

Anyscale Lineage Tracking User Guide

[About Lineage Tracking](#)

[What is Lineage Tracking](#)

[Why Lineage Tracking Matters](#)

[How Anyscale Lineage Tracking Works](#)

[Enable Lineage Tracking](#)

[Enable for Organization](#)

[Enable for Clouds](#)

[Considerations](#)

[Disable Lineage Plugins](#)

[Enable Debug Logging](#)

[Quickstart](#)

[Ray Data](#)

[MLflow](#)

[View Lineage](#)

[Datasets View](#)

[Models View](#)

[Related Workloads](#)

[Lineage Graph](#)

[Related Resources](#)

[Requirements](#)

[Base Image](#)

[Ray Version](#)

[MLflow Version](#)

[Enablement](#)

[Limitations](#)

[Supported Libraries](#)

[Supported APIs](#)

[Ray Data](#)

[MLflow](#)

[Supported Data Sources](#)

[Ray Initialization](#)

[What's Tracked](#)

[Anyscale Workload Details](#)

[Data Source](#)

[Data Schema](#)

[Resources](#)

About Lineage Tracking

NOTE: Lineage tracking is in private beta release. Contact [Anyscale support](#) to enable it for your organization.

What is Lineage Tracking

Lineage tracking lets you see how datasets and models move through your AI pipelines - what jobs produce them, which services consume them, and the environments they run in. Anyscale Lineage Tracking is an OpenLineage-powered observability feature that maps datasets and models across Workspaces, Jobs, and Services and visualizes them as an interactive graph in the Anyscale UI.

Why Lineage Tracking Matters

Without lineage, teams often jump between dashboards, logs, registries, and catalogs to guess which job produced a model or touched a dataset, making debugging and reproduction slow and error-prone. Anyscale Lineage Tracking removes this scavenger hunt by giving a single view that links each dataset and model to its producing and consuming workloads, including logs, parameters, and environment. This helps teams quickly reproduce runs, understand downstream impact before changes, and support audit and governance needs across the model lifecycle.

How Anyscale Lineage Tracking Works

Anyscale Lineage Tracking is built on the OpenLineage standard, with lineage plugins for the Ray Data and MLflow libraries. These plugins are loaded into workloads at runtime and emit OpenLineage events when data is read or written and when models are logged or loaded. Once lineage tracking is enabled for a workload, Anyscale automatically captures lineage metadata with no changes required to user code. The Anyscale Control Plane then normalizes these events, links them to the correct Workspaces, Jobs, and Services, and the Lineage tab in the Anyscale UI renders an interactive graph of lineage across workloads and data artifacts.

Enable Lineage Tracking

Enable for Organization

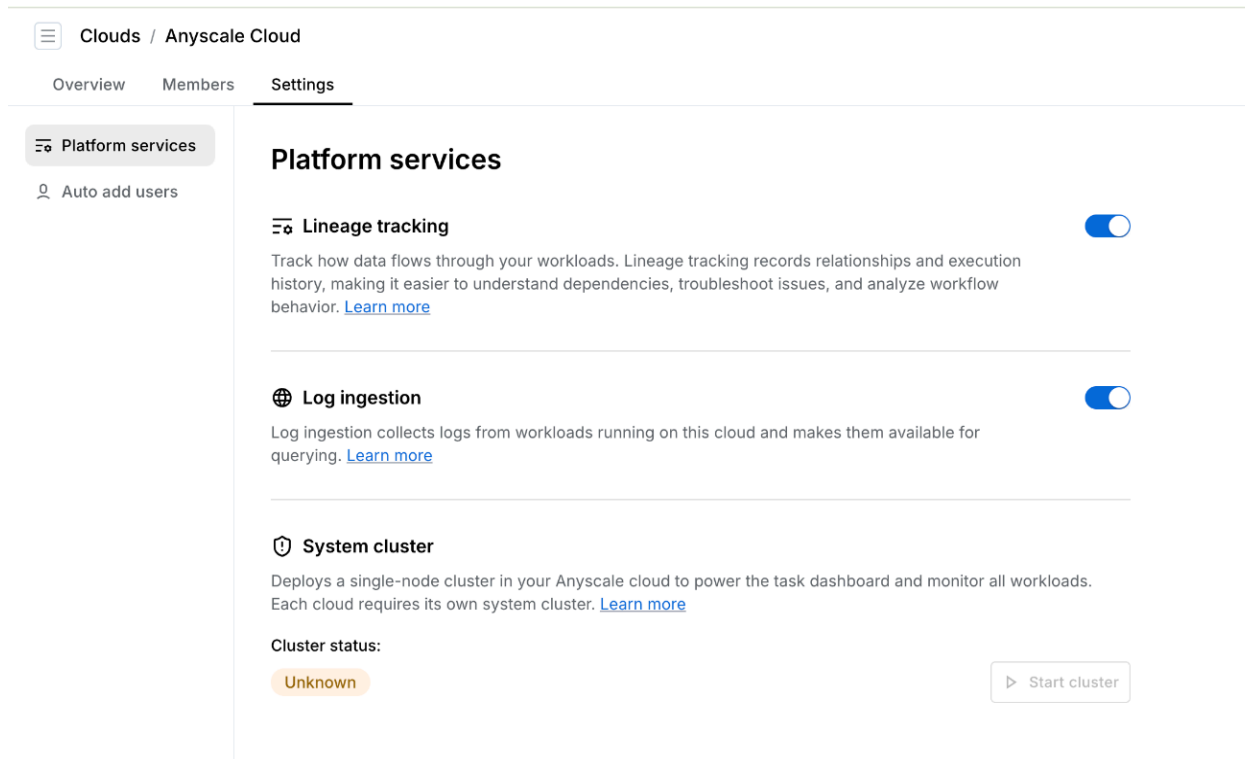
Lineage tracking on Anyscale is in private beta. It's not available by default. To enable lineage tracking features for your organization, contact [Anyscale support](#). Once tracking is enabled for your organization, Anyscale will automatically capture lineage metadata for workloads in lineage-enabled clouds and users can see lineage features on the user interface.

Enable for Clouds

Users have to manually enable lineage tracking for their clouds. The cloud-level toggle ensures that users can selectively enable tracking for specific workloads and that unwanted lineage metadata is not captured.

To enable lineage tracking for a given Anyscale cloud follow these steps:

1. Go to the cloud details page
2. Select the **Settings** tab
3. Select the **Platform services** menu
4. Use the **Lineage tracking** toggle



The screenshot shows the 'Settings' tab for an 'Anyscale Cloud'. The left sidebar has a 'Platform services' menu item selected. The main content area is titled 'Platform services' and contains three settings:

- Lineage tracking**: A toggle switch is turned on. Description: 'Track how data flows through your workloads. Lineage tracking records relationships and execution history, making it easier to understand dependencies, troubleshoot issues, and analyze workflow behavior. [Learn more](#)'
- Log ingestion**: A toggle switch is turned on. Description: 'Log ingestion collects logs from workloads running on this cloud and makes them available for querying. [Learn more](#)'
- System cluster**: A warning icon is shown. Description: 'Deploys a single-node cluster in your Anyscale cloud to power the task dashboard and monitor all workloads. Each cloud requires its own system cluster. [Learn more](#)'

Below the 'System cluster' section, the 'Cluster status' is shown as 'Unknown' in an orange pill. A 'Start cluster' button is located to the right.

Considerations

- You must have **Owner** permissions to perform these actions.
- When you enable lineage tracking, lineage is not captured for clusters that are already running or have been terminated. To capture lineage for running Workspaces, you need to restart them.
- When you disable lineage tracking, lineage will not be captured for new clusters, but it will still be captured for running clusters until they are terminated.

Disable Lineage Plugins

Users may need to disable Anyscale lineage plugins under certain circumstances. This can be done by setting the environment variable `ANYSCALE_LINEAGE_TRACKING_ENABLED` before Ray initialization.

None

```
export ANYSCALE_LINEAGE_TRACKING_ENABLED="false"
```

Enable Debug Logging

If users encounter errors in their workloads that may be related to lineage tracking, they can enable debug logging by setting `ANYSCALE_LINEAGE_LOG_LEVEL` and `ANYSCALE_LINEAGE_IGNORE_ERRORS` environment variables before Ray initialization. They can then share the error snippets or debug logs with the Anyscale team if they need help with troubleshooting.

None

```
export ANYSCALE_LINEAGE_LOG_LEVEL="DEBUG"
export ANYSCALE_LINEAGE_IGNORE_ERRORS="false"
```

Quickstart

Ray Data

Anyscale automatically captures lineage metadata for [Ray Data Datasets](#) if lineage tracking is enabled for the Job, Service, or Workspace executing that dataset. Run the following code snippet in a Job or a Workspace to see lineage tracking in action.

Python

```
import ray

ds = ray.data.read_images(
    "s3://doggos-dataset/train",
    include_paths=True,
    shuffle="files",
)
```

```
# lineage events are emitted only when a dataset is materialized
ds.schema()
```

When you run the above code, the Anyscale lineage plugin for Ray Data automatically captures metadata about the Ray Data dataset and the Anyscale workload (Job or Workspace) where the dataset is executed, and sends it to the Anyscale lineage backend. The lineage relationships are rendered in the user interface as graphs and related resources lists.

MLflow

Similar to Ray Data, the Anyscale lineage plugin for MLflow captures lineage metadata for models and artifacts logged or loaded via MLflow APIs. Running the code in the following snippet will create lineage entities for the model logged using the `mlflow.pytorch.log_model` API.

```
Python
import os
import shutil

import mlflow
import ray
import torch
import torch.nn as nn

# Ray initialization should happen before creating MLflow experiments and runs
ray.init(ignore_reinit_error=True)

EXPERIMENT_NAME = "doggos"
MODEL_REGISTRY = "/mnt/cluster_storage/mlflow/doggos"
ARTIFACT_LOCATION = f"{os.getenv('ANYSCALE_ARTIFACT_STORAGE')}/mlflow/doggos"

if os.path.isdir(MODEL_REGISTRY):
    shutil.rmtree(MODEL_REGISTRY)
os.makedirs(MODEL_REGISTRY, exist_ok=True)

mlflow.set_tracking_uri(f"file:{MODEL_REGISTRY}")

mlflow.create_experiment(
```

```

    EXPERIMENT_NAME,
    artifact_location=ARTIFACT_LOCATION,
)

class SimpleNet(nn.Module):
    def __init__(self, input_dim=10, hidden_dim=16, output_dim=1):
        super().__init__()
        self.net = nn.Sequential(
            nn.Linear(input_dim, hidden_dim),
            nn.ReLU(),
            nn.Linear(hidden_dim, output_dim),
        )

    def forward(self, x):
        return self.net(x)

mlflow.set_experiment(experiment_name=EXPERIMENT_NAME)

with mlflow.start_run() as run:
    model = SimpleNet()
    artifact_path = "simple_pytorch_model"
    mlflow.pytorch.log_model(
        pytorch_model=model,
        artifact_path=artifact_path,
    )

```

View Lineage

Datasets View

Lineage metadata is captured for all datasets that use [Ray Data Input/Output APIs](#) for reading or writing data. The captured metadata appears on the **Datasets** page on the Anyscale console.

All clouds

Datasets

Home

Dashboard

Workspaces

Jobs

Services

Datasets

Models

Configs

Datasets

Anyscale tracked

Datasets are automatically tracked when you read or write using Ray data in Anyscale.

Created by is me

Dataset name	Type	URI	Created at	Created by
train	Image	s3://doggos-dataset/train	3 Dec 2025, 02:14:19	Me
mnt/cluster_storage/doggos/embeddings	Parquet	file://expwrk_u2c6m8dkwc8yffjem...	9 Dec 2025, 16:26:05	Me
mnt/cluster_storage/doggos/preprocessed_data/pre...	Parquet	file://expwrk_u2c6m8dkwc8yffjem...	9 Dec 2025, 21:38:25	Me
mnt/cluster_storage/doggos/preprocessed_data/pre...	Parquet	file://expwrk_u2c6m8dkwc8yffjem...	9 Dec 2025, 21:40:59	Me
val	Image	s3://doggos-dataset/val	9 Dec 2025, 21:40:59	Me
org_7c1Kalm9WcX2bNijW53GUT/cld_kvedZWag2qA...	Unknown	s3://anyscale-staging-data-clid-kve...	10 Dec 2025, 01:16:30	Me
mnt/cluster_storage/doggos/preprocessed_data/pre...	Parquet	file://expwrk_99jgv4zjg8tfxs88mxjn...	11 Dec 2025, 18:00:02	Me
mnt/cluster_storage/doggos/preprocessed_data/pre...	Parquet	file://expwrk_99jgv4zjg8tfxs88mxjn...	11 Dec 2025, 18:00:17	Me
org_7c1Kalm9WcX2bNijW53GUT/cld_kvedZWag2qA...	CSV	s3://anyscale-staging-data-clid-kve...	11 Dec 2025, 22:37:53	Me
org_7c1Kalm9WcX2bNijW53GUT/cld_kvedZWag2qA...	CSV	s3://anyscale-staging-data-clid-kve...	11 Dec 2025, 22:38:43	Me

Rows per page: 10

1 - 10 of 21

The dataset naming follows the [OpenLineage convention](#). In addition to name, we also show dataset type (if it can be inferred from the metadata or file extension), dataset URI, the datetime when the dataset was logged for lineage tracking, and the user who created that dataset. Users can also filter the **Datasets** view with user email (``Created by`` filter), Anyscale cloud and project name (breadcrumb menu).

Clicking on a dataset name takes you to the detailed view of that dataset, where you can see the **Related workloads** and **Lineage** views.

Models View

Models and artifacts logged or loaded using common MLflow APIs are automatically tracked by Anyscale. The lineage metadata for these models and artifacts is displayed on the **Models** page.

All clouds / Models

Home

Dashboard

Workspaces

Jobs

Services

Datasets

Models

Configs

Models

View models auto-tracked in Anyscale or in the connected external registry or catalog.

Created by is me

Model ID ↑↓	URI	Created at ↑	Created by
sample-model	file://expwrk_u2c6m8dkwc8yfljzemdcyun...	9 Dec 2025, 22:13:03	Me
sample-model-v1	file://expwrk_u2c6m8dkwc8yfljzemdcyun...	9 Dec 2025, 22:13:03	Me
sample-model	file://expwrk_u2c6m8dkwc8yfljzemdcyun...	9 Dec 2025, 22:20:26	Me
sample-model-v1	file://expwrk_u2c6m8dkwc8yfljzemdcyun...	9 Dec 2025, 22:20:26	Me
model	file://expwrk_99jgv4zjg8tfxs88mxjn7gd9b2...	11 Dec 2025, 18:25:15	Me
SimplePytorchModel	file://expwrk_99jgv4zjg8tfxs88mxjn7gd9b2...	11 Dec 2025, 18:25:15	Me
model_ankur_901	file://expwrk_u2c6m8dkwc8yfljzemdcyun...	11 Dec 2025, 22:00:31	Me
model_ankur_901	file://expwrk_u2c6m8dkwc8yfljzemdcyun...	11 Dec 2025, 22:06:28	Me
model_ankur_902	file://expwrk_u2c6m8dkwc8yfljzemdcyun...	11 Dec 2025, 22:07:00	Me
sanket_pytorch_model	file://expwrk_99jgv4zjg8tfxs88mxjn7gd9b2...	11 Dec 2025, 22:18:37	Me

Rows per page: 10 1 - 10 of 38

Clicking on a model name takes you to the detailed view of that model, where you can see the **Related workloads** and **Lineage** views.

Related Workloads

The dataset or model **Overview** page also displays the **Related workloads** view. Here you can see all the **Upstream** dependencies and **Downstream** consumers of a given dataset or model.

Datasets / train

Overview

Lineage

train

Created at 3 Dec 2025, 02:14:19 by sanketrai@anyscale.com

Type

Image

URI

s3://doggos-dataset/train

Related workloads

Upstream

No upstream found

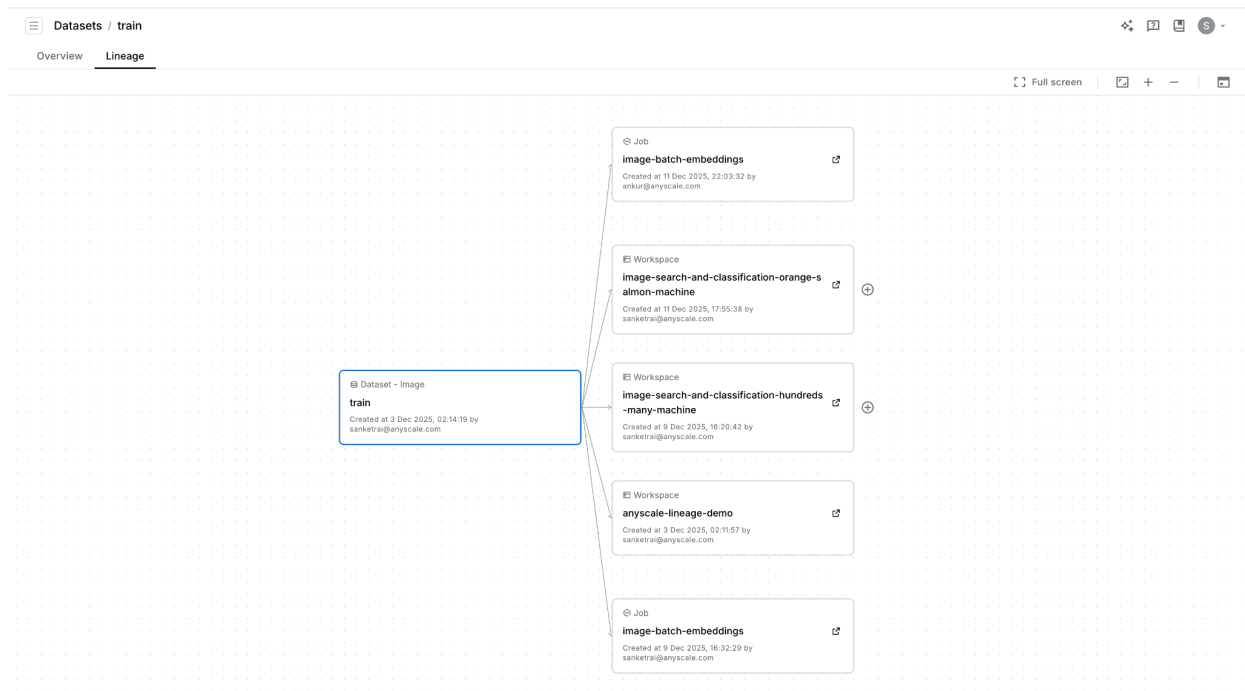
Downstream

Type	Name	Status	Created at	Created by
Job	image-batch-embeddings	Success	11 Dec 2025, 22:03:32	ankur@anyscale.com
Workspace	image-search-and-classificati...	Terminated	11 Dec 2025, 17:55:38	Me
Workspace	image-search-and-classificati...	Terminated	9 Dec 2025, 16:20:42	Me
Workspace	anyscale-lineage-demo	Terminated	3 Dec 2025, 02:11:57	Me
Job	image-batch-embeddings	Success	9 Dec 2025, 16:32:29	Me

Clicking on a workload name will open the **Overview** page of that workload (Job, Service, or Workspace) in a new browser tab.

Lineage Graph

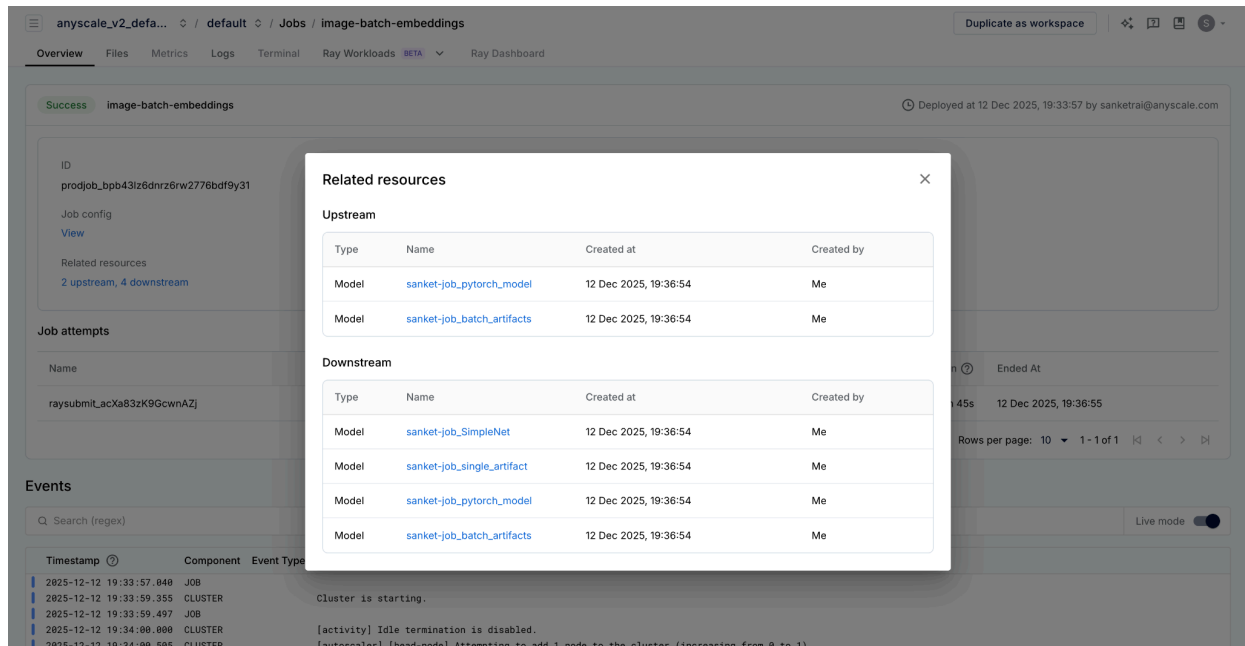
The **Lineage** view renders an expandable lineage graph for a given dataset or model.



By default, we only show one level deep lineage, i.e., the workloads directly related to the dataset or model as an upstream dependency or a downstream consumer. Users can expand the graph to see indirect lineage relationships.

Related Resources

We also show resources (datasets and models) related to a workload (Job, Service, or Workspace) on the workload **Overview** page. When you click **Related resources**, a modal displaying the **Upstream** and **Downstream** related resources appears.



Clicking on a resource name will open the **Overview** page of that resource in a new browser tab.

Requirements

Base Image

Lineage tracking is only available with Anyscale runtime base images. Open-source Ray base images don't support this feature.

Ray Version

You need to use an Anyscale runtime base image with Ray version greater than or equal to **2.53.0** to use lineage tracking.

MLflow Version

The feature is tested for MLflow versions greater than or equal to `2.19.0`. It may work for previous versions, but that's not guaranteed.

Enablement

Lineage tracking should be enabled by an organization admin for their organization and clouds in order for the feature to be available for workloads in those clouds.

Limitations

Supported Libraries

We currently support lineage tracking for **Ray Data** and **MLflow** workloads only. Support for other libraries like Ray Train and Weights and Biases may be added in the future.

Supported APIs

Ray Data

- [Parquet read and write](#)
- [CSV read and write](#)
- [JSON read and write](#)
- [Text read](#)
- [Audio read](#)
- [Avro read](#)
- [Images read and write](#)
- [Binary read](#)
- [TFRecords read and write](#)
- [Video read](#)
- [WebDataset read](#)

MLflow

- [mlflow.<model-flavor>.load_model](#)
- [mlflow.<model-flavor>.log_model](#)
- [mlflow.register_model](#)
- [mlflow.log_artifact](#)
- [mlflow.log_artifacts](#)
- [mlflow.artifacts.download_artifacts](#)

Supported Data Sources

Any data source that can be expressed using the [OpenLineage dataset naming conventions](#) is supported for lineage tracking on Anyscale. For local filesystem sources, Anyscale only supports tracking ``/mnt/cluster_storage/*`` and ``/mnt/shared_storage/*`` paths in [Anyscale shared storage](#). URIs for cluster storage paths use the format

``file://<anyscale-workload-id>/mnt/cluster_storage/*``, where ``<anyscale-workload-id>`` is the ID of an Anyscale workload (Job, Service, or Workspace). URIs for shared storage paths use the format ``file://<anyscale-cloud-id>/mnt/shared_storage/*``, where ``<anyscale-cloud-id>`` is the ID of an Anyscale cloud.

Ray Initialization

Currently Ray initialization is a prerequisite for lineage metadata to be captured for a workload. For Ray Data workloads, Ray initialization is implicit in most cases, so users don't have to initialize Ray explicitly in their code. For MLflow workloads, users should ensure that Ray is initialized before experiments or runs are created.

What's Tracked

Anyscale Workload Details

- Workload type: Job, Service, or Workspace
- Workload name
- Workload ID
- Organization ID
- Cloud ID
- Project ID
- Owner email
- Ray version
- Python version
- Operating system version

Data Source

- URI
- Name
- File format

Data Schema

- Input and output schemas (captured if available)

Resources

- [Announcement blog post](#)
- [Marketing webinar](#)