BEAM

COLLEGE

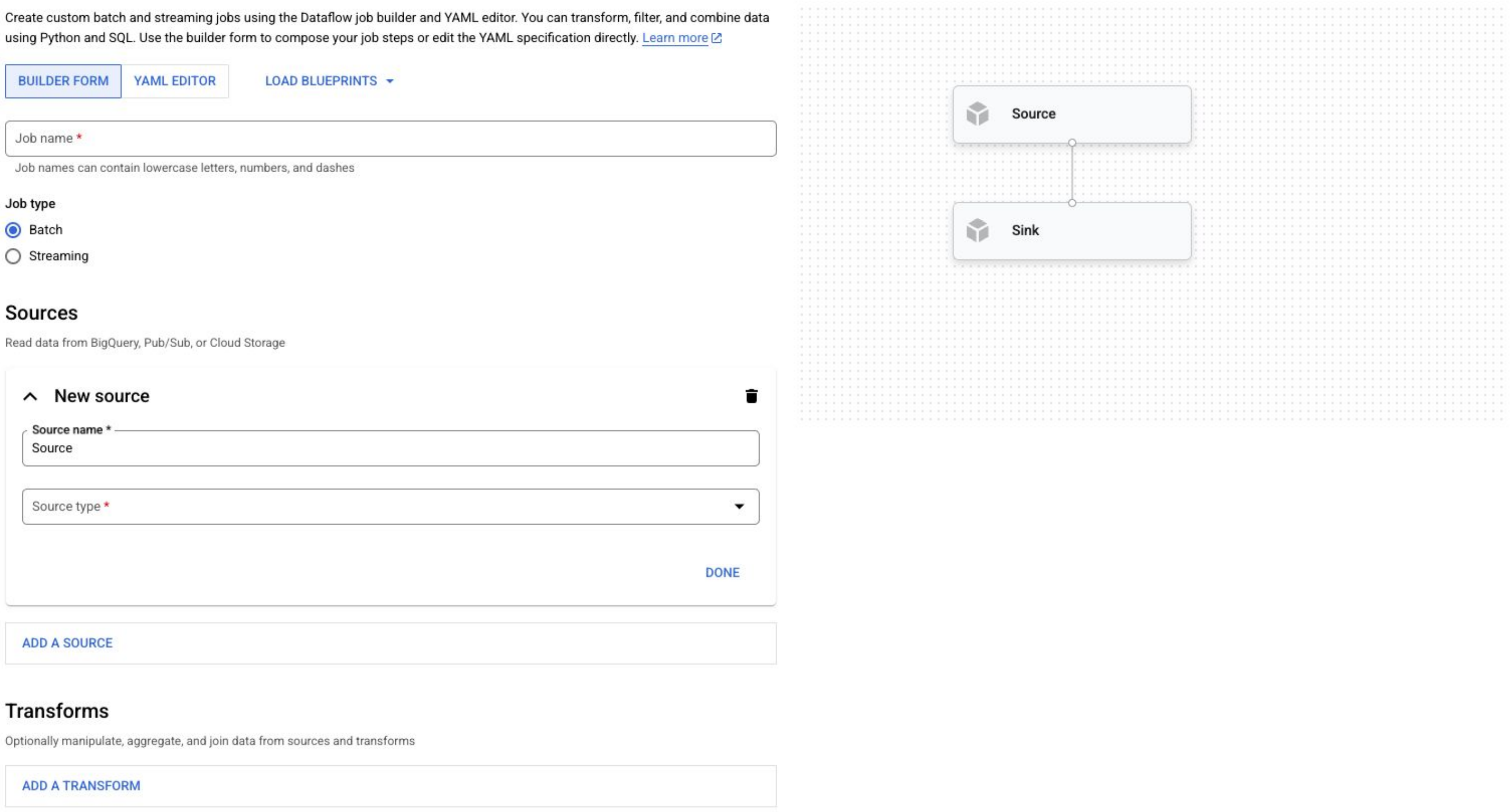
The Dataflow Job Builder

Ryan Madden

Agenda

- The Job Builder & Beam YAML
- Why use the Job Builder?
- Tour of the Job Builder
- Working with YAML

What is the Dataflow Job Builder?



Create custom batch and streaming jobs using the Dataflow job builder and YAML editor. You can transform, filter, and combine data using Python and SQL. Use the builder form to compose your job steps or edit the YAML specification directly. Learn more

Job name *

Job names can contain lowercase letters, numbers, and dashes

Job type

● Batch

○ Streaming

Sources

Read data from BigQuery, Pub/Sub, or Cloud Storage

New source

Source name *: Source

Source type *

Transforms

Optionally manipulate, aggregate, and join data from sources and transforms

What is Beam YAML?

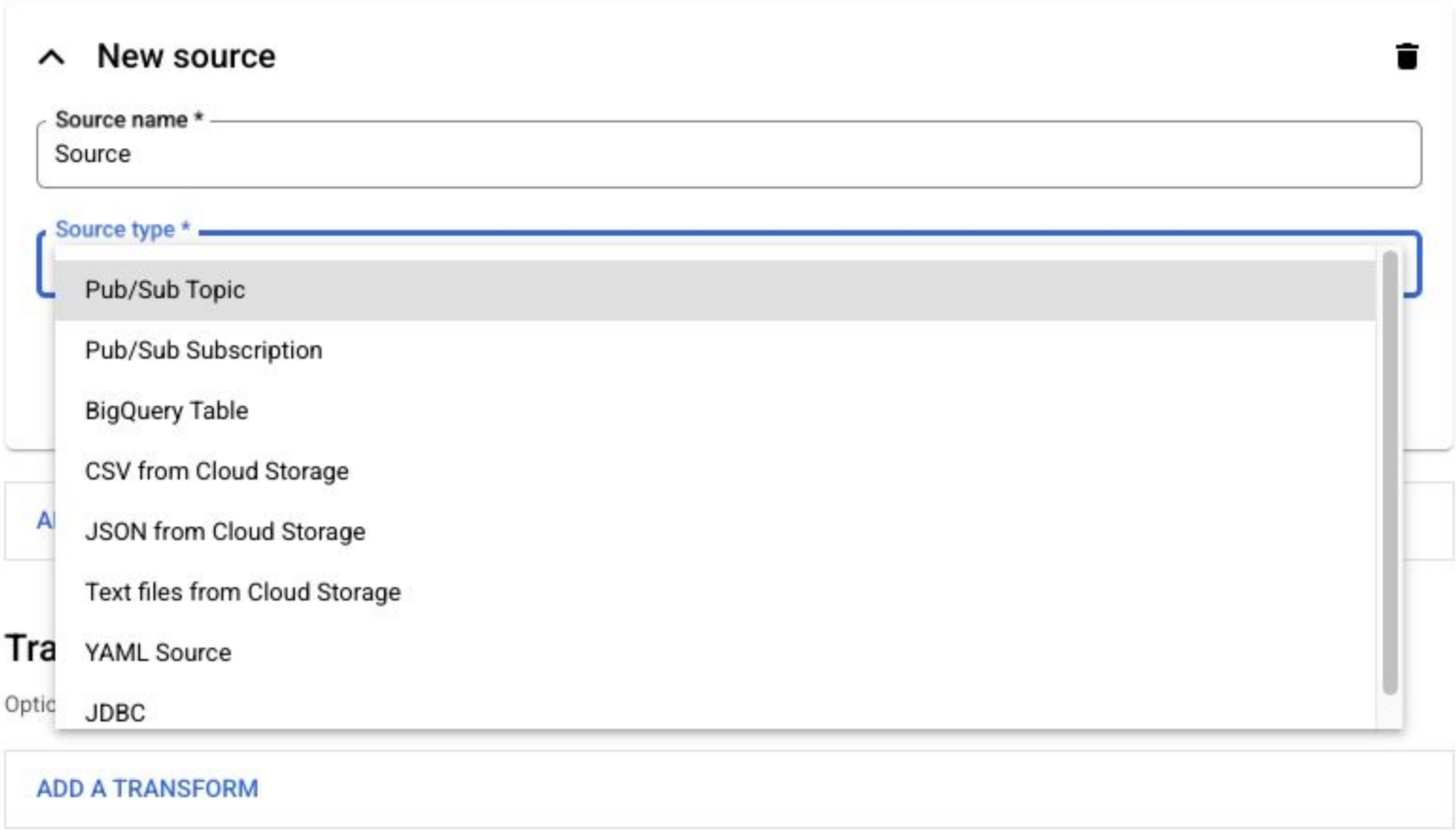
Write:

```
pipeline:
  transforms:
    - type: Create
      config:
        elements: [1, 2, 3]
    - type: LogForTesting
        input: Create
```

Run:

```
python -m apache_beam.yaml.main --yaml_pipeline_file=pipeline.yaml
```

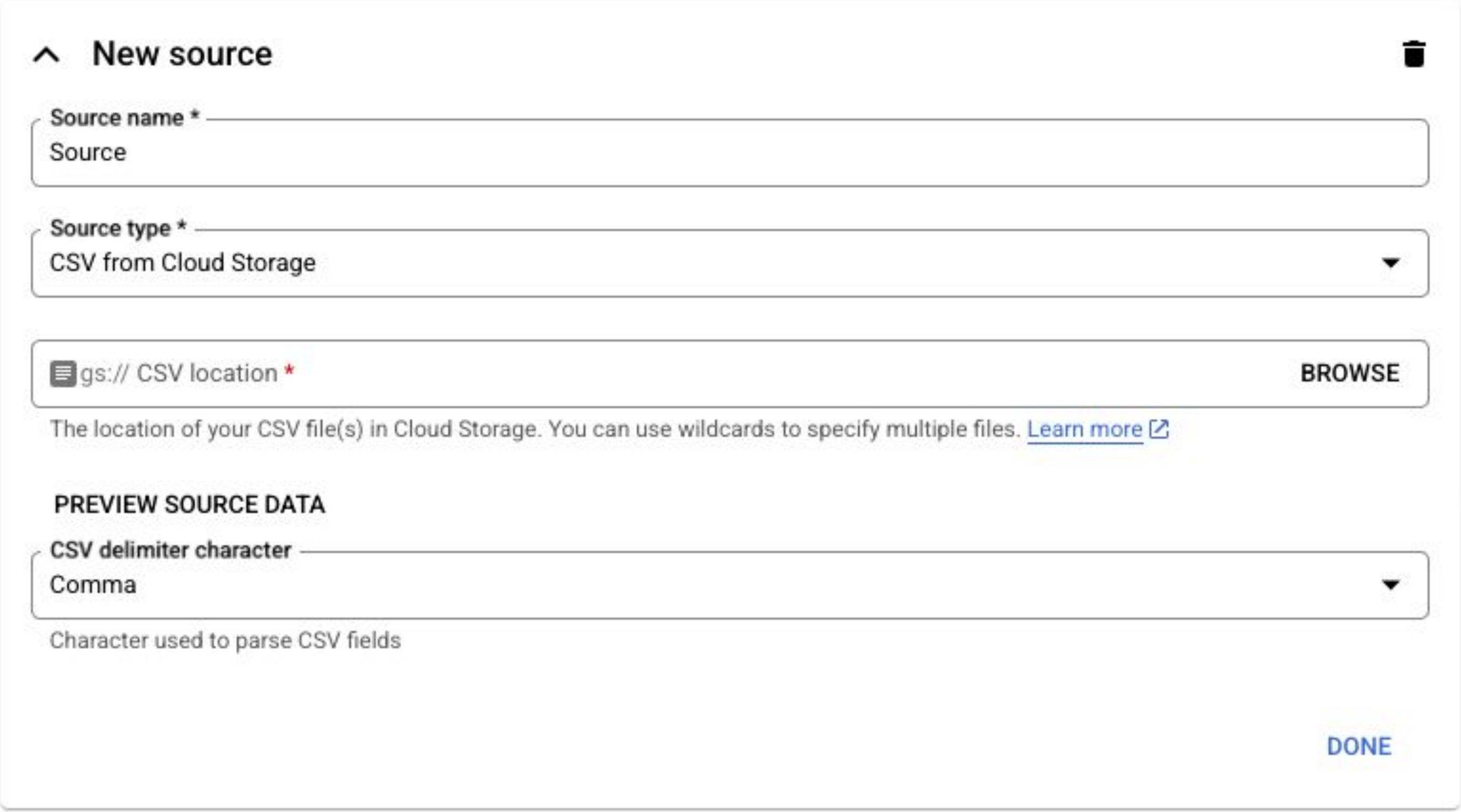
Why use the Dataflow Job Builder?



New source

Source name *: Source

Source type *



New source

Source name *
Source

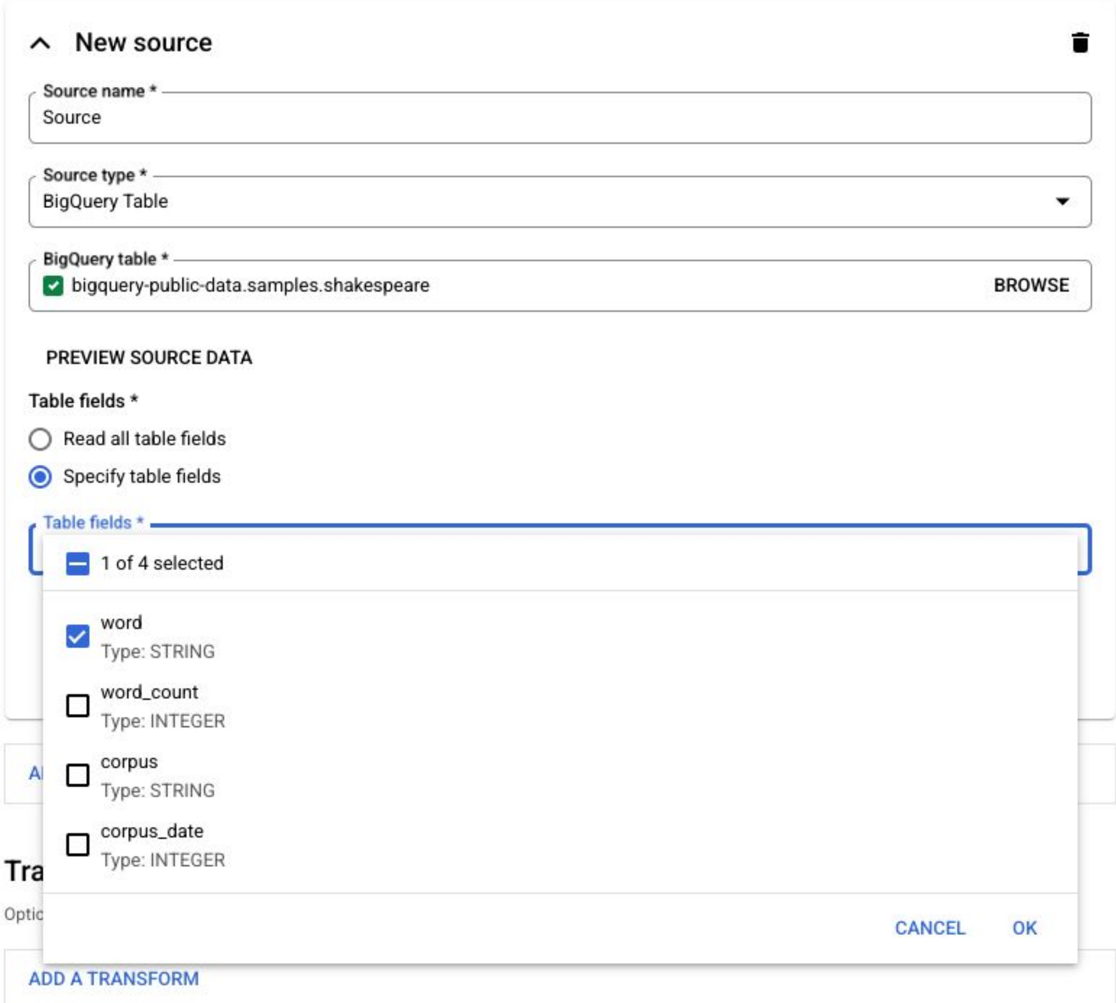
Source type *
CSV from Cloud Storage

gs:// CSV location *

The location of your CSV file(s) in Cloud Storage. You can use wildcards to specify multiple files. Learn more

CSV delimiter character
Comma

Character used to parse CSV fields



New source

Source name *
Source

Source type *
BigQuery Table

BigQuery table *
☑ bigquery-public-data.samples.shakespeare

Table fields *
○ Read all table fields
● Specify table fields

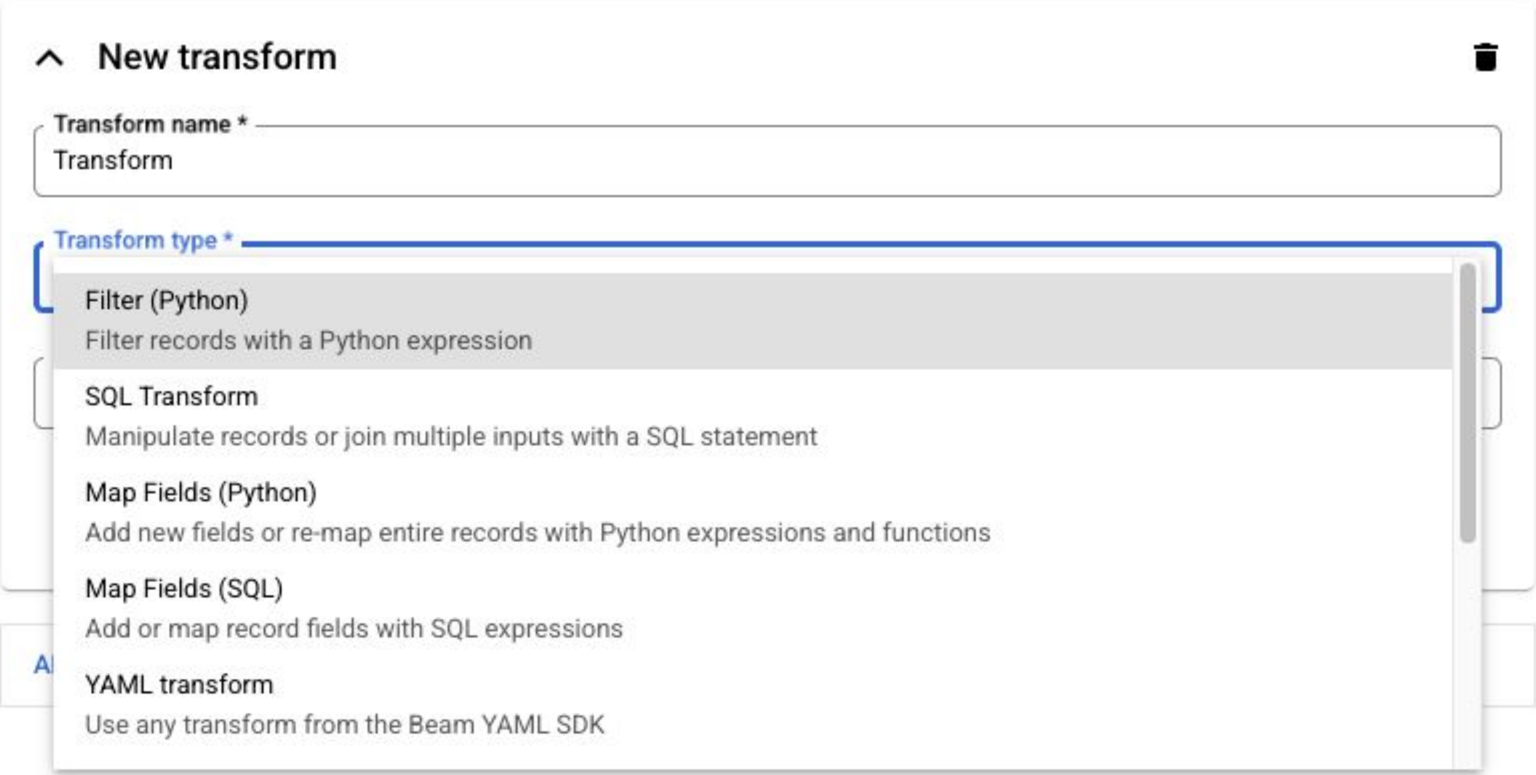
Table fields *
▣ 1 of 4 selected

☑ word
Type: STRING

□ word_count
Type: INTEGER

□ corpus
Type: STRING

□ corpus_date
Type: INTEGER

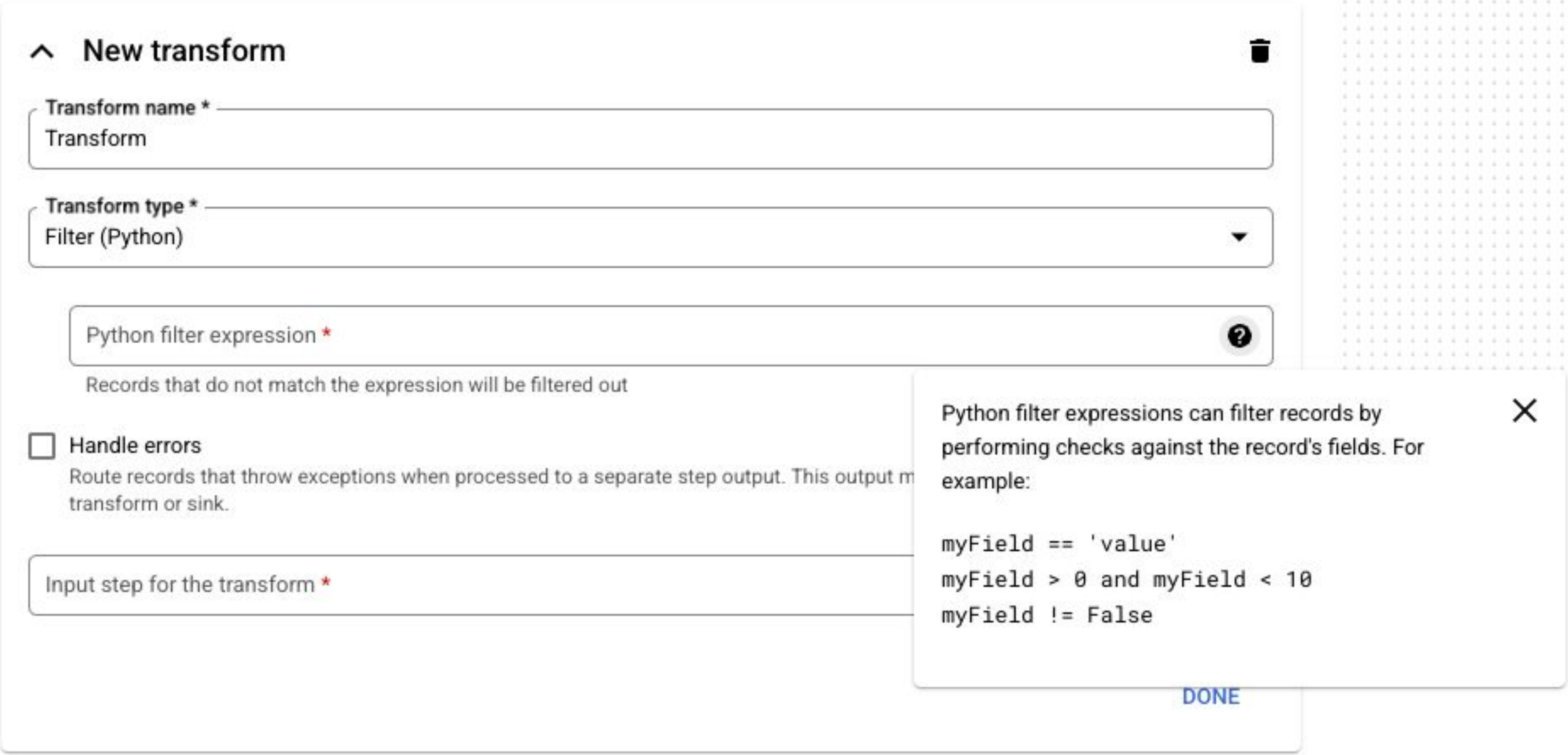


New transform

Transform name *
Transform

Transform type *

Filter (Python)
Filter records with a Python expression
SQL Transform
Manipulate records or join multiple inputs with a SQL statement
Map Fields (Python)
Add new fields or re-map entire records with Python expressions and functions
Map Fields (SQL)
Add or map record fields with SQL expressions
YAML transform
Use any transform from the Beam YAML SDK
Al



New transform

Transform name *

Transform

Transform type *

Filter (Python)

Python filter expression *

Records that do not match the expression will be filtered out

□ Handle errors

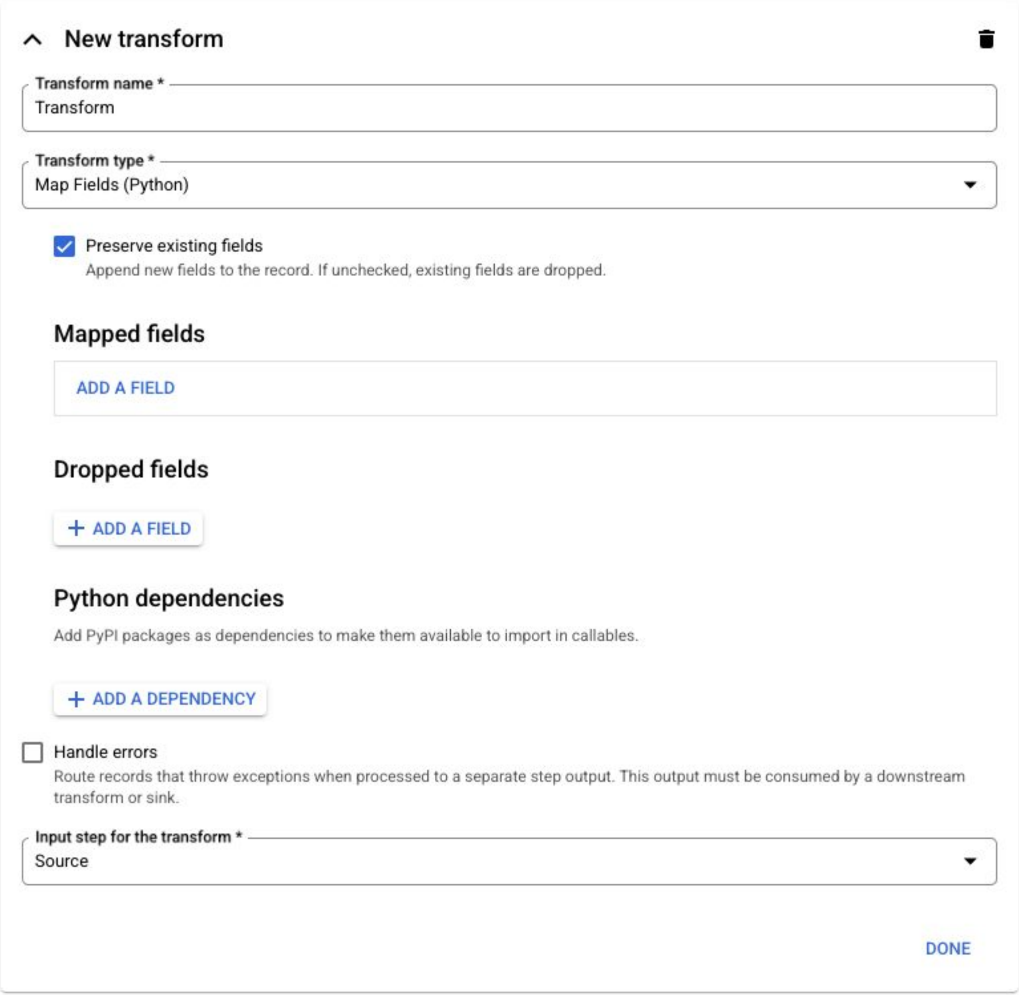
Route records that throw exceptions when processed to a separate step output. This output m

transform or sink.

Input step for the transform *

Python filter expressions can filter records by performing checks against the record's fields. For example:

```
myField == 'value'
myField > 0 and myField < 10
myField != False
```



New transform

Transform name *
Transform

Transform type *
Map Fields (Python)

☑ Preserve existing fields
Append new fields to the record. If unchecked, existing fields are dropped.

Mapped fields

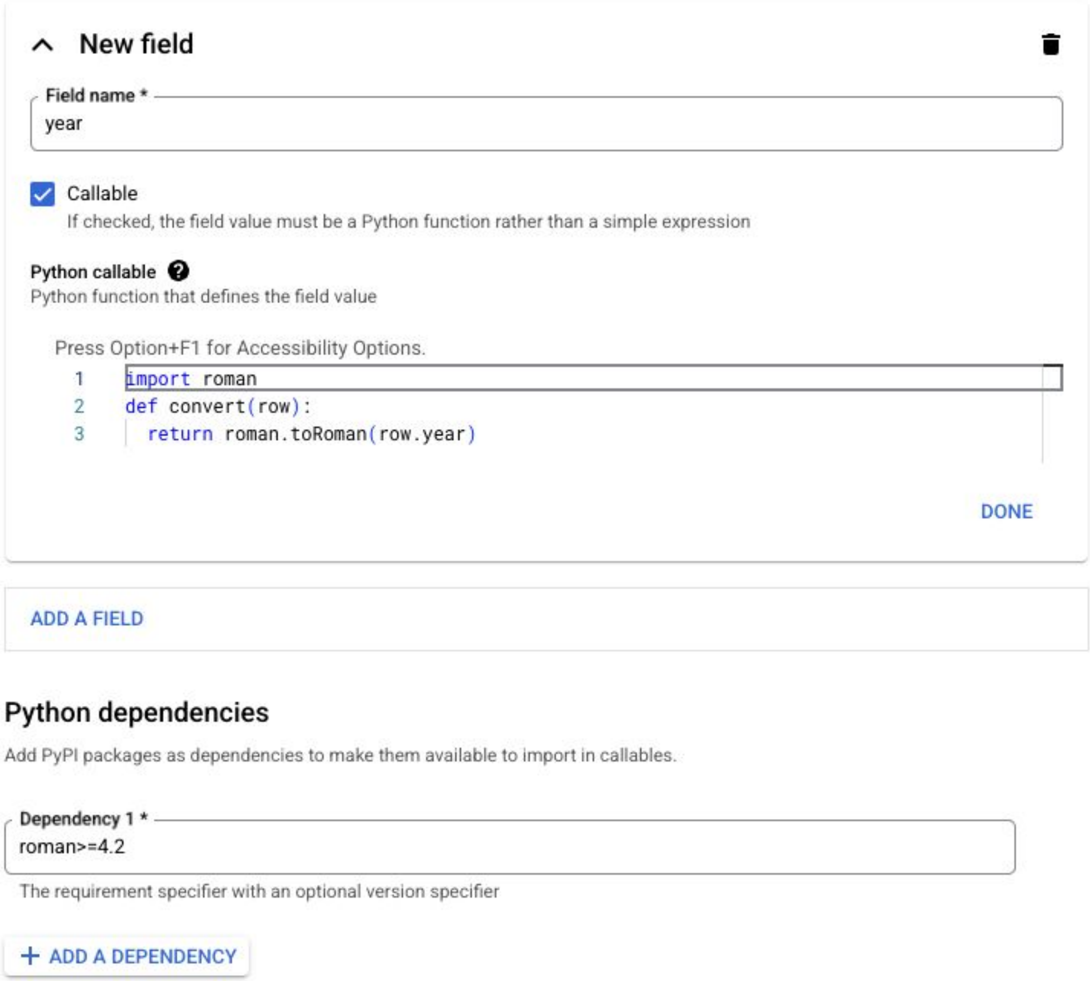
Dropped fields

Python dependencies

Add PyPI packages as dependencies to make them available to import in callables.

□ Handle errors
Route records that throw exceptions when processed to a separate step output. This output must be consumed by a downstream transform or sink.

Input step for the transform *
Source



Field name *: year

☑ Callable

If checked, the field value must be a Python function rather than a simple expression

Python callable

Python function that defines the field value

Press Option+F1 for Accessibility Options.

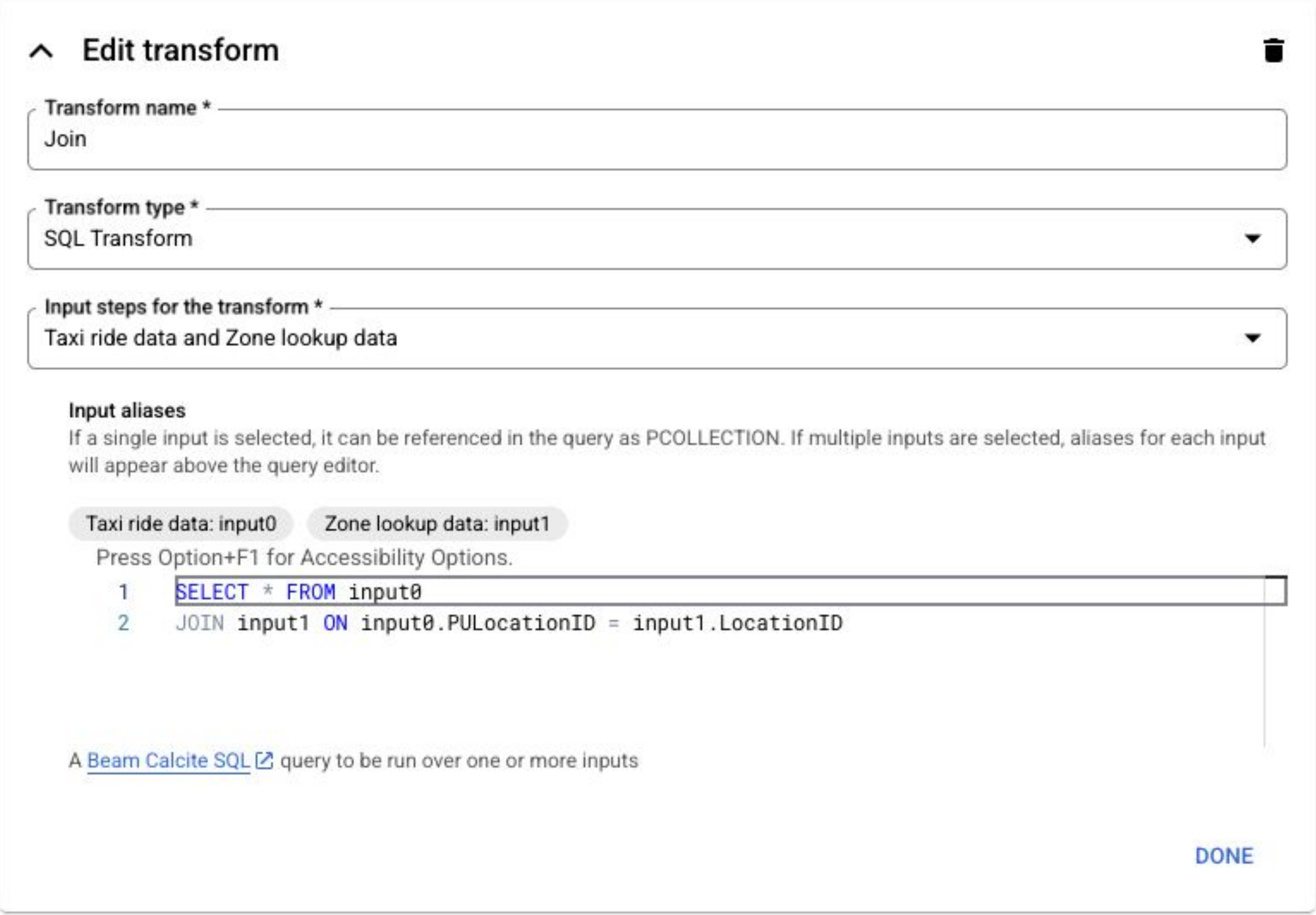
```
import roman
def convert(row):
  return roman.toRoman(row.year)
```

Python dependencies

Add PyPI packages as dependencies to make them available to import in callables.

Dependency 1 *: roman>=4.2

The requirement specifier with an optional version specifier



Edit transform

Transform name *
Join

Transform type *
SQL Transform

Input steps for the transform *
Taxi ride data and Zone lookup data

Input aliases

If a single input is selected, it can be referenced in the query as PCOLLECTION. If multiple inputs are selected, aliases for each input will appear above the query editor.

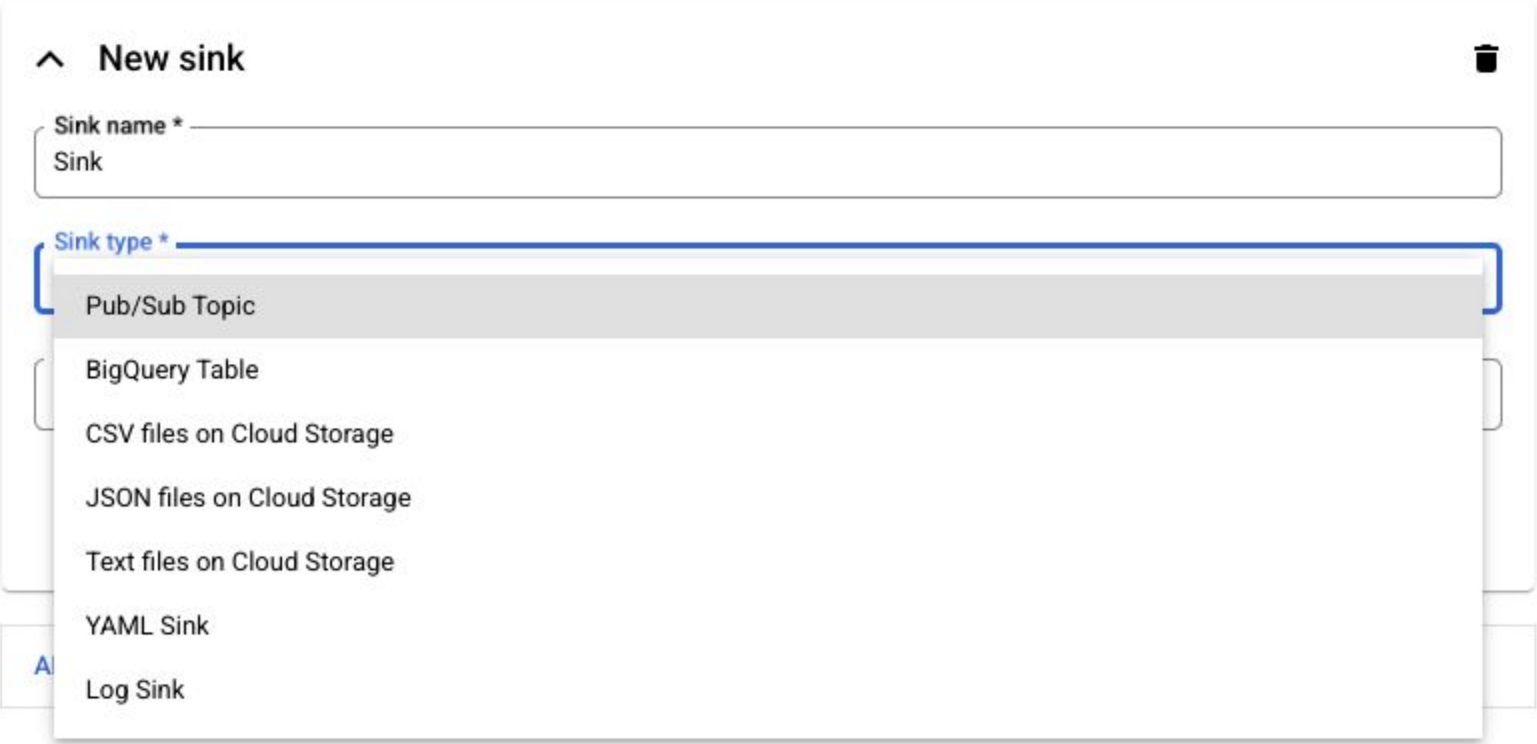
Taxi ride data: input0

Zone lookup data: input1

Press Option+F1 for Accessibility Options.

```sql
SELECT * FROM input0
JOIN input1 ON input0.PULocationID = input1.LocationID
```

A Beam Calcite SQL query to be run over one or more inputs



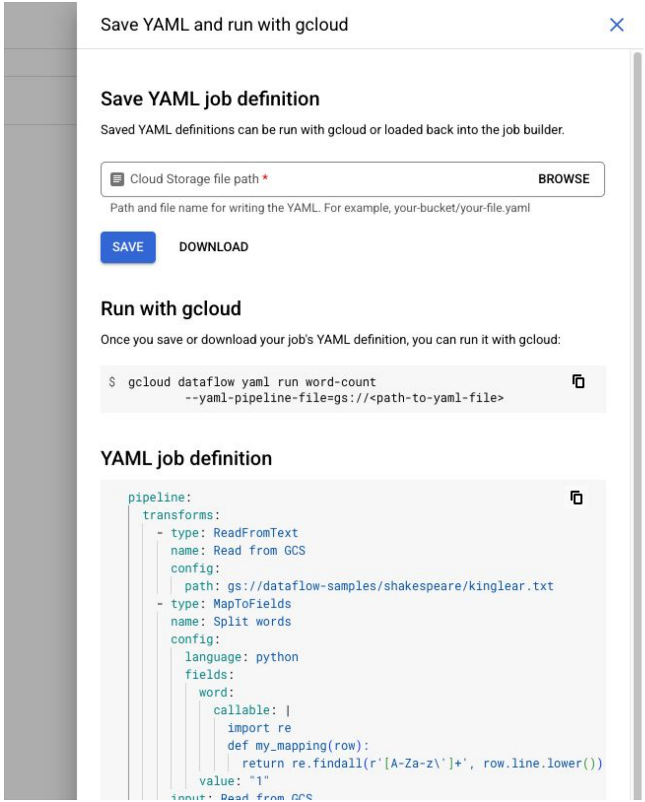
New sink

Sink name *
Sink

Sink type *

Pub/Sub Topic
BigQuery Table
CSV files on Cloud Storage
JSON files on Cloud Storage
Text files on Cloud Storage
YAML Sink
Log Sink
A

Working with YAML



Save YAML and run with gcloud

Save YAML job definition

Saved YAML definitions can be run with gcloud or loaded back into the job builder.

Path and file name for writing the YAML. For example, your-bucket/your-file.yaml

Run with gcloud

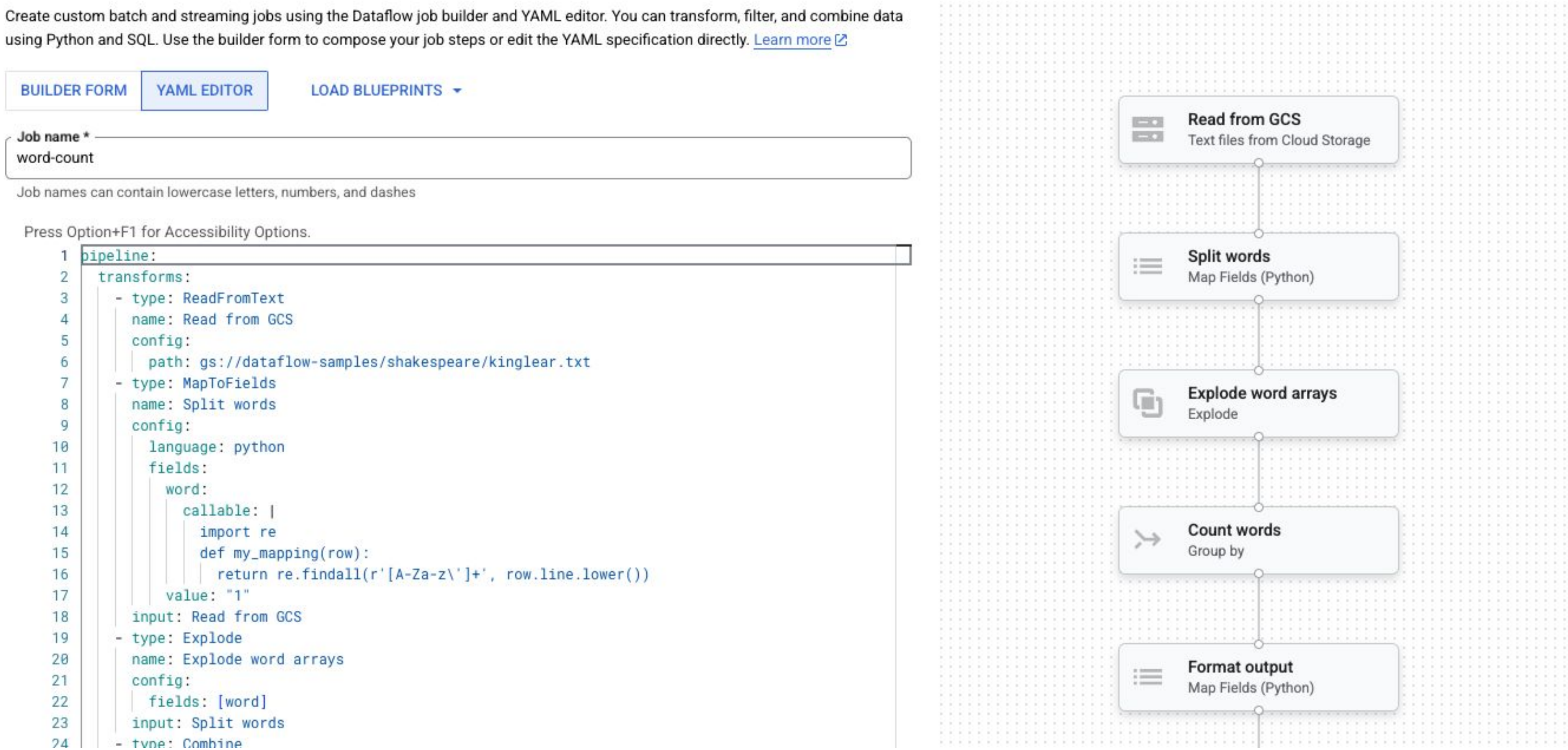
Once you save or download your job's YAML definition, you can run it with gcloud:

```
$ gcloud dataflow yaml run word-count
        --yaml-pipeline-file=gs://<path-to-yaml-file>
```

YAML job definition

```
pipeline:
  transforms:
    - type: ReadFromText
      name: Read from GCS
      config:
        path: gs://dataflow-samples/shakespeare/kinglear.txt
    - type: MapToFields
      name: Split words
      config:
        language: python
        fields:
          word:
            callable: |
              import re
              def my_mapping(row):
                return re.findall(r'[A-Za-z\']+', row.line.lower())
          value: "1"
      input: Read from GCS
```

Working with YAML



Create custom batch and streaming jobs using the Dataflow job builder and YAML editor. You can transform, filter, and combine data using Python and SQL. Use the builder form to compose your job steps or edit the YAML specification directly. Learn more

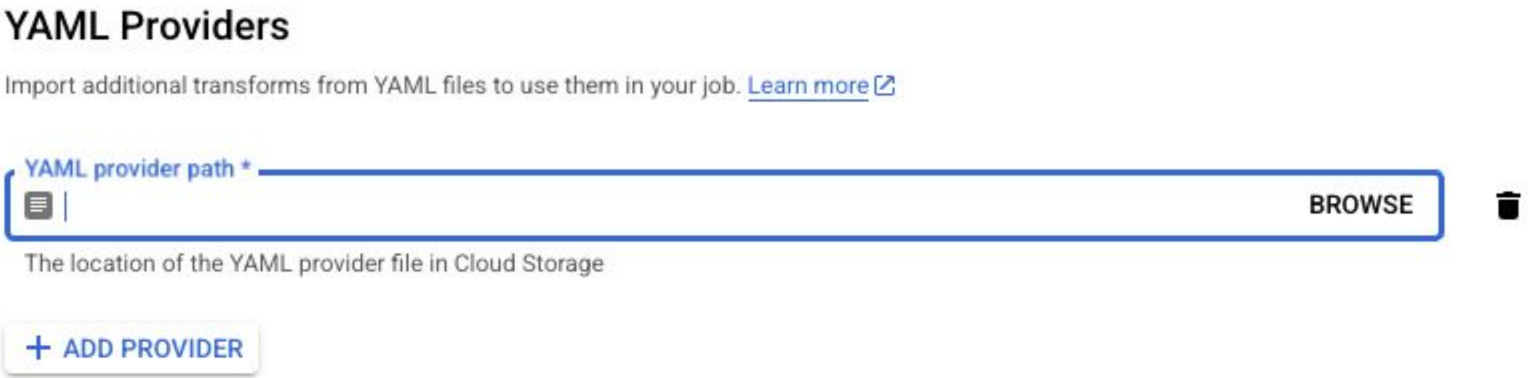
Job name *
word-count

Job names can contain lowercase letters, numbers, and dashes

Press Option+F1 for Accessibility Options.

```yaml
pipeline:
  transforms:
    - type: ReadFromText
      name: Read from GCS
      config:
        path: gs://dataflow-samples/shakespeare/kinglear.txt
    - type: MapToFields
      name: Split words
      config:
        language: python
        fields:
          word:
            callable: |
              import re
              def my_mapping(row):
                return re.findall(r'[A-Za-z\']+', row.line.lower())
          value: "1"
      input: Read from GCS
    - type: Explode
      name: Explode word arrays
      config:
        fields: [word]
      input: Split words
    - type: Combine
```

YAML Providers



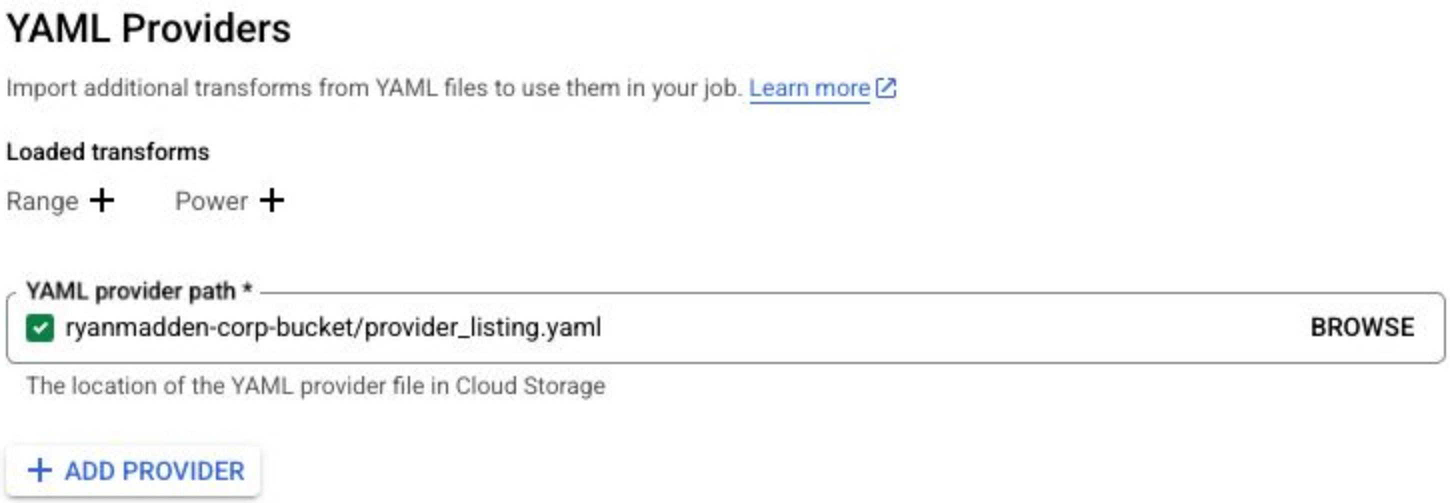
YAML Providers

Import additional transforms from YAML files to use them in your job. Learn more

YAML provider path *

The location of the YAML provider file in Cloud Storage

YAML Providers



YAML Providers

Import additional transforms from YAML files to use them in your job. Learn more

Loaded transforms

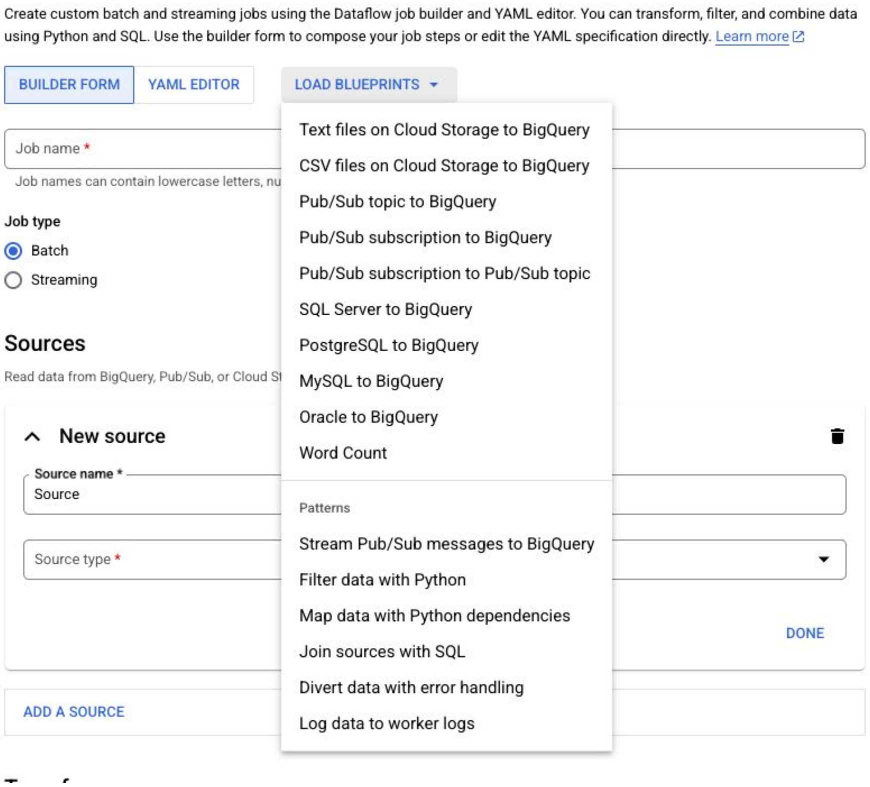
Range + Power +

YAML provider path *

✓ ryanmadden-corp-bucket/provider_listing.yaml

The location of the YAML provider file in Cloud Storage

Getting Started



Create custom batch and streaming jobs using the Dataflow job builder and YAML editor. You can transform, filter, and combine data using Python and SQL. Use the builder form to compose your job steps or edit the YAML specification directly. Learn more

Job name *

Job names can contain lowercase letters, nu

Job type

● Batch

○ Streaming

Sources

Read data from BigQuery, Pub/Sub, or Cloud St

New source

Source name * Source

Source type *

BUILDER FORM
YAML EDITOR
LOAD BLUEPRINTS
Text files on Cloud Storage to BigQuery
CSV files on Cloud Storage to BigQuery
Pub/Sub topic to BigQuery
Pub/Sub subscription to BigQuery
Pub/Sub subscription to Pub/Sub topic
SQL Server to BigQuery
PostgreSQL to BigQuery
MySQL to BigQuery
Oracle to BigQuery
Word Count
Patterns
Stream Pub/Sub messages to BigQuery
Filter data with Python
Map data with Python dependencies
Join sources with SQL
Divert data with error handling
Log data to worker logs
DONE
ADD A SOURCE

Demo!

Learn more

- Dataflow Job Builder
  - http://console.cloud.google.com/dataflow/createjob;createMode=builder
- Job Builder documentation
  - https://cloud.google.com/dataflow/docs/guides/job-builder
- Beam YAML documentation
  - https://beam.apache.org/documentation/sdks/yaml/