Google Cloud

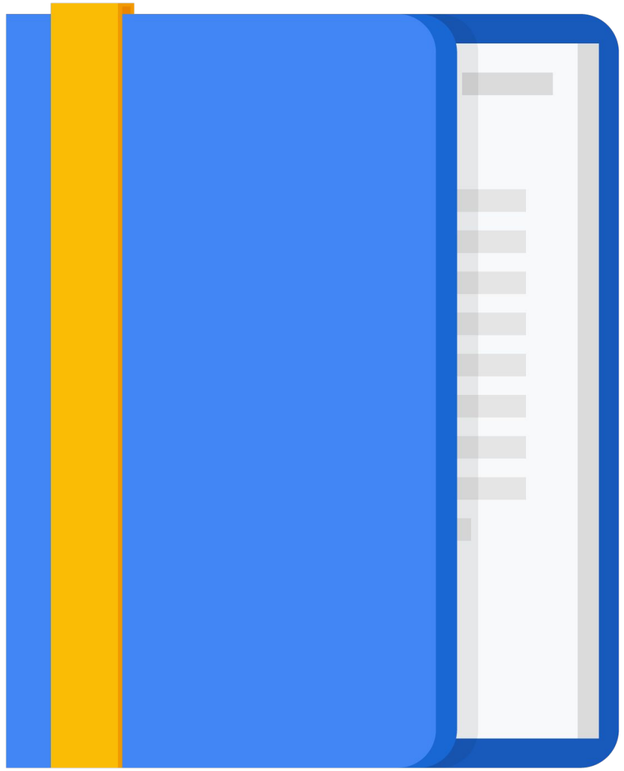
Production ML Pipelines with Kubeflow

Agenda

Ways to do ML on GCP

Kubeflow

AI Hub



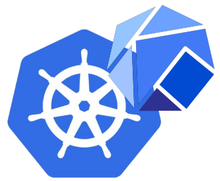
Create and deploy custom models with Kubeflow

Build a Custom Model


Cloud TPUs


Compute Engine


Cloud Dataproc

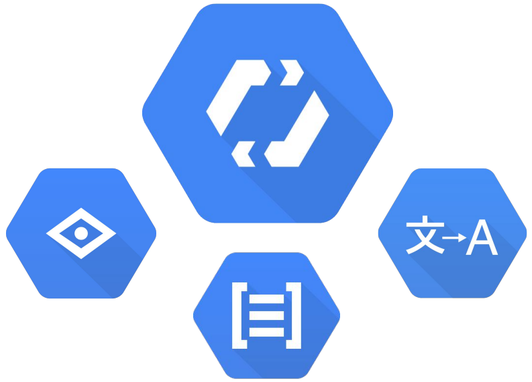

Kubernetes Engine


Cloud AI Platform


BigQuery ML

Build Custom Model (codeless)

AutoML



Call a Pretrained Model


Cloud Translation API


Cloud Vision API


Cloud Speech API


Cloud Video Intelligence API


Data Loss Prevention API

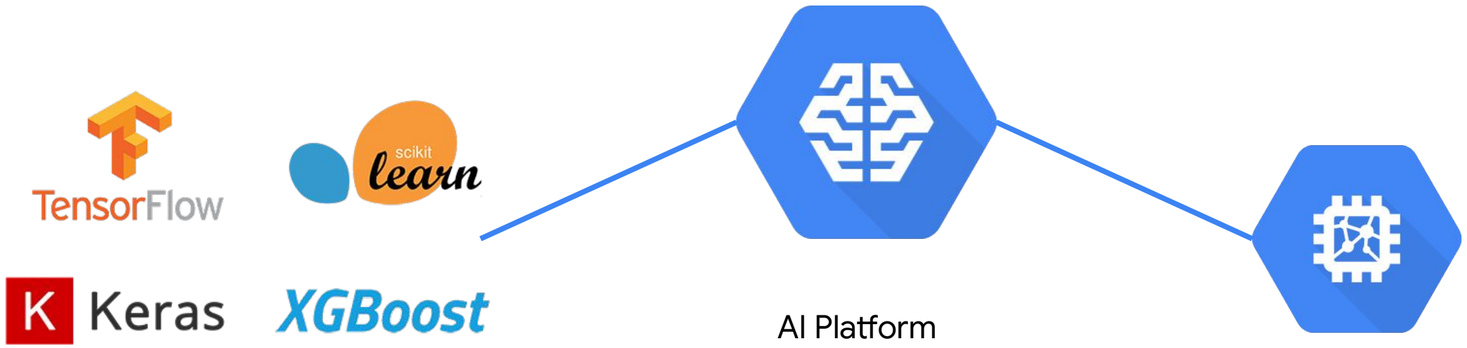

Cloud Speech Synthesis API


Cloud Natural Language API


Dialogflow

Google Cloud

Cloud AI Platform is a fully managed service for custom machine learning models



- Scales to production
- Batching and distribution of model training
- Performs transformations on input data
- Hyper-parameter tuning
- Host and autoscale predictions
- Serverless - self-tuning - manages overhead

In this course, we don't cover writing TensorFlow models, only ways to operationalize them

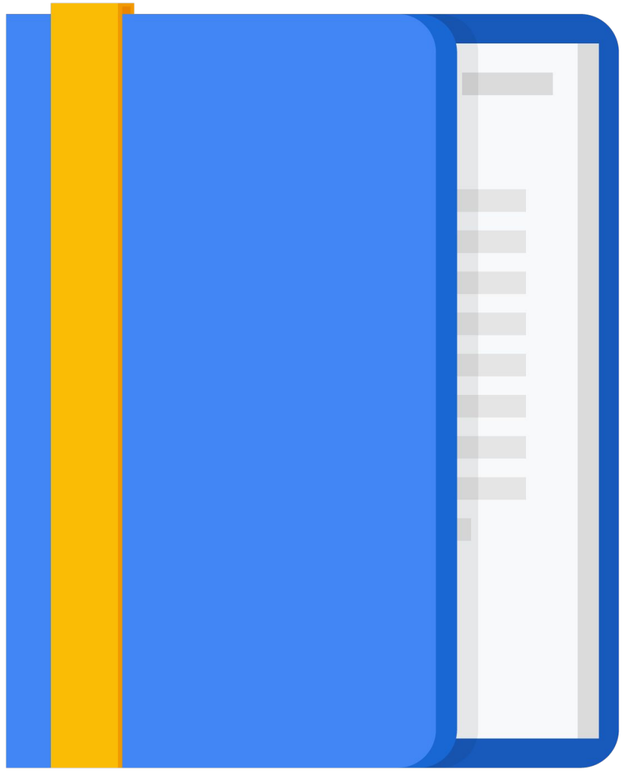
- Intro to ML on GCP Specialization on Coursera

Agenda

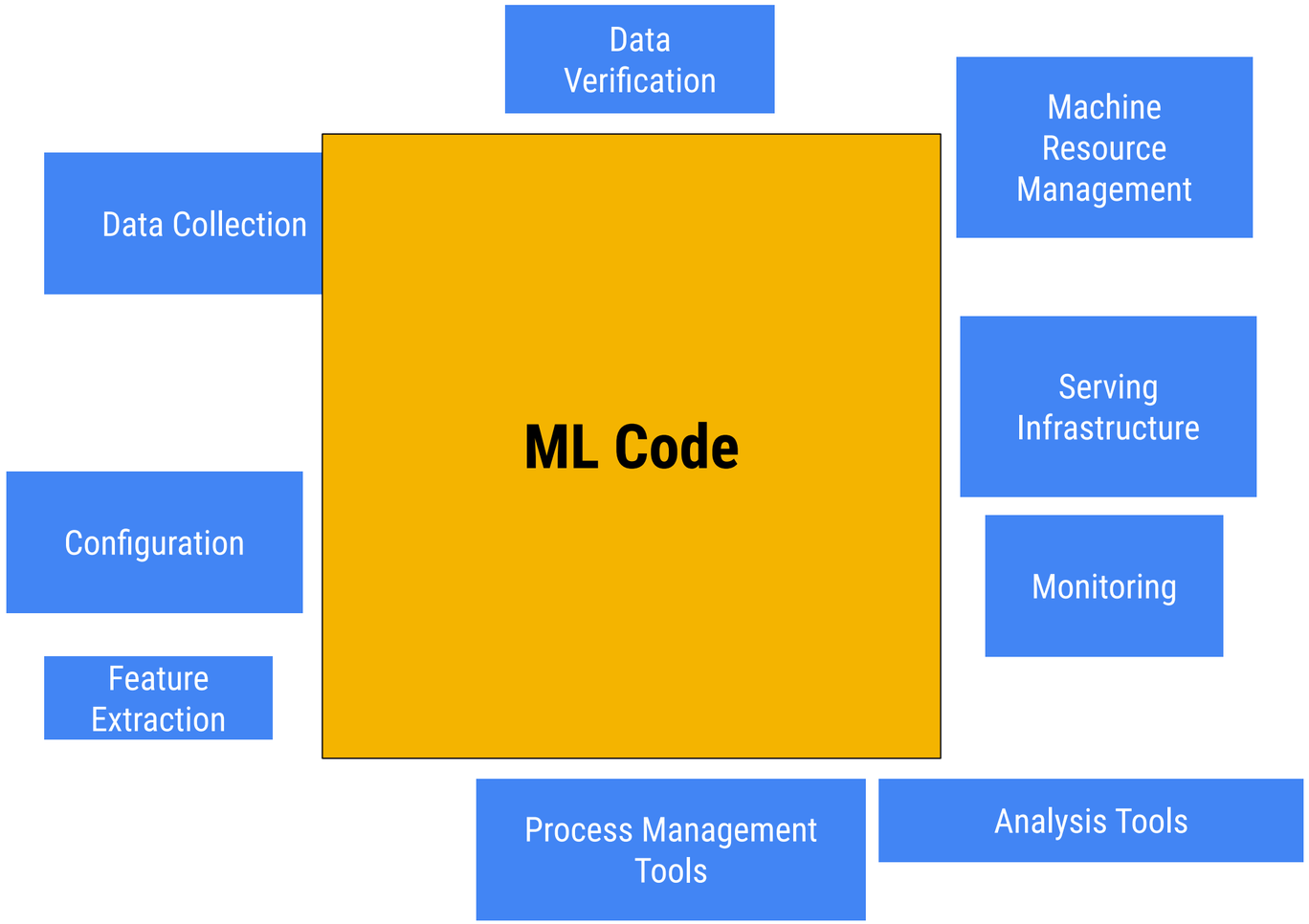
Ways to do ML on GCP

Kubeflow

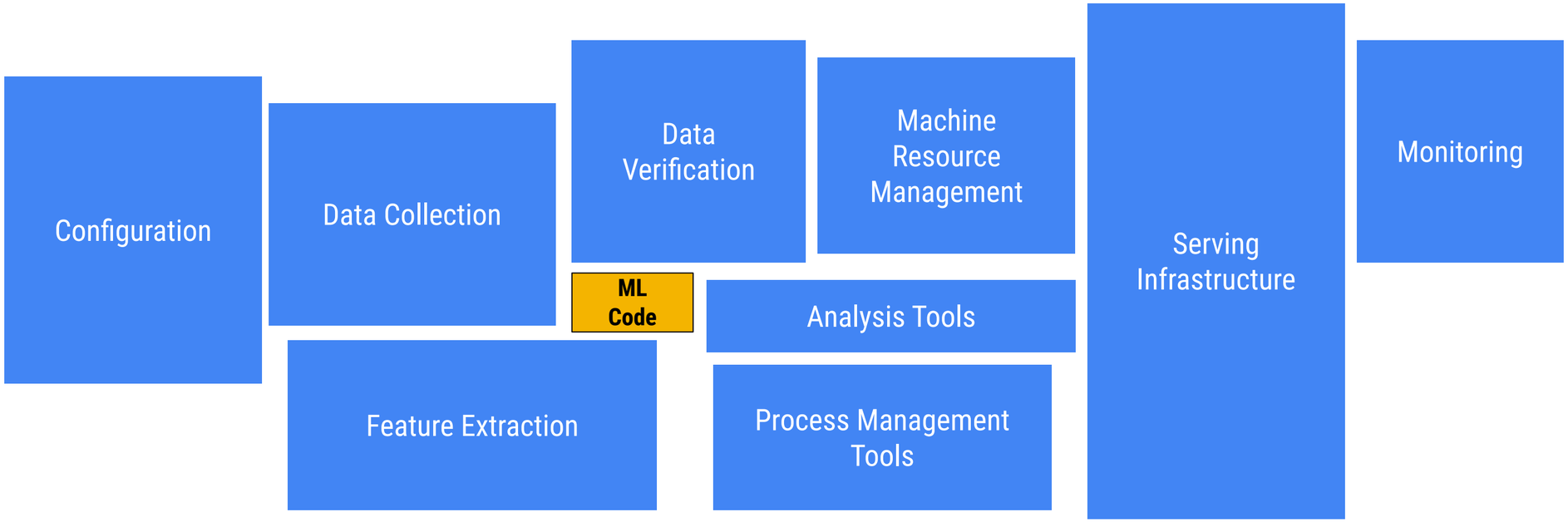
AI Hub



Perception: ML products are mostly about ML



Reality: ML Requires lots of DevOps



Source: Sculley et al.: Hidden Technical Debt in Machine Learning Systems

Kubeflow provides a platform for building ML products

- Leverage containers and Kubernetes to solve the challenges of building ML products
- Kubeflow = Cloud Native, multi-cloud solution for ML.
- Kubeflow provides a platform for composable, portable and scalable ML pipelines
- If you have a Kubernetes conformant cluster, you can run Kubeflow

Kubernetes is a great platform for ML

- Containers
- Scaling built in
- Unified architecture
- Easy to integrate building blocks
  - ML APIs
  - Dataflow
- Lots of options for CI/CD
- Portability
  - Dev, On-Prem, Multi-cloud: same stack

Kubeflow Pipelines enable:

ML Workflow Orchestration

Share, Re-use & Compose

Rapid Reliable Experimentation

What constitutes a Kubeflow Pipeline

Containerized implementations of ML Tasks

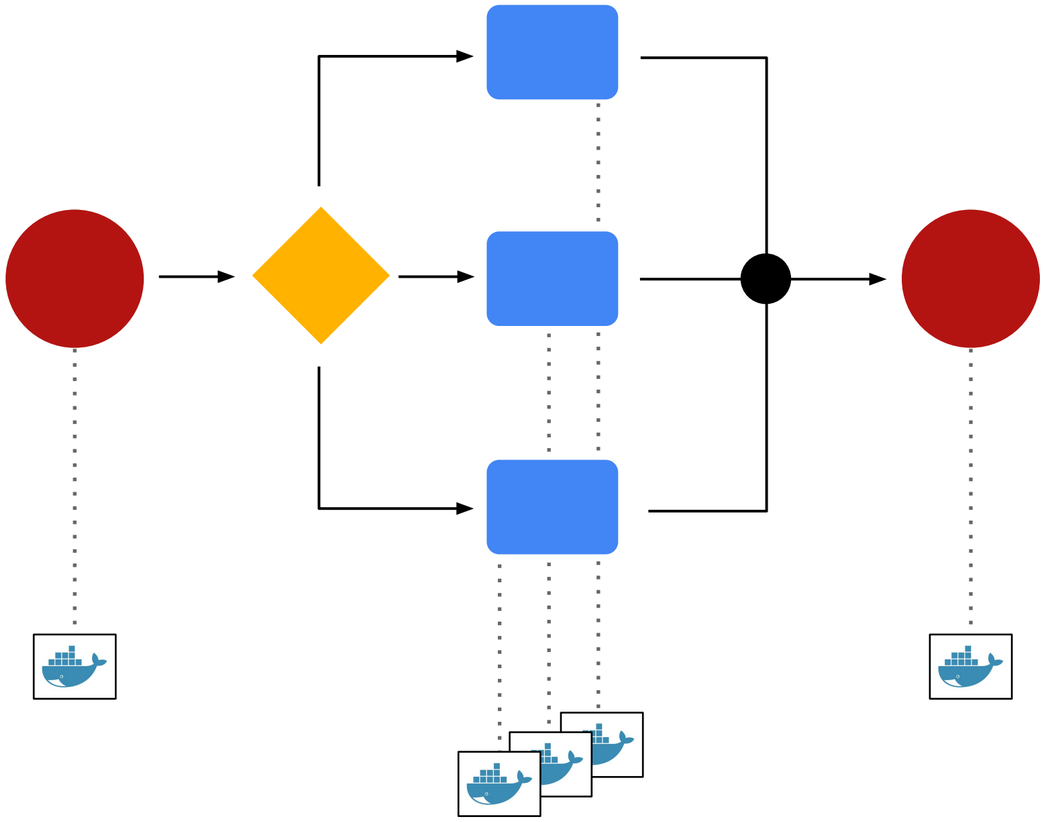
- Containers provide portability, repeatability and encapsulation
- A task can be single node or *distributed*
- A containerized task can invoke other services like CMLE, Dataflow or Dataproc

Specification of the sequence of steps

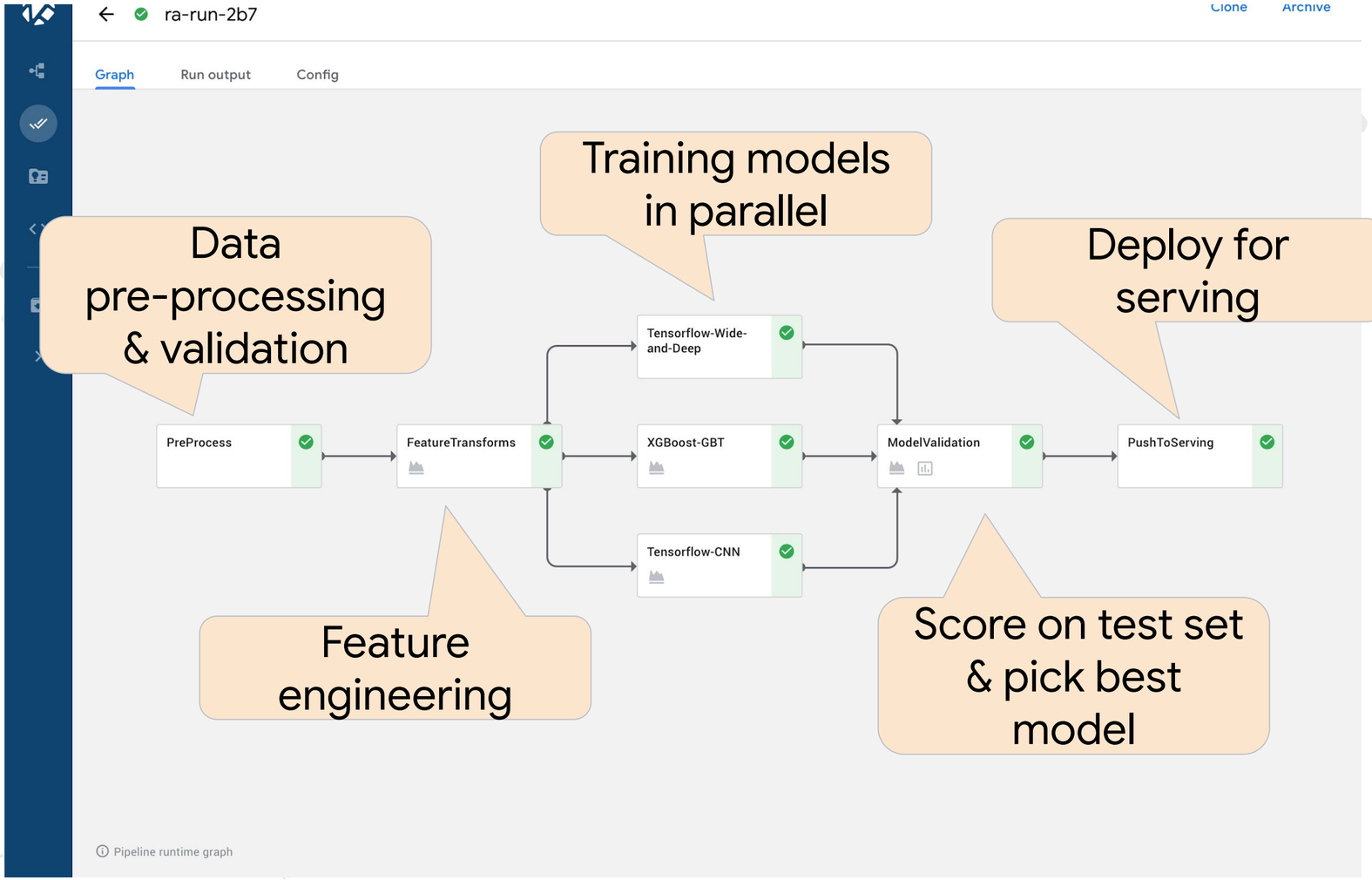
- Specified via Python SDK

Input Parameters

- A “Job” = Pipeline invoked w/ specific parameters

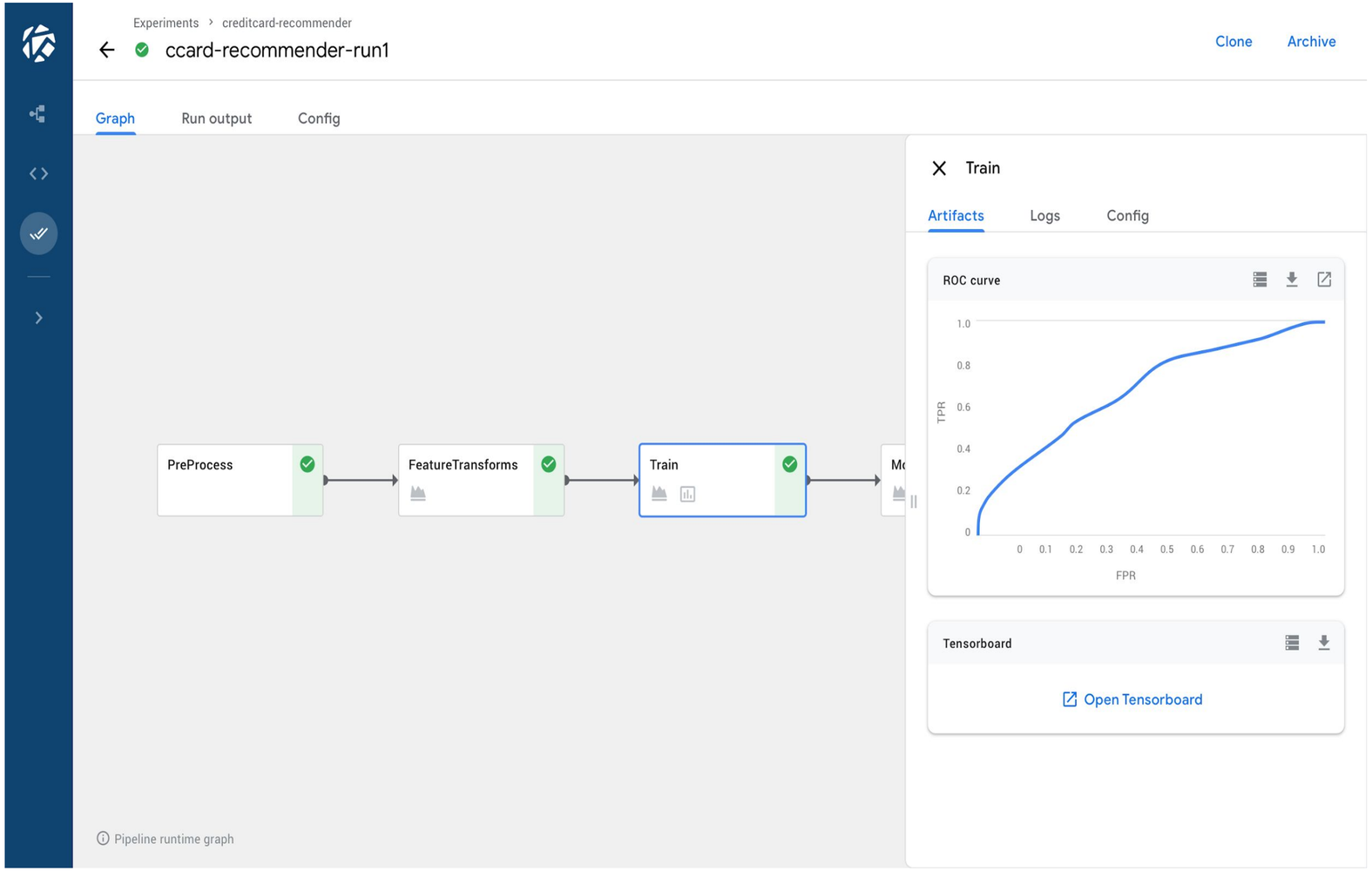


Visual depiction of pipeline topology



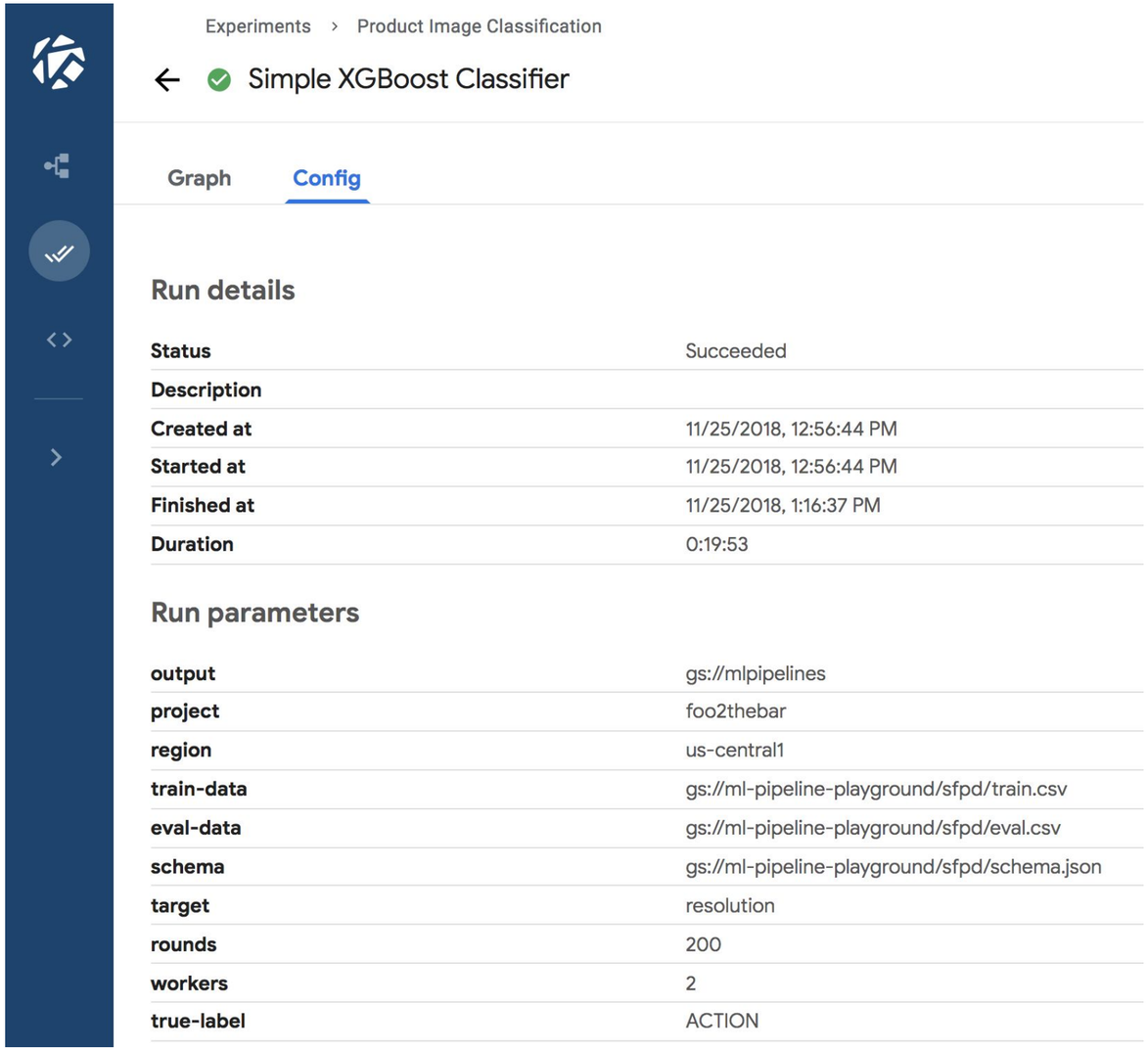
Google Cloud

Rich visualization of metrics



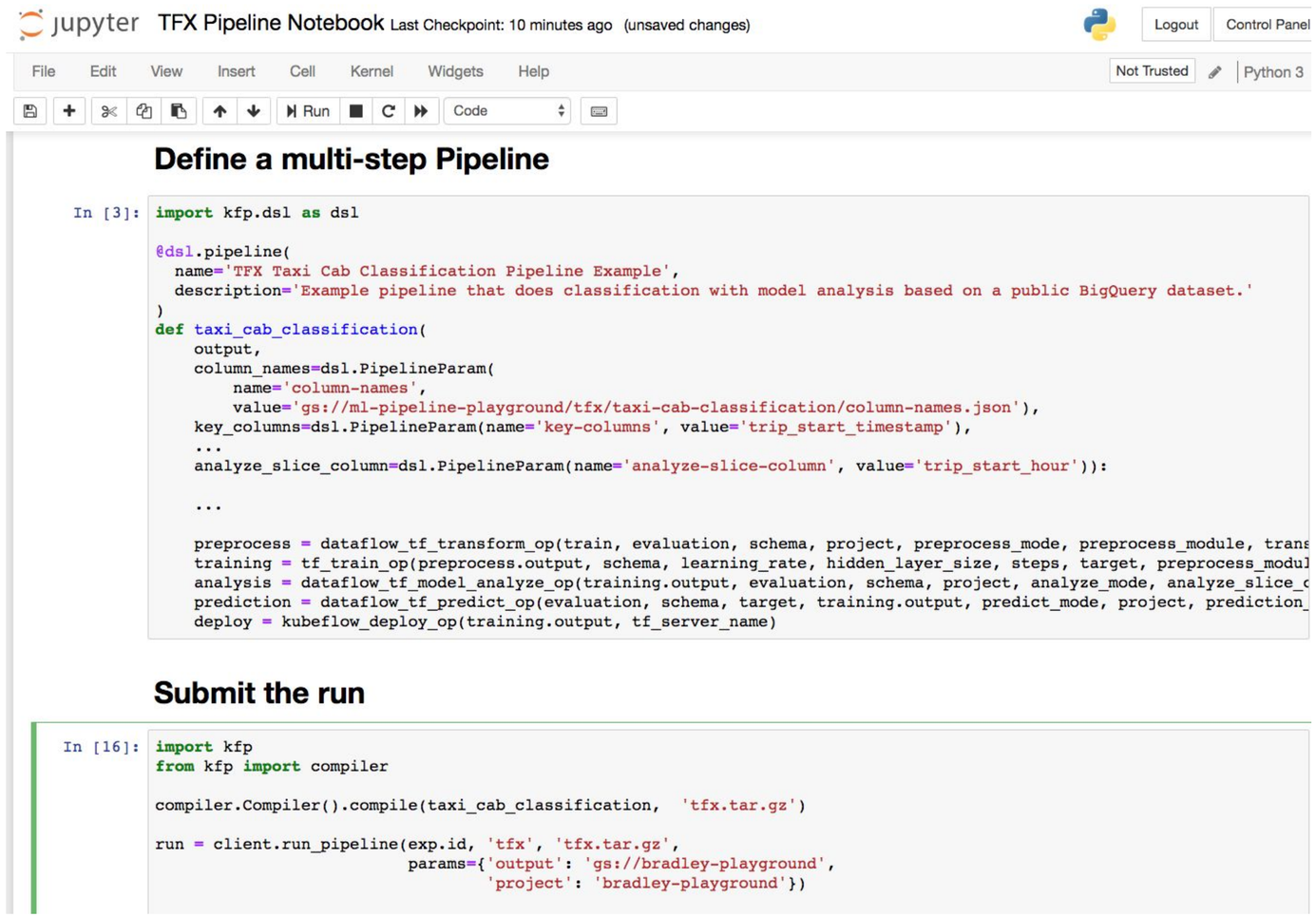
Google Cloud

View all configs, inputs and outputs



Google Cloud

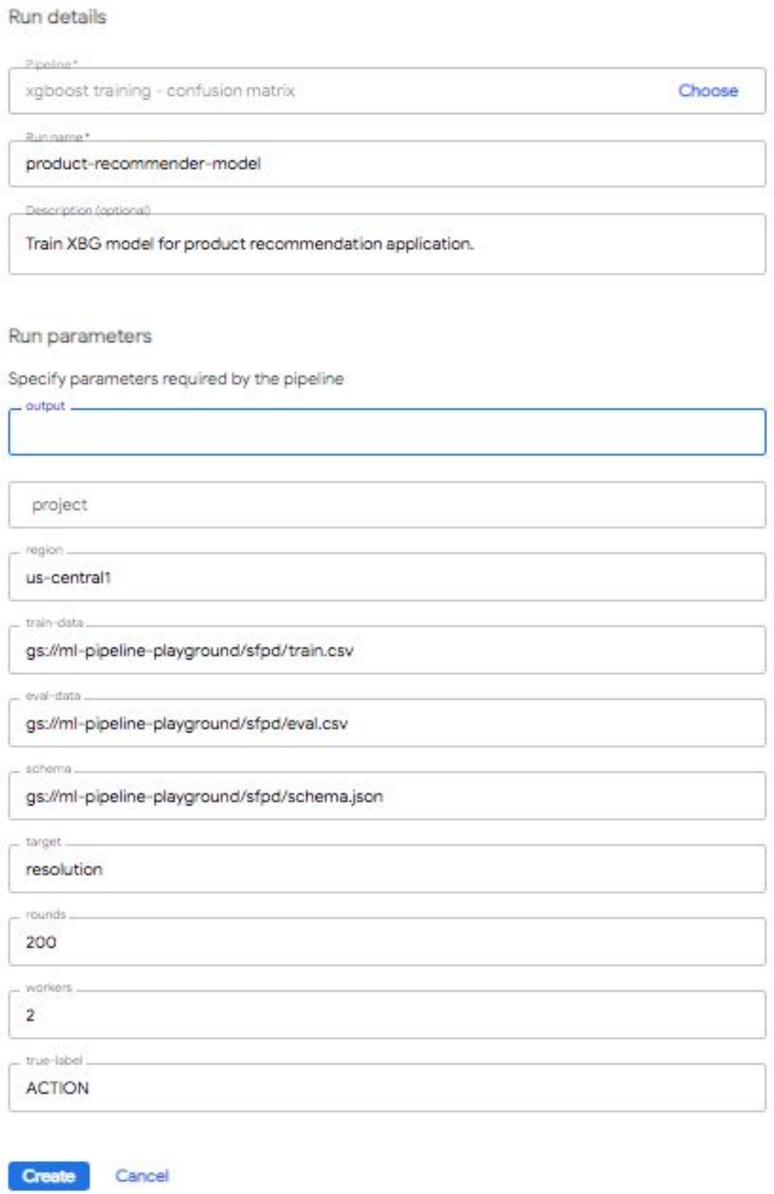
Author pipelines with an intuitive Python SDK



Google Cloud

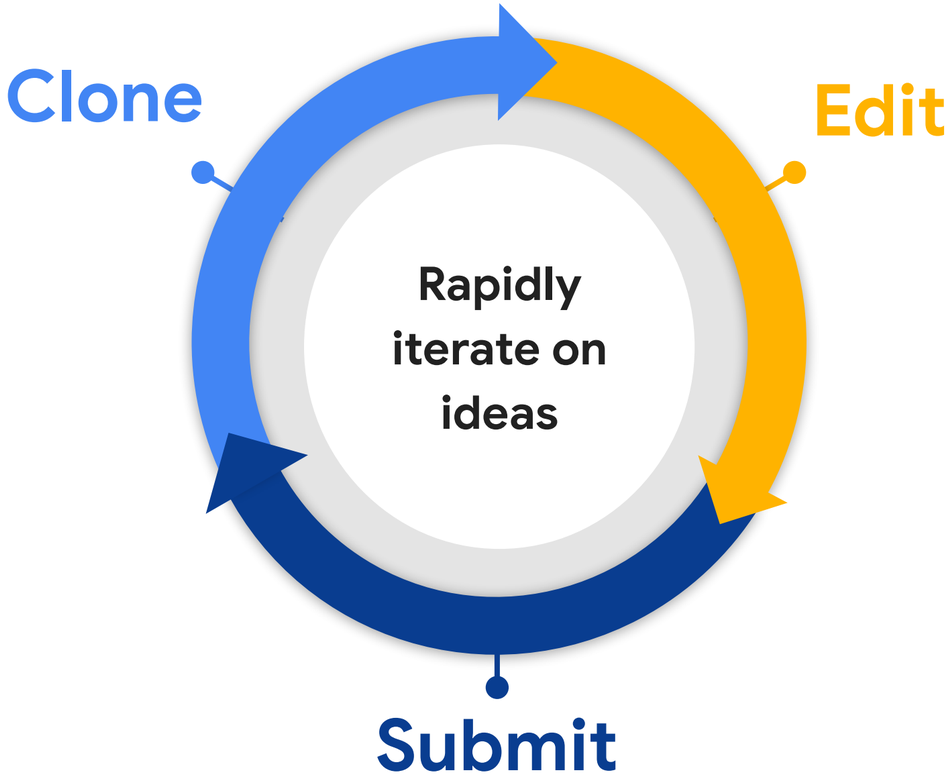
Package & share pipelines as zip files

- Upload and execute pipelines via UI (in addition to API/SDK)
- Pipeline steps can be authored as reusable components

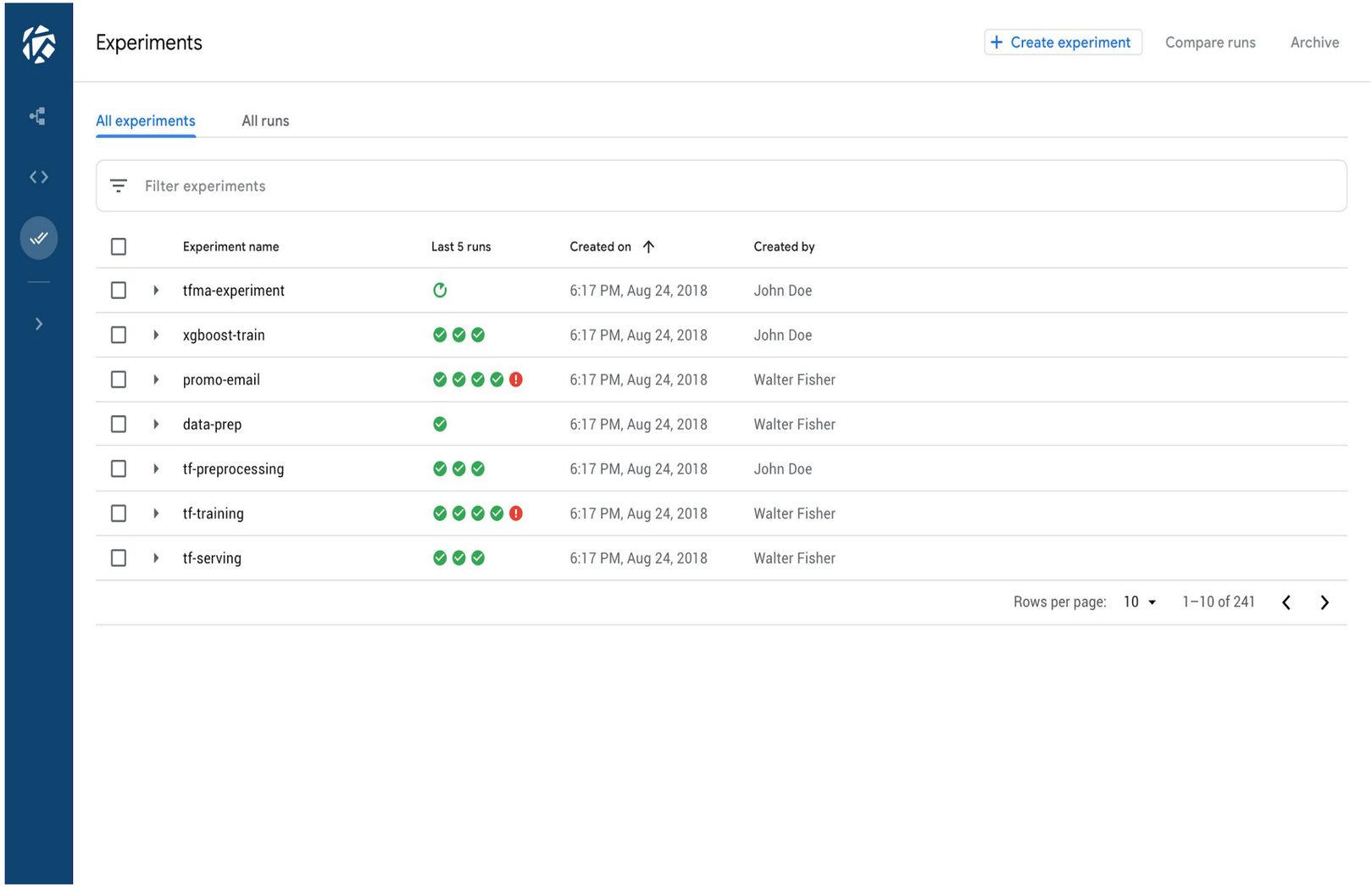


Rapid, Reliable, Experimentation

- Every run logged with all config params, inputs, outputs & metrics
- Easily search and find old runs
- Clone and re-run or modify

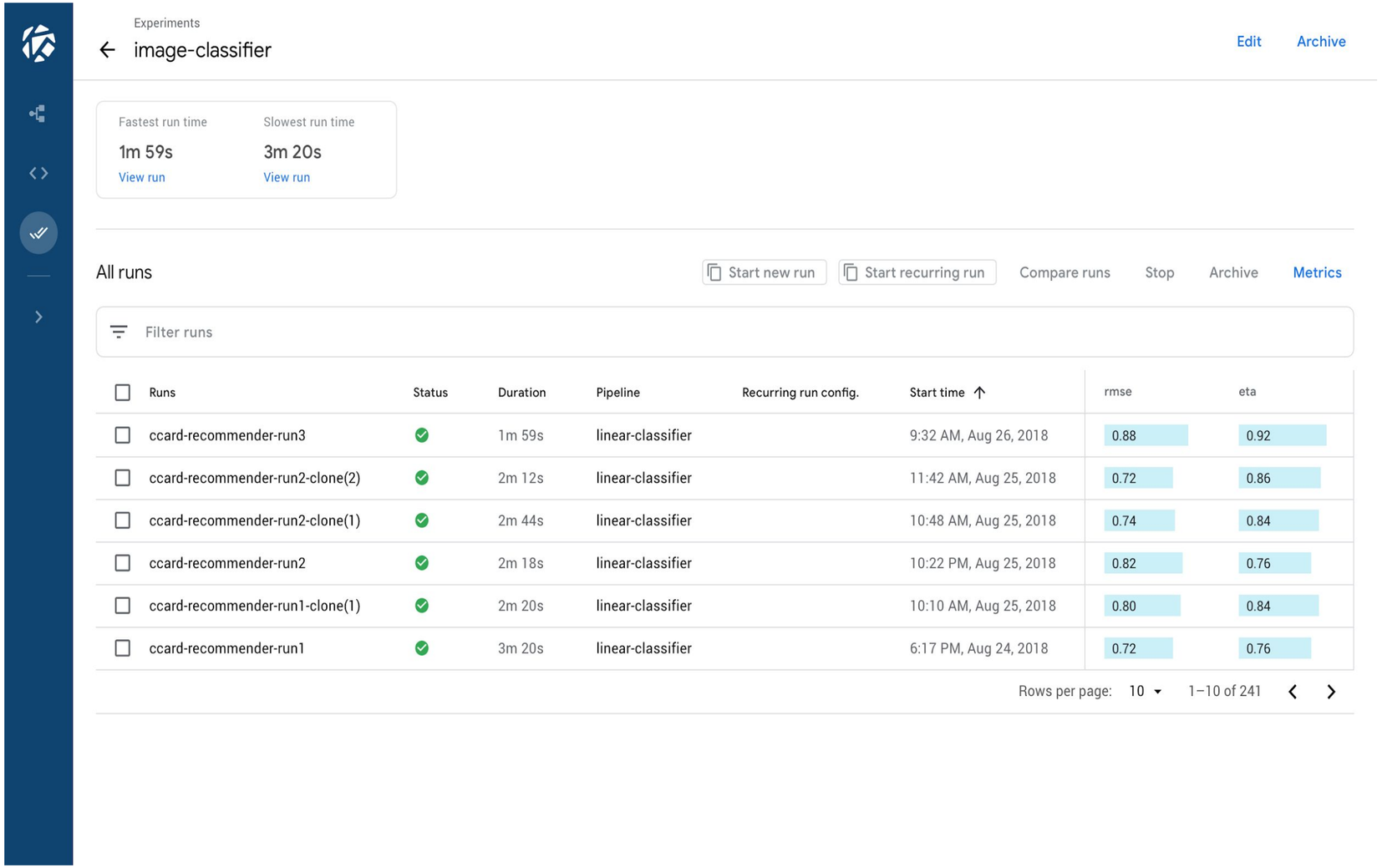


View all current and past runs in one place



Google Cloud

Easy comparison and analysis of runs



Google Cloud

Easy comparison and analysis of runs

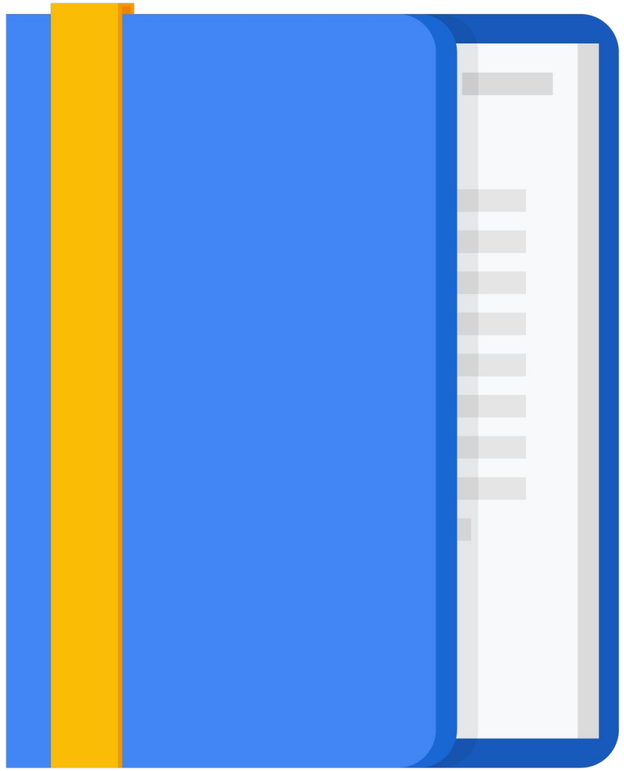
Google Cloud

Agenda

Ways to do ML on GCP

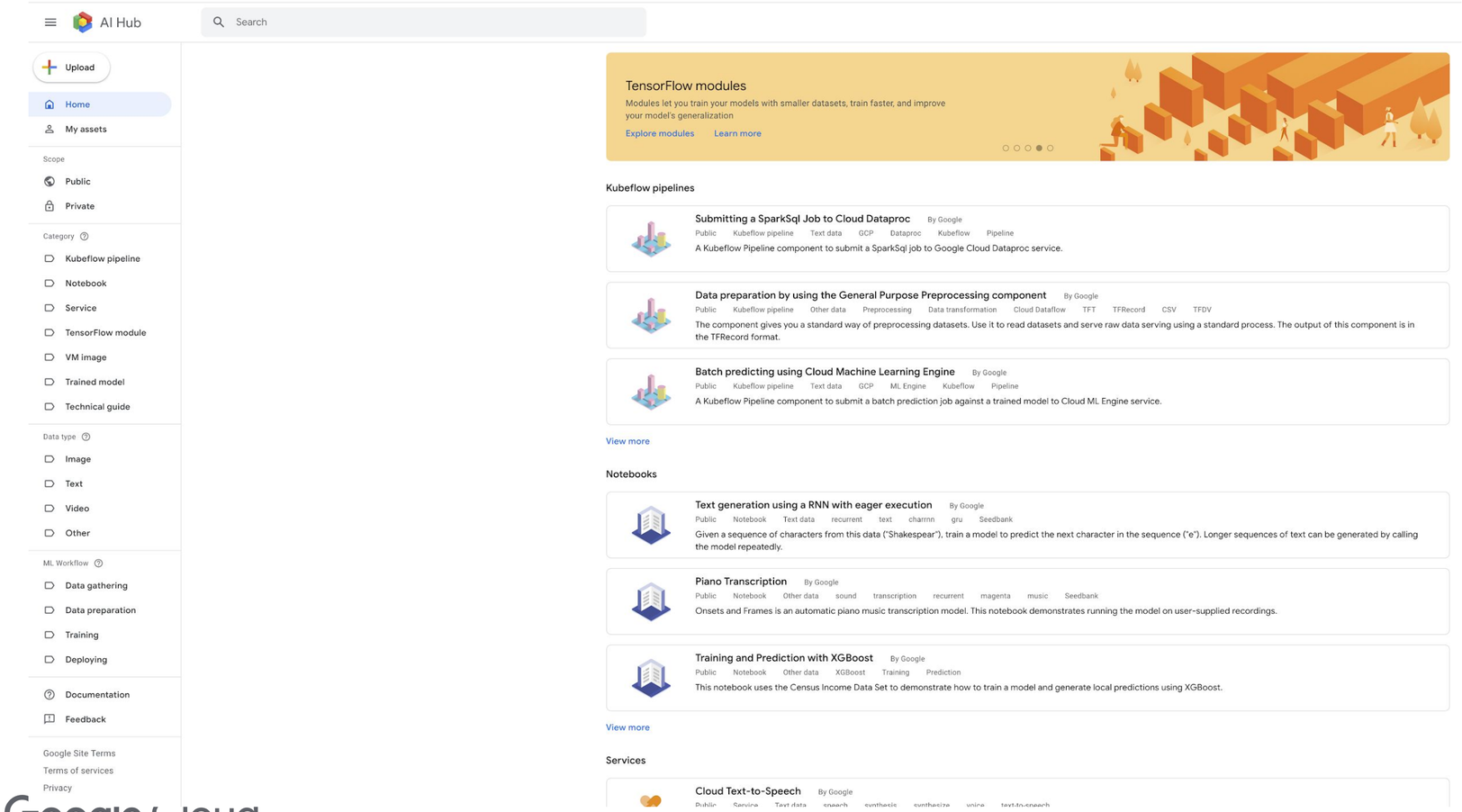
Kubeflow

AI Hub



AI Hub is a repository for AI assets

- Don’t reinvent the wheel! Find and deploy ML pipelines

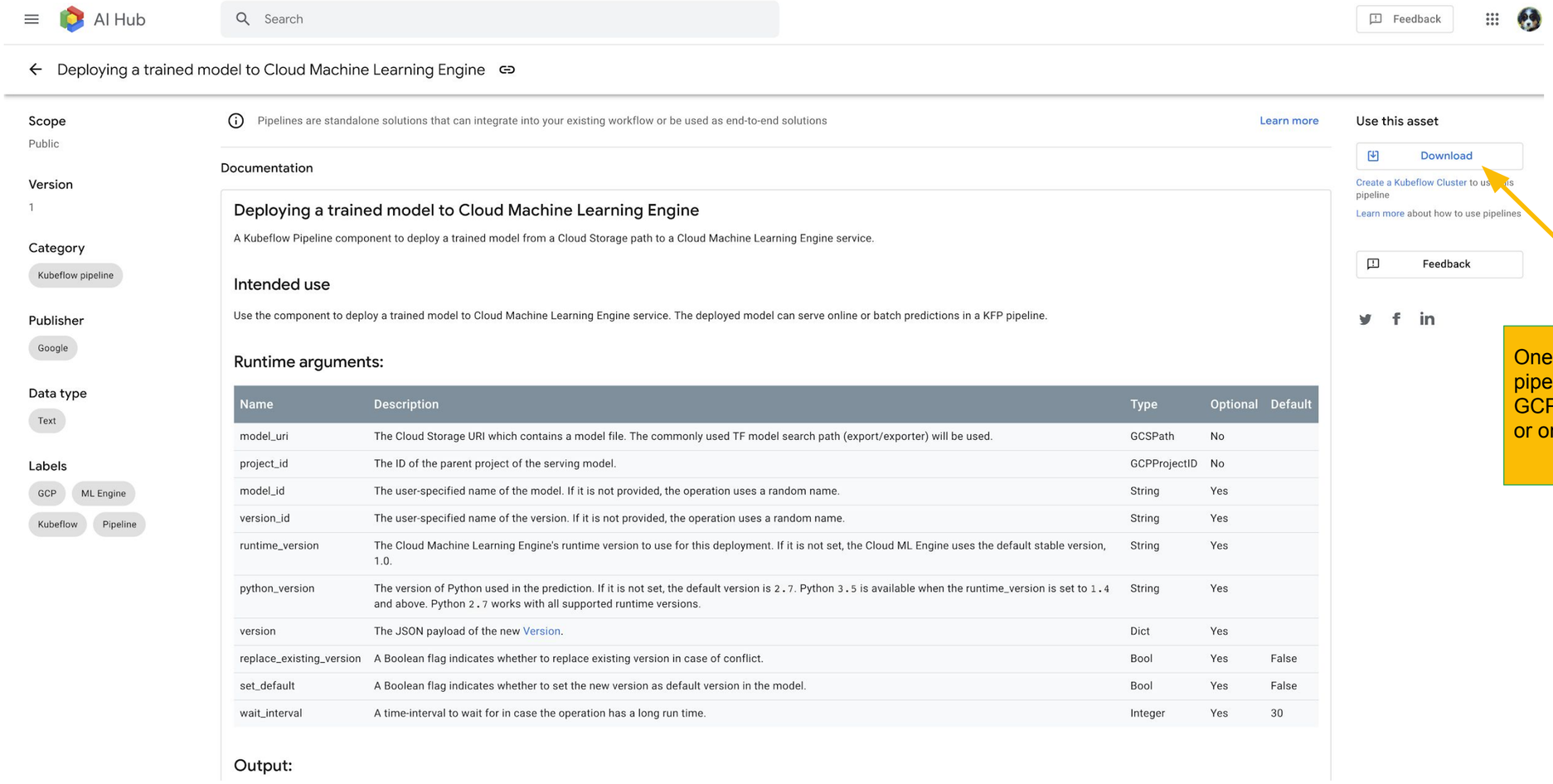


Google Cloud

AI Hub stores various asset types

- Kubeflow pipelines and components
- Jupyter notebooks
- TensorFlow modules
- Trained models
- Services
- VM images

This is what a typical asset looks like...



One-click deployment of ML pipelines via Kubeflow on GCP as platform for AI, or on premise.

Google Cloud

Assets on AI Hub are collected in two scopes: public assets and restricted assets

- Public scope are available to all AI Hub users
- Restricted scope contains AI components that you have uploaded and assets that have been shared with you

Lab

Running AI models on Kubeflow

Objectives

- Set up Kubeflow on a Kubernetes Engine cluster
- Package a TensorFlow program in a container and upload it to Google Container Registry
- Submit a tf-train job and save the resulting model to Google Cloud Storage
- Serve and interact with a trained model

Module Summary

- Use ML on GCP using either
  - AI Platform (your model, your data)
  - AutoML (our models, your data)
  - Perception API (our models, our data)
- Use Kubeflow to deploy end-to-end ML pipelines
- Don’t reinvent the wheel for your ML pipeline! Leverage pipelines on AI Hub