Arrikto | Google

From Notebook to Kubeflow Pipelines to KFServing
The Data Science Odyssey

A complete data science workflow,
from notebooks to model serving
with Kubeflow Pipelines, KF Serving, and Kale

Stefano Fioravanzo
Arrikto

Karl Weinmeister
Google

KubeCon | CloudNativeCon
North America 2020
Virtual

From Notebook to Kubeflow Pipelines to KFServing
The Data Science Odyssey

Arrikto | Google

Stefano Fioravanzo

Software Engineer

Karl Weinmeister

Cloud AI Developer Advocacy Manager

What You'll Learn In This Session

Arrikto | Google

Run hyperparameter optimization with the click of a button, and serve the best result using KF Serving and Kale

Why is this important?

- ✓ **Simplify** your HP tuning and Serving workflows using intuitive UIs
- ✓ **Accelerate** your time to production. You can now reduce the training time and the time needed from training to serving
- ✓ **Collaborate** faster, reducing the friction between the data science team and the MLOps team

Don't forget, you can grab the slides right now at arrik.to/kubeconBOS as well as enter the draw to win a fabulous prize

*Get your questions answered **live** on Twitter and LinkedIn using the three hashtags* #kubecon #ml #arrikto

What is Kubeflow

Kubeflow

The Kubeflow project is dedicated to making deployments of machine learning (ML) workflows on Kubernetes: simple, portable and scalable.

Use cases

Arrikto | Google

- Deployment and management of a complex ML system at scale
- Rapid experimentation
- Hyperparameter tuning
- Hybrid and multi-cloud workloads
- Continuous integration and deployment (CI/CD)

Arrikto | Google

ML tools

Chainer
Jupyter
MPI
MXNet
PyTorch
scikit-learn
TensorFlow
XGBoost

Kubeflow applications and scaffolding

Jupyter notebook web app and controller
Kubeflow UI
Training operators: MPI, MXNet, PyTorch, TFJob, XGBoost
Hyperparameter tuning (Katib)
Kale
Metadata
Pipelines
KFServing

PyTorch Serving
TensorFlow Serving
Seldon Core

Istio
Argo
Prometheus
Spartakus

Platforms / clouds

Kubernetes

GCP
AWS
Azure
IBM Cloud
OpenShift
On prem

ML tools

Chainer | Jupyter | MPI | MXNet
PyTorch | scikit-learn | TensorFlow | XGBoost

Arrikto | Google

Kubeflow applications and scaffolding

- Jupyter notebook web app and controller
- Kubeflow UI
- Training operators: MPI, MXNet, PyTorch, TFJob, XGBoost
- Hyperparameter tuning (Katib)
- Kale
- Metadata
- Pipelines
- KFServing

PyTorch Serving
TensorFlow Serving
Seldon Core

Istio
Argo
Prometheus
Spartakus

Platforms / clouds

Kubernetes

GCP | AWS | Azure | IBM Cloud | OpenShift | On prem

ML tools

Chainer | Jupyter | MPI | MXNet
PyTorch | scikit-learn | TensorFlow | XGBoost

Arrikto | Google

Kubeflow applications and scaffolding

- Jupyter notebook web app and controller
- Kubeflow UI
- Training operators: MPI, MXNet, PyTorch, TFJob, XGBoost
- Hyperparameter tuning (Katib)
- Kale
- Metadata
- Pipelines
- KFServing

PyTorch Serving
TensorFlow Serving
Seldon Core

Istio
Argo
Prometheus
Spartakus

Platforms / clouds

Kubernetes

GCP | AWS | Azure | IBM Cloud | OpenShift | On prem

Arrikto | Google

ML tools

- Chainer
- Jupyter
- MPI
- MXNet
- PyTorch
- scikit-learn
- TensorFlow
- XGBoost

Kubeflow applications and scaffolding

- Jupyter notebook web app and controller
- Kubeflow UI
- Training operators: MPI, MXNet, PyTorch, TFJob, XGBoost
- Hyperparameter tuning (Katib)
- Kale
- Metadata
- Pipelines
- KFServing

- PyTorch Serving
- TensorFlow Serving
- Seldon Core

- Istio
- Argo
- Prometheus
- Spartakus

Platforms / clouds

Kubernetes

- GCP
- AWS
- Azure
- IBM Cloud
- OpenShift
- On prem

ML workflow

Arrikto | Google

Identify problem and collect and analyse data	Choose an ML algorithm and code your model	Experiment with data and model training	Tune the model hyperparameters	Serve the model for online/batch prediction
Jupyter Notebook	PyTorch	Jupyter Notebook	Katib	KFServing
	scikit-learn	Kale		NVIDIA TensorRT
	TensorFlow	Pipelines		PyTorch
	XGBoost			TFServing
				Seldon Core

Kubeflow

Interacting with Kubeflow

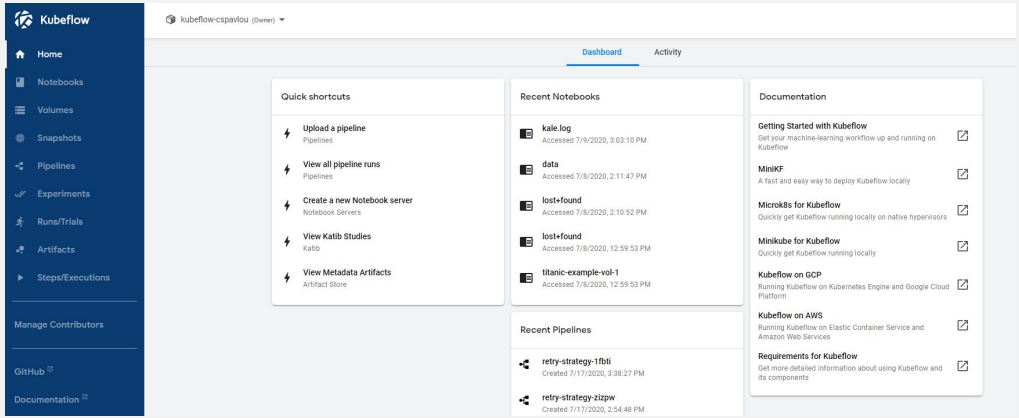
Arrikto | Google

User interface (UI)

kfctl CLI

kubectl CLI

APIs and SDKs



Arrikto | Google

Kubeflow

kubeflow-cspavlou (Owner)

- Home
- Notebooks
- Volumes
- Snapshots
- Pipelines
- Experiments
- Runs/Trials
- Artifacts
- Steps/Executions

Manage Contributors

GitHub

Documentation

Dashboard | Activity

Quick shortcuts

- Upload a pipeline — Pipelines
- View all pipeline runs — Pipelines
- Create a new Notebook server — Notebook Servers
- View Katib Studies — Katib
- View Metadata Artifacts — Artifact Store

Recent Notebooks

- kale.log — Accessed 7/9/2020, 3:03:10 PM
- data — Accessed 7/8/2020, 2:11:47 PM
- lost+found — Accessed 7/8/2020, 2:10:52 PM
- lost+found — Accessed 7/8/2020, 12:59:53 PM
- titanic-example-vol-1 — Accessed 7/8/2020, 12:59:53 PM

Recent Pipelines

- retry-strategy-1fbti — Created 7/17/2020, 3:38:27 PM
- retry-strategy-zizpw — Created 7/17/2020, 2:54:48 PM

Documentation

- **Getting Started with Kubeflow** — Get your machine-learning workflow up and running on Kubeflow
- **MiniKF** — A fast and easy way to deploy Kubeflow locally
- **Microk8s for Kubeflow** — Quickly get Kubeflow running locally on native hypervisors
- **Minikube for Kubeflow** — Quickly get Kubeflow running locally
- **Kubeflow on GCP** — Running Kubeflow on Kubernetes Engine and Google Cloud Platform
- **Kubeflow on AWS** — Running Kubeflow on Elastic Container Service and Amazon Web Services
- **Requirements for Kubeflow** — Get more detailed information about using Kubeflow and its components

Interacting with Kubeflow

Arrikto | Google

User interface (UI)

kfctl CLI → `kfctl apply -V -f ${CONFIG_URI}`

kubectl CLI → `kubectl -n kubeflow get all`

APIs and SDKs

Interacting with Kubeflow

Arrikto | Google

User interface (UI)

kfctl CLI

kubectl CLI

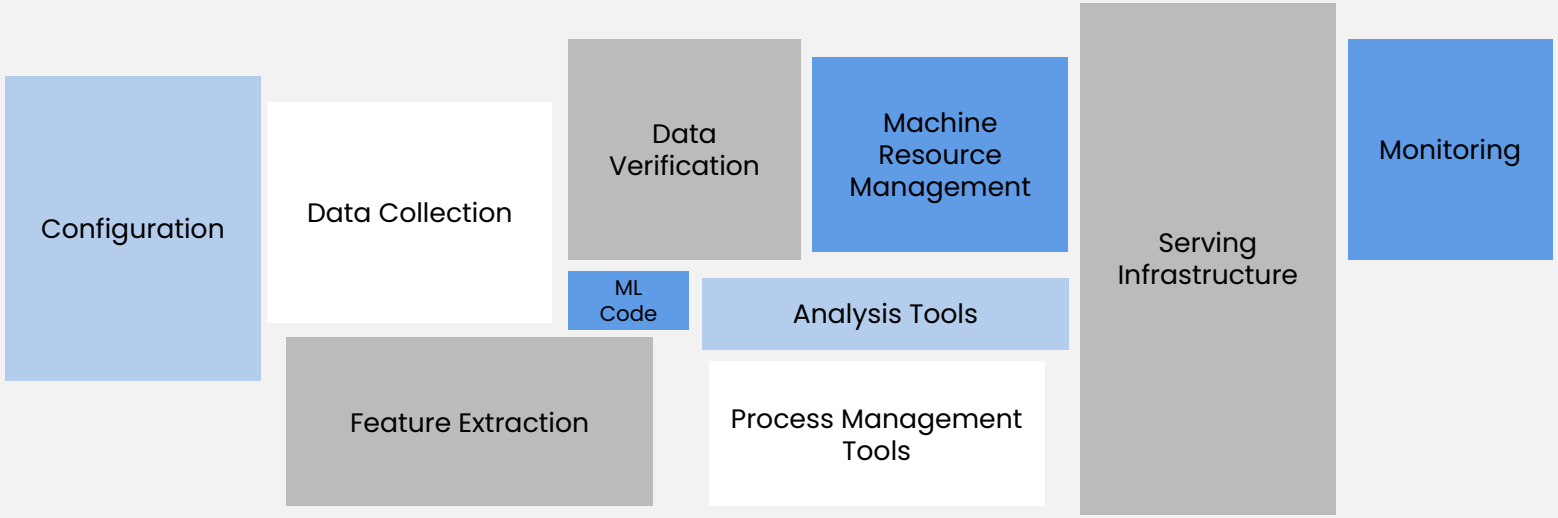
APIs and SDKs →

Examples:

- Pipelines SDK
- Katib API
- Metadata SDK

ML Applications are Distributed Systems

Arrikto | Google



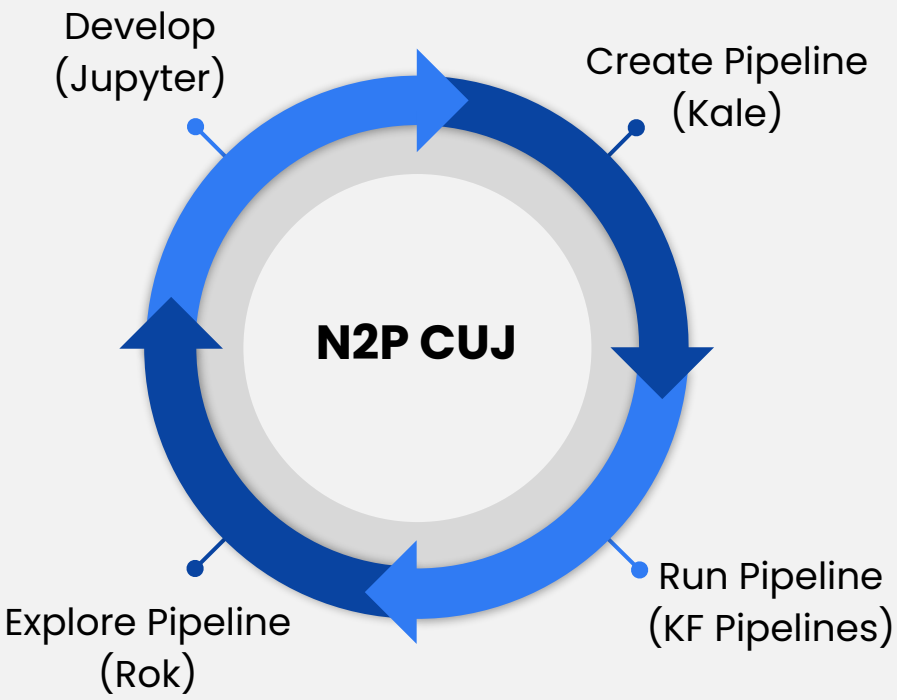
Credit: Hidden Technical Debt of Machine Learning Systems, D. Sculley, et al.

CI/CD for ML

Arrikto | Google

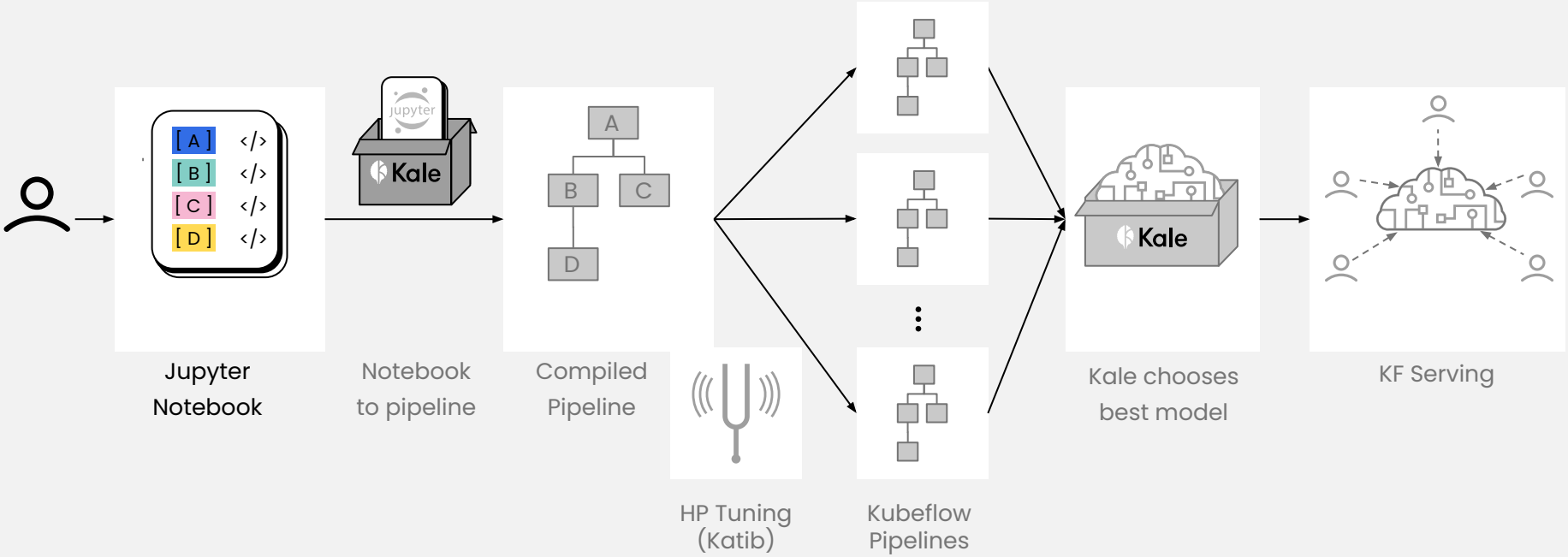
How can data scientists continually improve and validate models?

- Develop models and pipelines in Jupyter
- Convert notebook to pipeline using Kale
- Run pipeline using Kubeflow Pipelines
- Explore and debug pipelines using Rok



Data Science Workflows

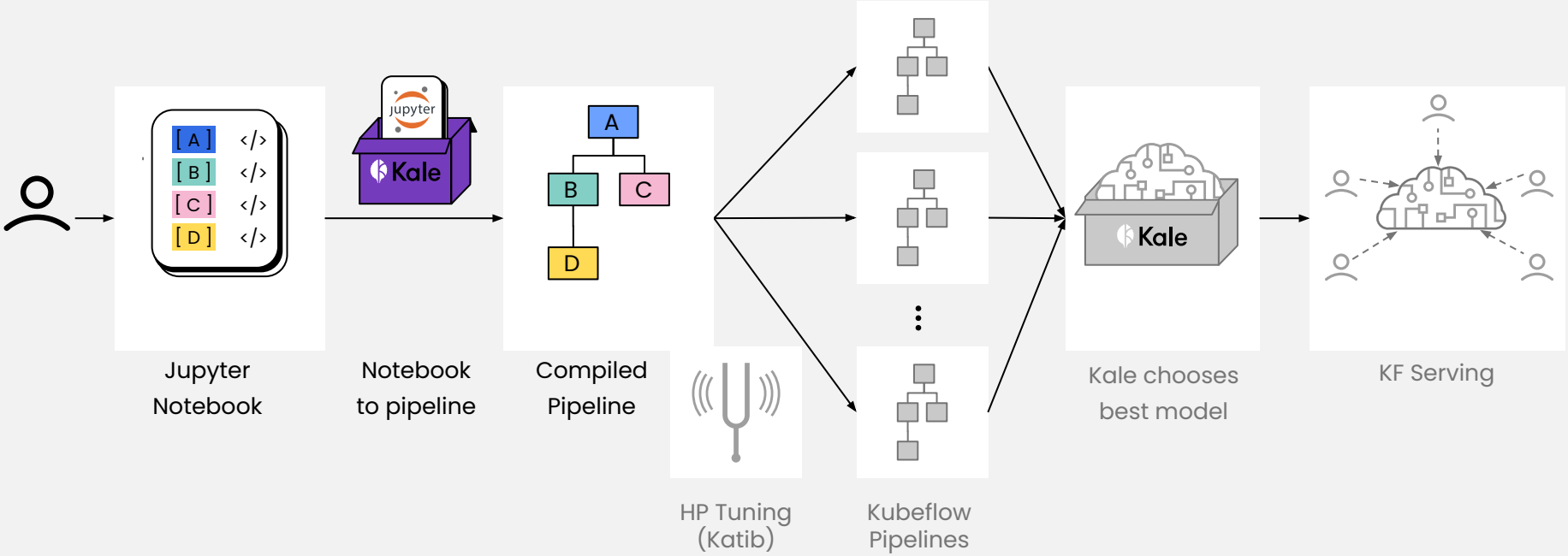
Arrikto | Google



A Step 1
B Step 2
C Step 3
D Step 4

Data Science Workflows

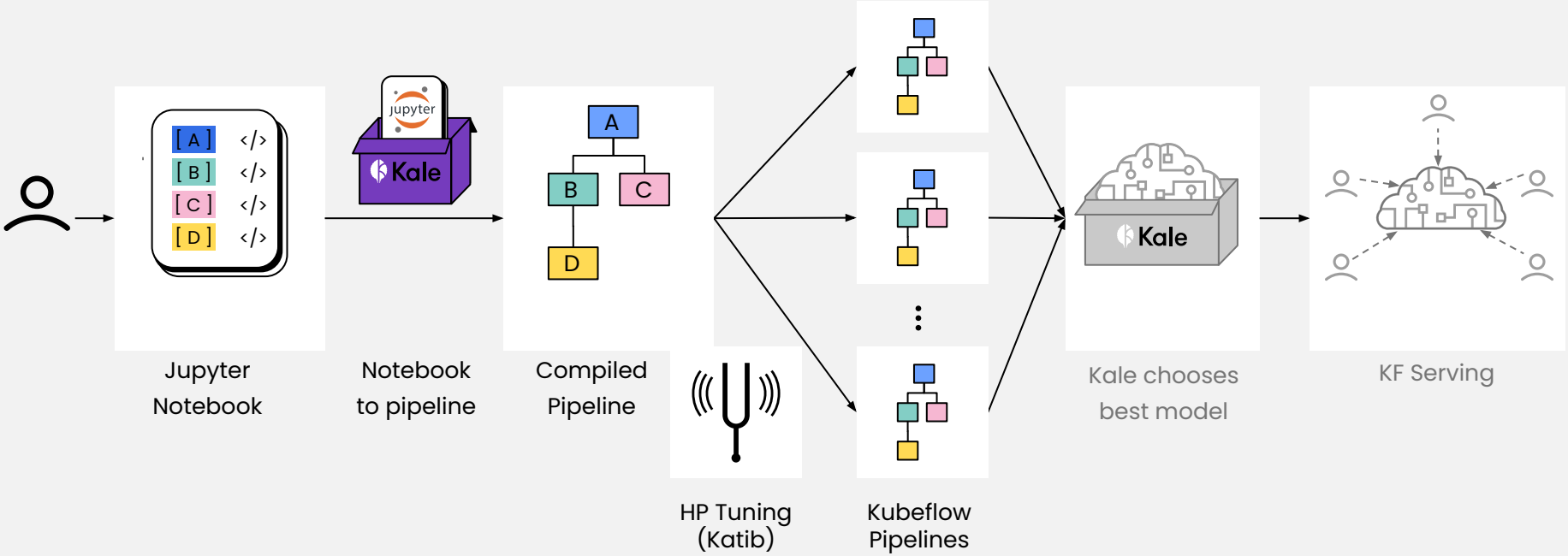
Arrikto | Google



A Step 1
B Step 2
C Step 3
D Step 4

Data Science Workflows

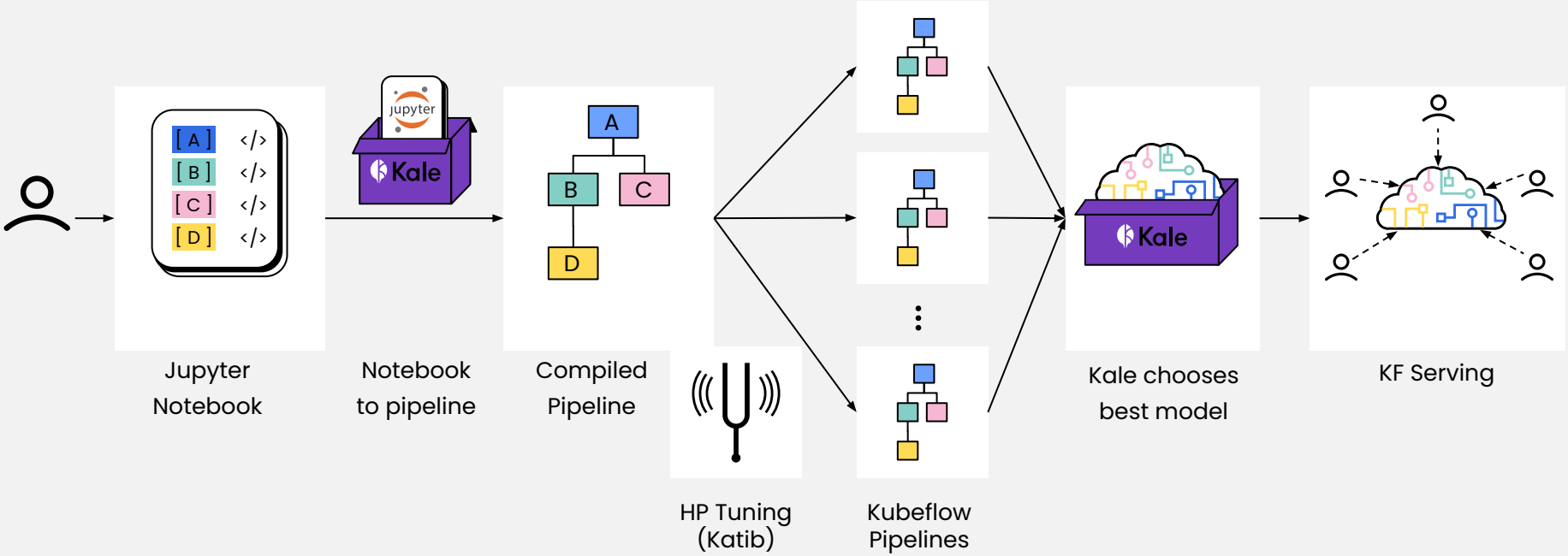
Arrikto | Google



A Step 1
B Step 2
C Step 3
D Step 4

Data Science Workflows

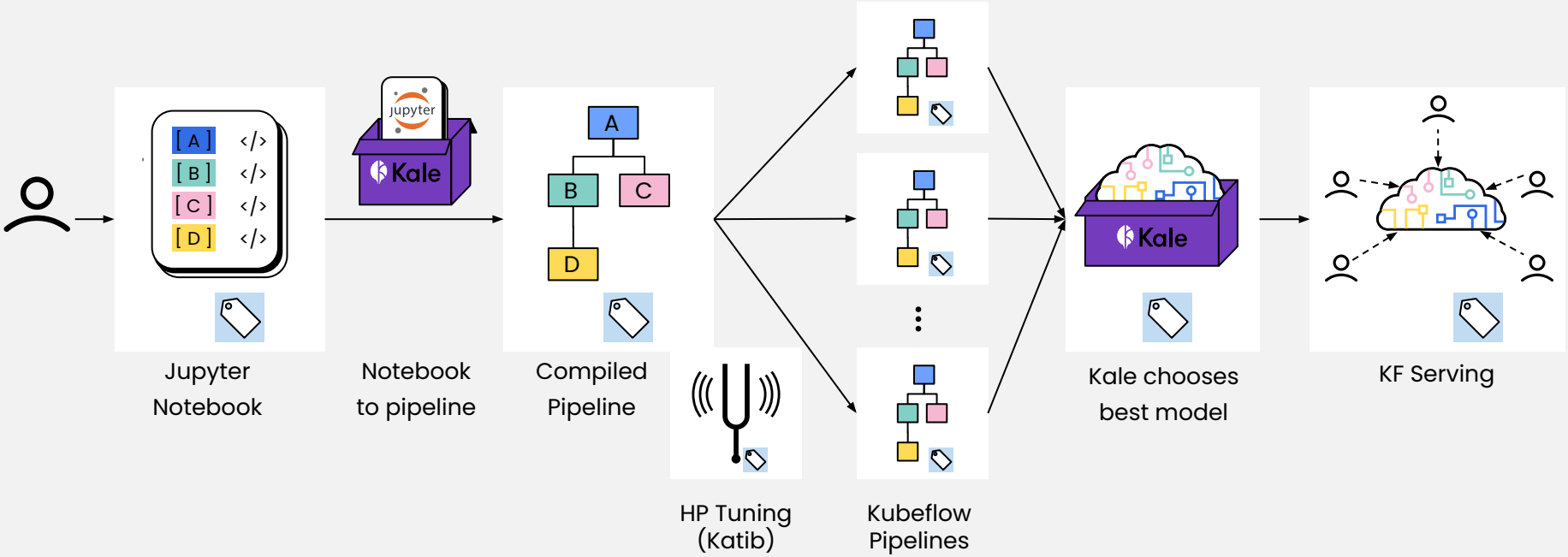
Arrikto | Google



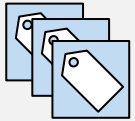
A Step 1
B Step 2
C Step 3
D Step 4

Data Science Workflows

Arrikto | Google



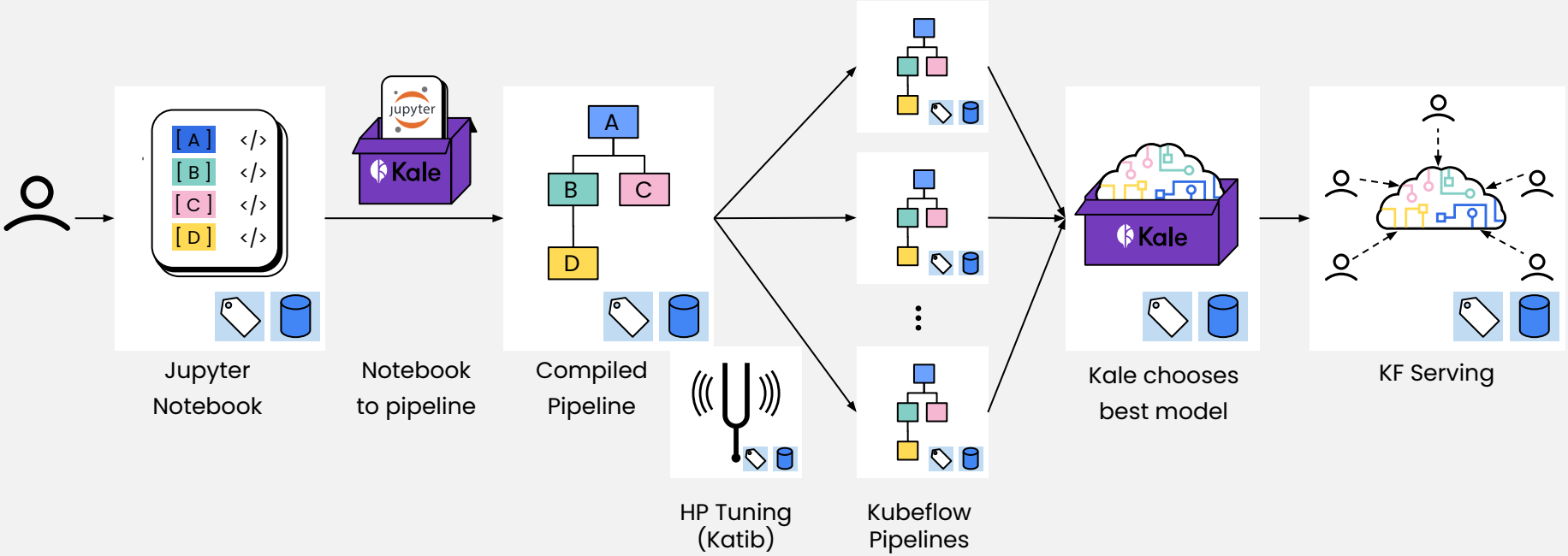
A Step 1
B Step 2
C Step 3
D Step 4



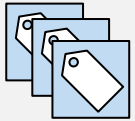
MLMD

Data Science Workflows

Arrikto | Google



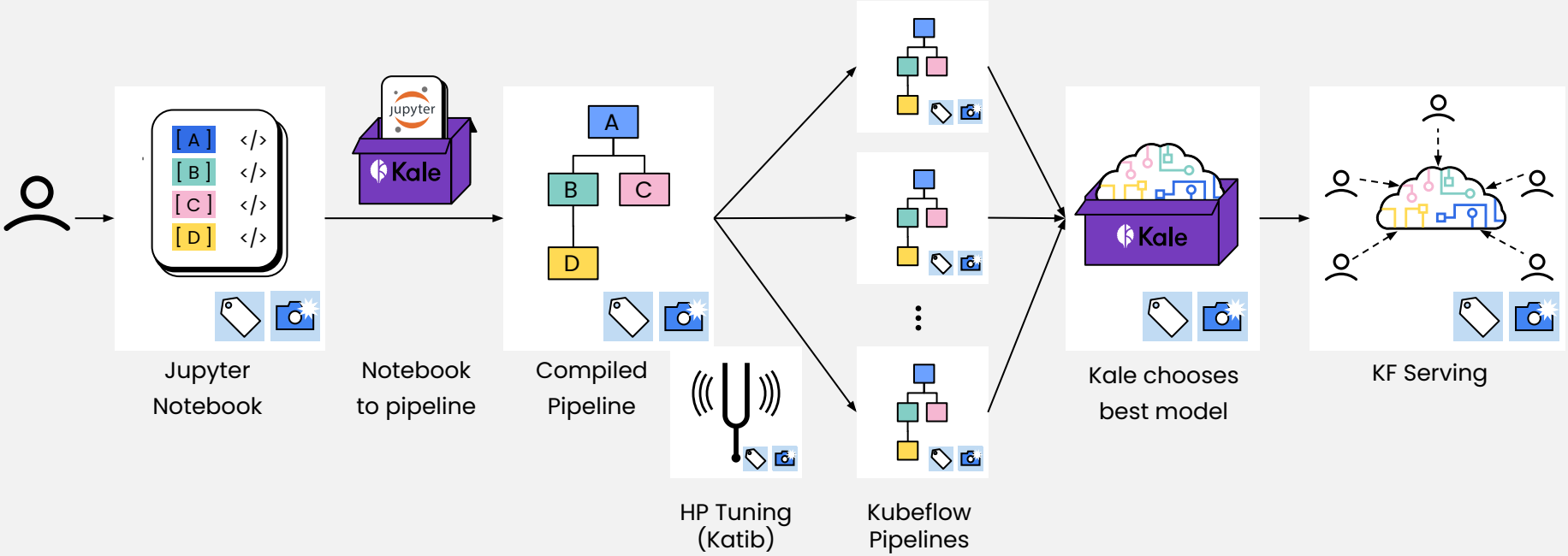
A Step 1
B Step 2
C Step 3
D Step 4



MLMD

Data Science Workflows

Arrikto | Google



A Step 1
B Step 2
C Step 3
D Step 4

MLMD

Rok
Data Management

Agenda

Arrikto | Google

Go to arrik.to/democ2p to find the Codelab with the step-by-step instructions for this tutorial

1. Install MiniKF
2. Notebook to Pipelines User Journey
3. Notebook to Katib User Journey
4. Notebook to Serving User Journey
5. Summary
6. Q&A

Agenda

Arrikto | Google

1 Install MiniKF

2 Notebook to Pipelines User Journey

3 Notebook to Katib User Journey

4 Notebook to Serving User Journey

5 Summary

6 Q&A

What is MiniKF

Arrikto | Google

- Kubeflow on GCP, your laptop, or on-prem infrastructure in just a few minutes
- All-in-one, single-node, Kubeflow distribution
- Very easy to spin up on your own environment on-prem or in the cloud
- MiniKF = MiniKube + Kubeflow + Arrikto's Rok Data Management Platform

Demo - Install MiniKF

Agenda

Arrikto | Google

1 Install MiniKF

2 Notebook to Pipelines User Journey

3 Notebook to Katib User Journey

4 Notebook to Serving User Journey

5 Summary

6 Q&A

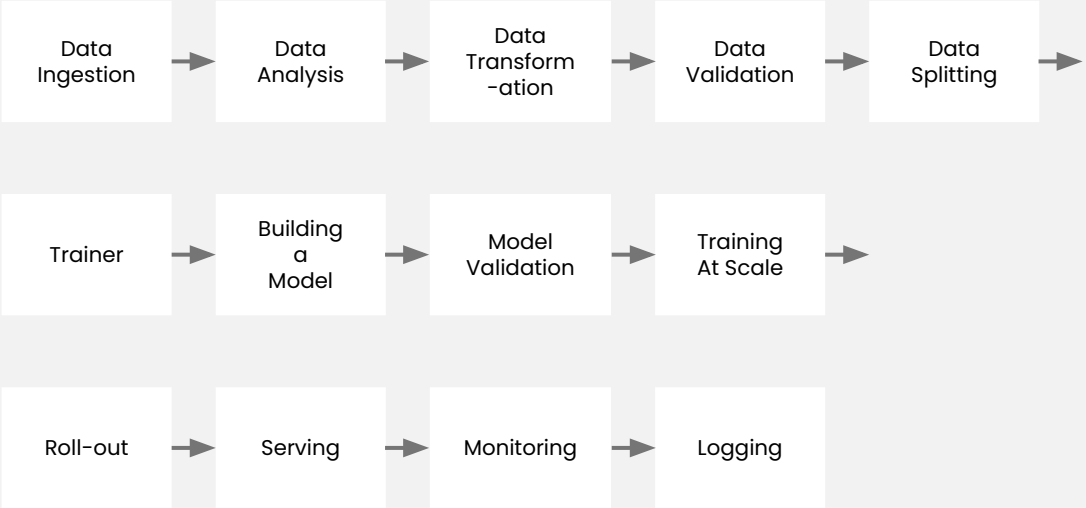
Data Science with Kubeflow

Arrikto | Google

Kubeflow Pipelines exists because Data Science and ML are inherently **pipeline processes**

This workshop will focus on two essential aspects:

- **Low barrier to entry:** deploy a Jupyter Notebook to Kubeflow Pipelines in the Cloud using a fully GUI-based approach
- **Reproducibility:** automatic data versioning to enable reproducibility and better collaboration between data scientists



Data Science with Kubeflow

Arrikto | Google

Kubeflow Pipelines exists because Data Science and ML are inherently **pipeline processes**

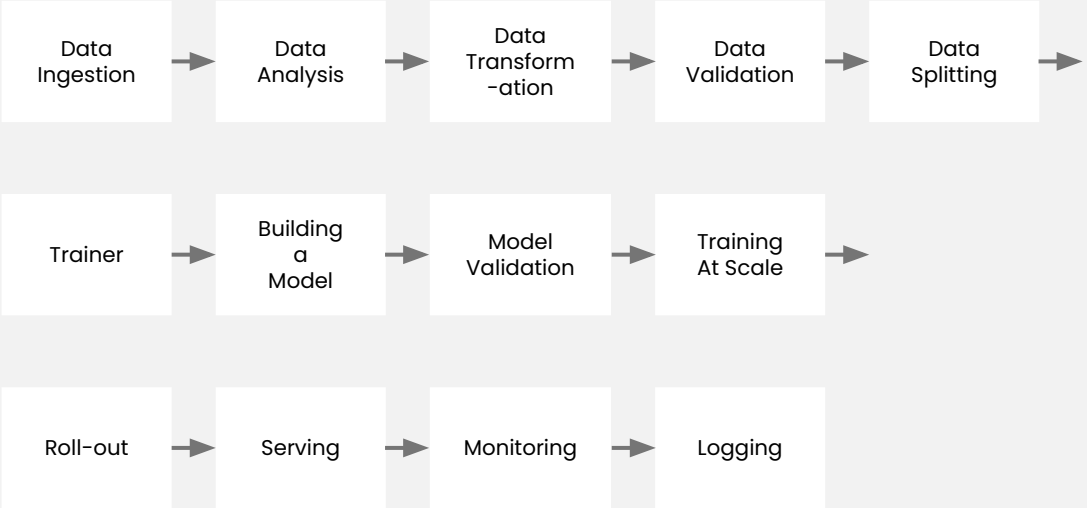
This workshop will focus on two essential aspects:

- **Low barrier to entry:** deploy a Jupyter Notebook to Kubeflow Pipelines in the Cloud using a fully GUI-based approach

  Kale

- **Reproducibility:** automatic data versioning to enable reproducibility and better collaboration between data scientists

  Arrikto



Benefits of running a Notebook as a Pipeline

Arrikto | Google

- The steps of the workflow are clearly defined
- Parallelization & isolation
  - Hyperparameter tuning
- Data versioning
- Different infrastructure requirements
  - Different hardware (GPU/CPU)

Workflow

Arrikto | Google

Before

Write your ML code

↓

Create Docker images

↓

Write DSL KFP code

↓

Compile DSL KFP

↓

Upload pipeline to KFP

↓

Run the Pipeline

Amend your ML code?

Workflow

Arrikto | Google

Before

Write your ML code

↓

Create Docker images

↓

Write DSL KFP code

↓

Compile DSL KFP

↓

Upload pipeline to KFP

↓

Run the Pipeline

Amend your ML code?

After

Write your ML code

↓

Tag your Notebook cells

↓

Run the Pipeline at the click of a button

Amend your ML code? → Just edit your Notebook!

Workflow

Arrikto | Google

Before

Write your ML code

↓

Create Docker images

↓

Write DSL KFP code

↓

Compile DSL KFP

↓

Upload pipeline to KFP

↓

Run the Pipeline

Amend your ML code?

After

Write your ML code

↓

Tag your Notebook cells

↓

Run the Pipeline at the click of a button

A Data Scientist can now reduce the time taken to write ML code and run a pipeline by 70%.

That means you can now run 3x as many experiments as you did before.

What that really means is that you can deliver work faster to the business and drive more revenue

Amend your ML code? → Just edit your Notebook!

Arrikto Rok

KDD 2017 Applied Data Science Paper

KDD'17, August 13–17, 2017, Halifax, NS, Canada

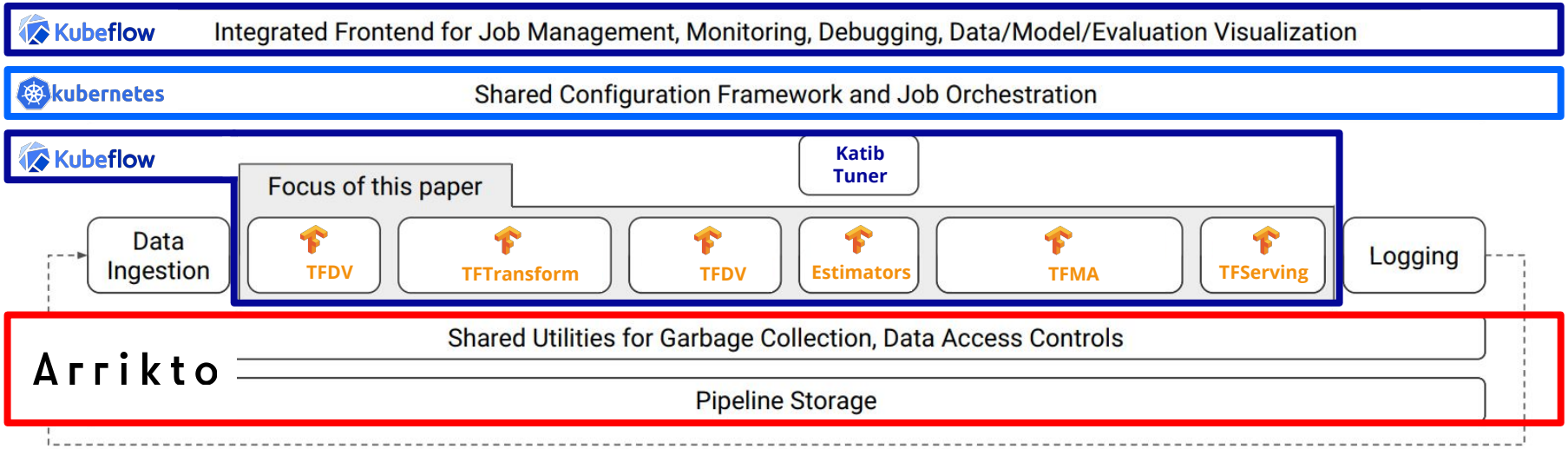


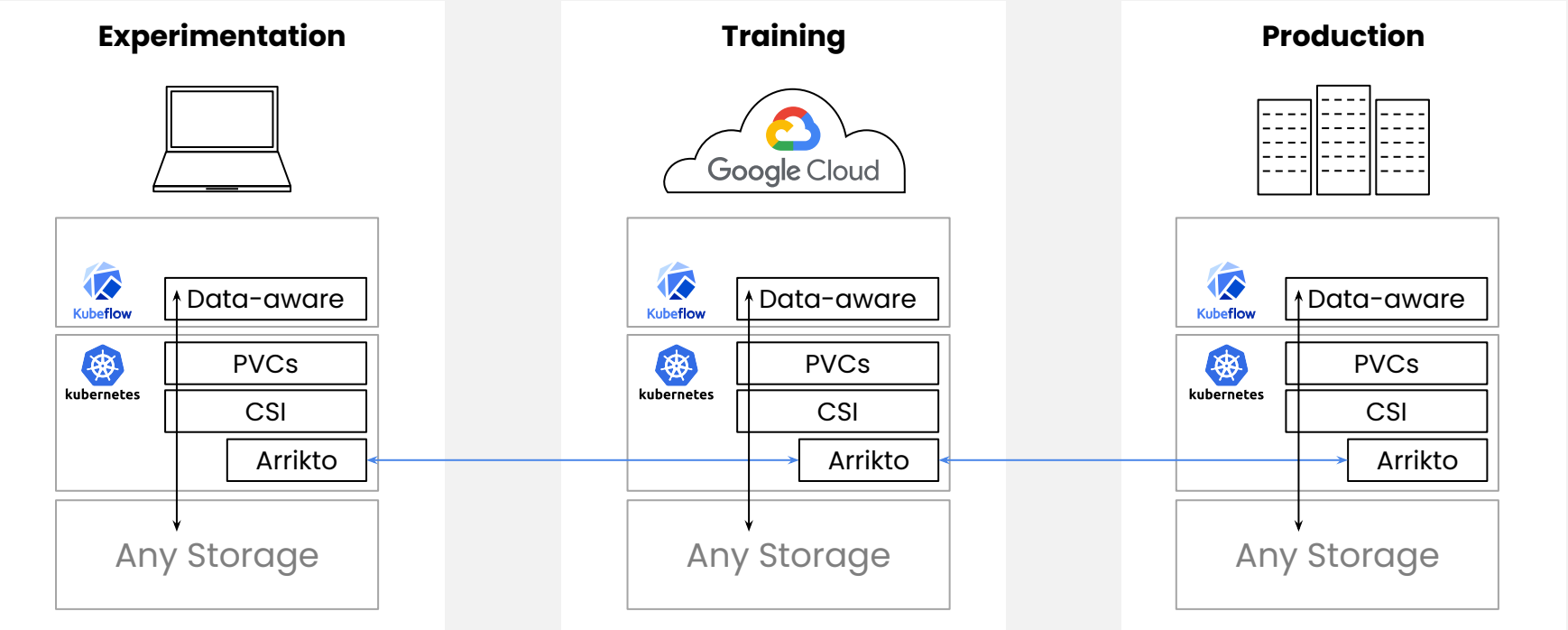
Figure 1: High-level component overview of a machine learning platform.

Arrikto Rok

Arrikto | Google

Data Versioning, Packaging, and Sharing

Across teams and cloud boundaries for complete Reproducibility, Provenance, and Portability



Demo - Notebook to Pipelines

Arrikto | Google

Agenda

Arrikto | Google

1 Install MiniKF

2 Notebook to Pipelines User Journey

3 Notebook to Katib User Journey

4 Notebook to Serving User Journey

5 Summary

6 Q&A

Hyperparameter optimization

Arrikto | Google

The two ways of life

- Change the parameters manually
- Use Katib

What is Katib

Arrikto | Google

Katib

Katib is a Kubernetes-based system for Hyperparameter Tuning and Neural Architecture Search. It supports a number of ML frameworks, including TensorFlow, Apache MXNet, PyTorch, XGBoost, and others.

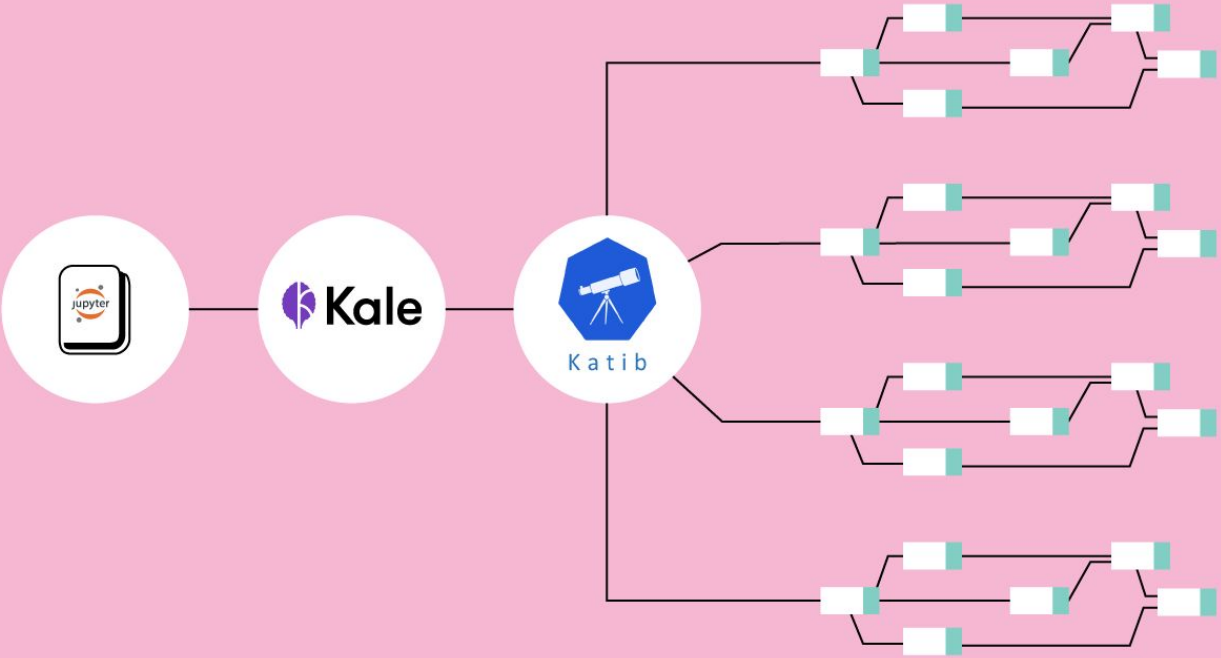
Hyperparameter optimization

Arrikto | Google

Combining the N2P CUJ with Katib

- Configure parameters, search algorithm, and objectives using a GUI
- Start HP tuning with the click of a button
- Reproducibility of every pipeline and every step
- Run Katib Trials as Pipelines
- Complete visibility of every different Katib Trial
- Caching for faster computation

Demo - Notebook to Katib



Agenda

Arrikto | Google

1 Install MiniKF

2 Notebook to Pipelines User Journey

3 Notebook to Katib User Journey

4 Notebook to Serving User Journey

5 Summary

6 Q&A

What is KFServing

Arrikto | Google

KFServing enables serverless inferencing on Kubernetes and provides performant, high abstraction interfaces for common machine learning (ML) frameworks like TensorFlow, XGBoost, scikit-learn, PyTorch, and ONNX to solve production model serving use cases.

Serving from a notebook

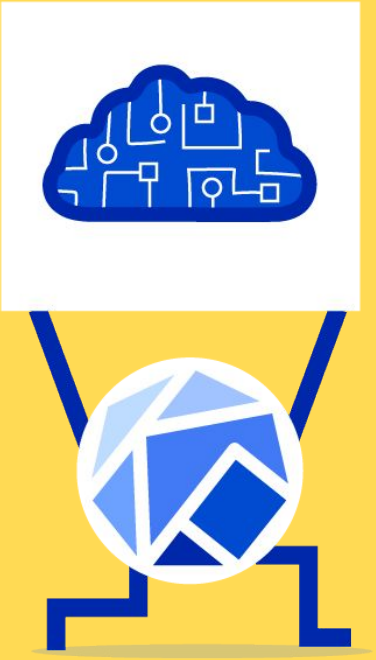
Arrikto | Google

Kale provides a simple to use API to serve a model

- Choose the best Trial of a HP Tuning experiment
- Restore a notebook from a Rok snapshot
- Create and deploy InferenceService CRs with a convenient API
- No need to build new Docker images
- Run predictions directly from the notebook

Demo - Notebook to Serving

Arrikto | Google



Agenda

Arrikto | Google

1 Install MiniKF

2 Notebook to Pipelines User Journey

3 Notebook to Katib User Journey

4 Notebook to Serving User Journey

5 Summary

6 Q&A

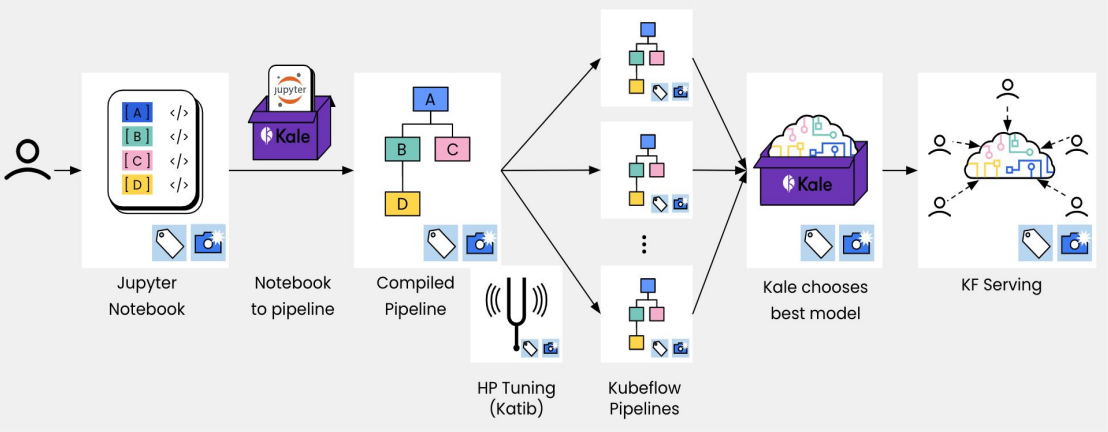
Summary

Arrikto | Google

What you have learned during this tutorial:

- Run a pipeline-based hyperparameter tuning workflow starting from your Jupyter Notebook
- Use Kale as a workflow tool to orchestrate Katib and Kubeflow Pipelines experiments
- Run and monitor model servers directly from your notebook
- Use the new and intuitive HP Tuning and Models UI

- Navigate between Kubeflow UIs, across linked entities tracked by MLMD
- **Simplify** your ML workflows using intuitive UIs
- Exploit the caching feature so that you ***accelerate*** your pipeline runs
- **Collaborate** faster and more easily, and have complete visibility of your training



Just a small sample of community contributions

Arrikto | Google

- Jupyter manager UI
- Pipelines volume support
- MiniKF
- Auth with Istio + Dex
- On-premise installation
- Linux Kernel

Community

Arrikto | Google

Kubeflow is open

- Open community
- Open design
- Open source
- Open to ideas

Get involved

- github.com/kubeflow
- kubeflow.slack.com
- @kubeflow
- kubeflow-discuss@googlegroups.com
- Community call on Tuesdays

Thank you, team!

Arrikto | Google

Ilias Katsakioris,
Dimitris Poulopoulos,
Kimonas Sotirchos,
Apostolos Plakias,
Konstantinos Palaiologos,
Chris Pavlou

Thank You

Arrikto | Google

Google

More Info
cloud.google.com

google

linkedin.com/in/karlweinmeister

Email Address:
kweinmeister@google.com

More Info
arrik.to/kubeconBOS

arrikto

linkedin.com/in/stefanofioravanzo

Email Address:
stefano@arrikto.com

Agenda

Arrikto | Google

1 Install MiniKF

2 Notebook to Pipelines User Journey

3 Notebook to Katib User Journey

4 Notebook to Serving User Journey

5 Summary

6 Q&A