain is not supported.

## Pricing

Dataflow SQL uses the standard Dataflow pricing; it does not have separate pricing. You are billed for the resources consumed by the Dataflow jobs that you create based on your SQL statements. The charges for these resource are the standard Dataflow charges for vCPU, memory, Persistent Disk, Streaming Engine, and Dataflow Shuffle.

A Dataflow SQL job might consume additional resources such as Pub/Sub and BigQuery, each billed at their own pricing.

For more information about Dataflow pricing, see the Dataflow pricing page (/dataflow/pricing).

## What's next

- Walk through the Joining streaming data with Dataflow SQL (/dataflow/docs/samples/join-streaming-data-with-sql) tutorial.
- Read about using data sources and destinations (/dataflow/docs/guides/sql/data-sources-destinations).
- Explore the `gcloud` command-line tool for Dataflow SQL (/sdk/gcloud/reference/dataflow/sql/query).

Last updated 2020-06-26 UTC.