

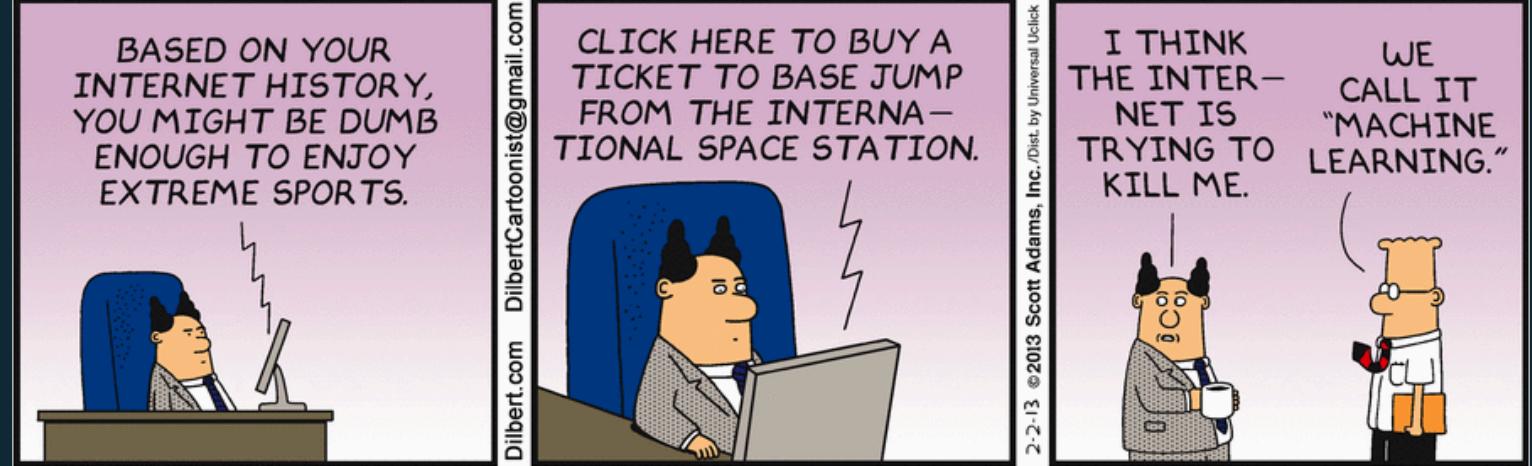


# Machine Learning using Kubeflow and Kubernetes



Chris Fregly, Developer Advocate, AI and Machine Learning  
Amazon Web Services  
@cfregly

*(Slides from Arun Gupta, @arungupta)*



<https://dilbert.com/strip/2013-02-02>

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



# Machine Learning is Hard!

Supervised/unsupervised/reinforcement learning ...

Data sourcing, cleanup, tagging & classification

Linear/logistic regression, Random forest, Decision tree, ...

Linear algebra, Statistics, Probability

TensorFlow, PyTorch, MXNet, Caffe2, Keras, SciKit-Learn, ...

Python, Julia, R, ...

Training and evaluating models

Distributed training

IntelliJ, VSCode, PyCharm, Jupyter notebook

Hyperparameter Tuning

GPU or CPU

MLOps

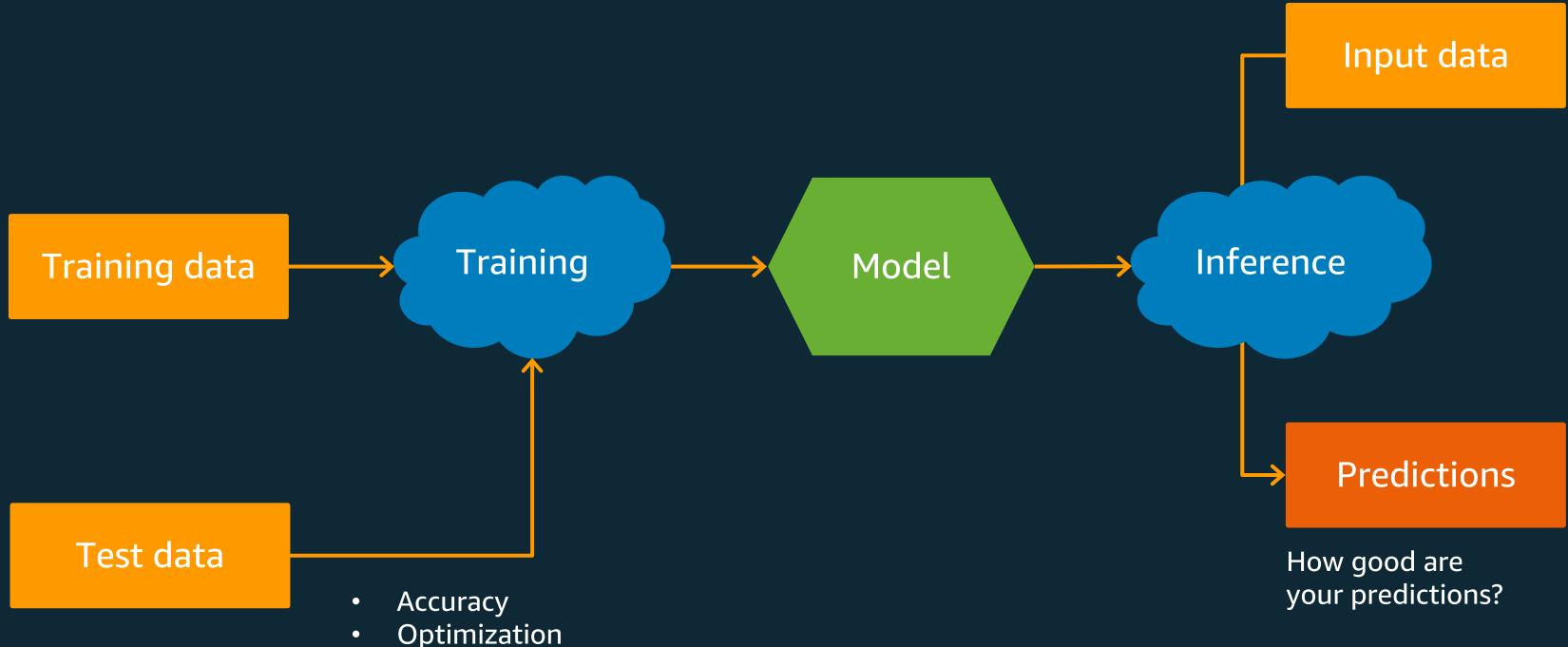


<https://happykaty.com/2018/05/15/drinking-from-a-fire-hose/>

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



# Machine Learning 101



# THE AWS ML STACK

Broadest and deepest set of capabilities

## ML Frameworks + Infrastructure

FRAMEWORKS	INTERFACES	INFRASTRUCTURE								
 TensorFlow  PYTORCH	 mxnet  Keras		 EC2 P3 & P3DN  EC2 G4 EC2 C5   DL CONTAINERS & AMIS	 ELASTIC CONTAINER SERVICE	 ELASTIC KUBERNETES SERVICE	 GREENGRASS	 ELASTIC INFERENCE	 INFERENTIA		

# THE AWS ML STACK

---

Broadest and deepest set of capabilities

## ML Services



# THE AWS ML STACK

Broadest and deepest set of capabilities

## AI Services

VISION	SPEECH	LANGUAGE	CHATBOTS	FORECASTING	RECOMMENDATIONS			
 REKOGNITION IMAGE	 REKOGNITION VIDEO	 TEXPTRACT	 POLLY	 TRANSCRIBE	 TRANSLATE COMPREHEND & COMPREHEND MEDICAL	 LEX	 FORECAST	 PERSONALIZE

# THE AWS ML STACK

Broadest and deepest set of capabilities

## AI Services

VISION	SPEECH	LANGUAGE	CHATBOTS	FORECASTING	RECOMMENDATIONS
 REKOGNITION IMAGE  REKOGNITION VIDEO  TTEXTRACT	 POLLY  TRANSCRIBE	 TRANSLATE COMPREHEND & COMPREHEND MEDICAL	 LEX	 FORECAST	 PERSONALIZE

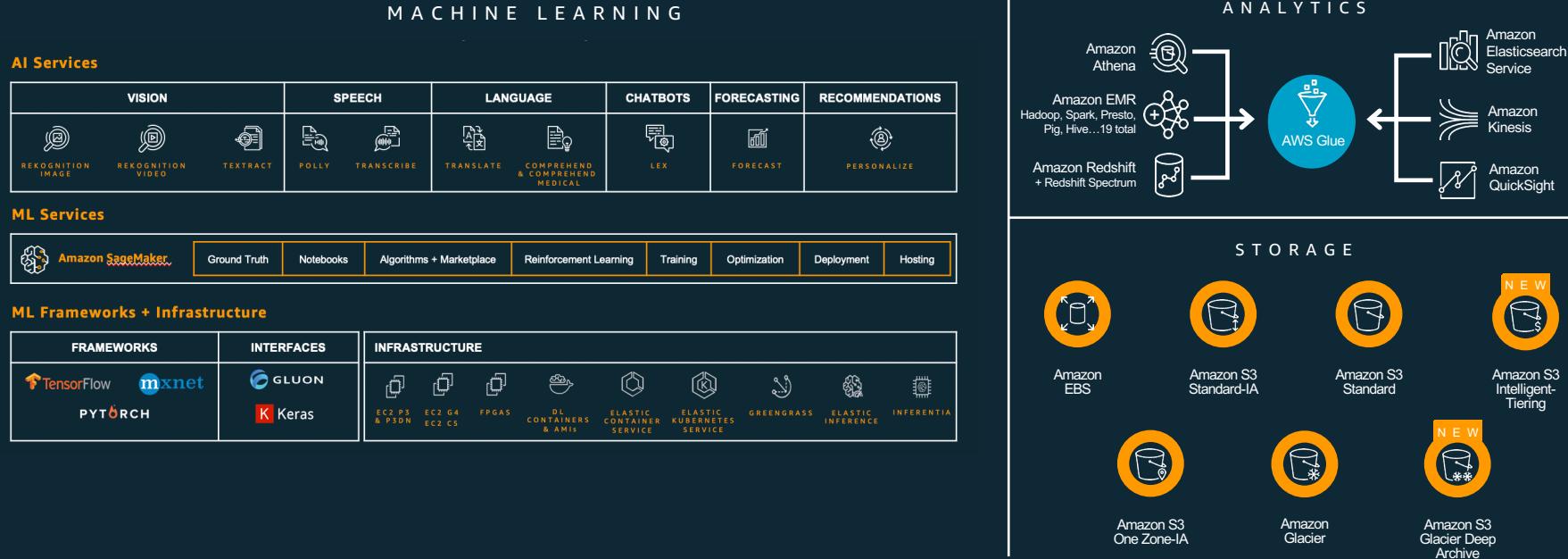
## ML Services

 Amazon SageMaker	Ground Truth	Notebooks	Algorithms + Marketplace	Reinforcement Learning	Training	Optimization	Deployment	Hosting
--	--------------	-----------	--------------------------	------------------------	----------	--------------	------------	---------

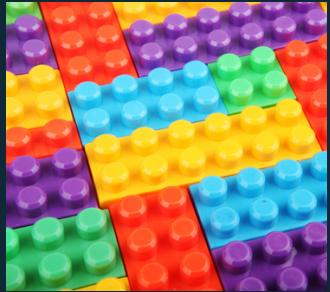
## ML Frameworks + Infrastructure

FRAMEWORKS	INTERFACES	INFRASTRUCTURE								
 TensorFlow  PYTORCH	 mxnet  Keras		 EC2 P3 & P3DN  EC2 C5	 FPGAs	 DL CONTAINERS & AMIS	 ELASTIC CONTAINER SERVICE	 ELASTIC KUBERNETES SERVICE	 GREENGRASS	 ELASTIC INFERENCE	 INFERENTIA

# Storage and Analytics for Machine Learning



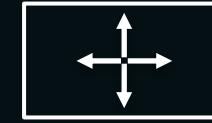
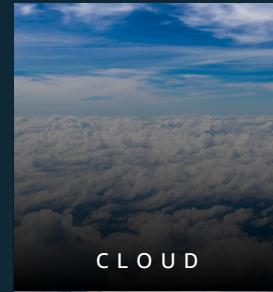
# Why Machine Learning on Kubernetes?



Composability

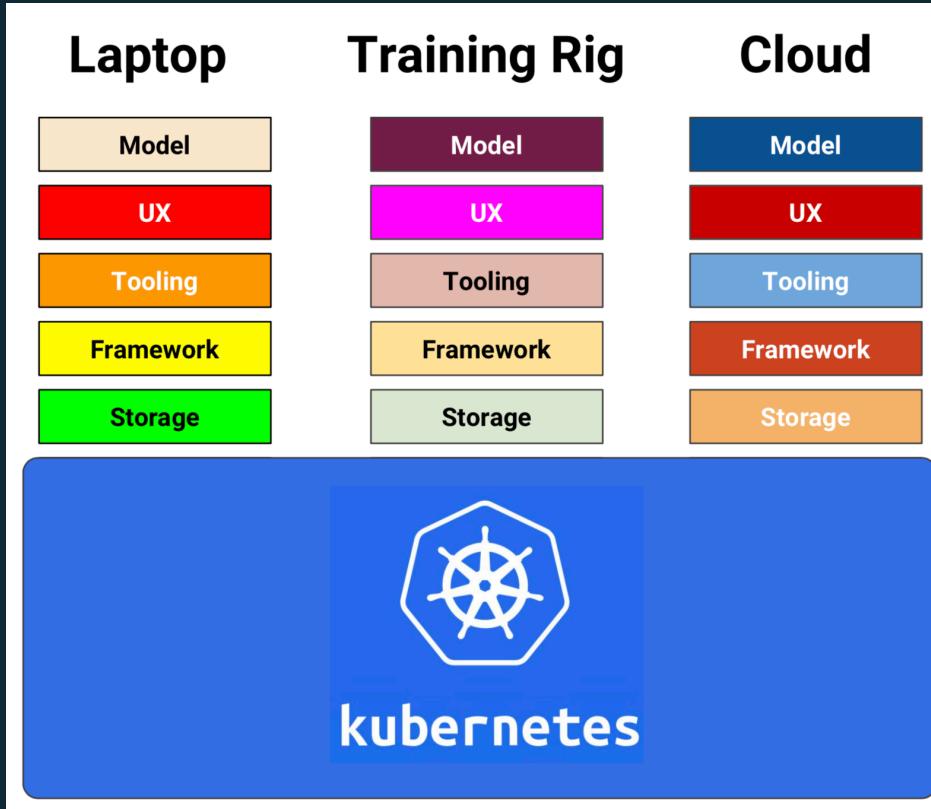


Portability



Scalability

# Machine Learning on K8s: Without KubeFlow

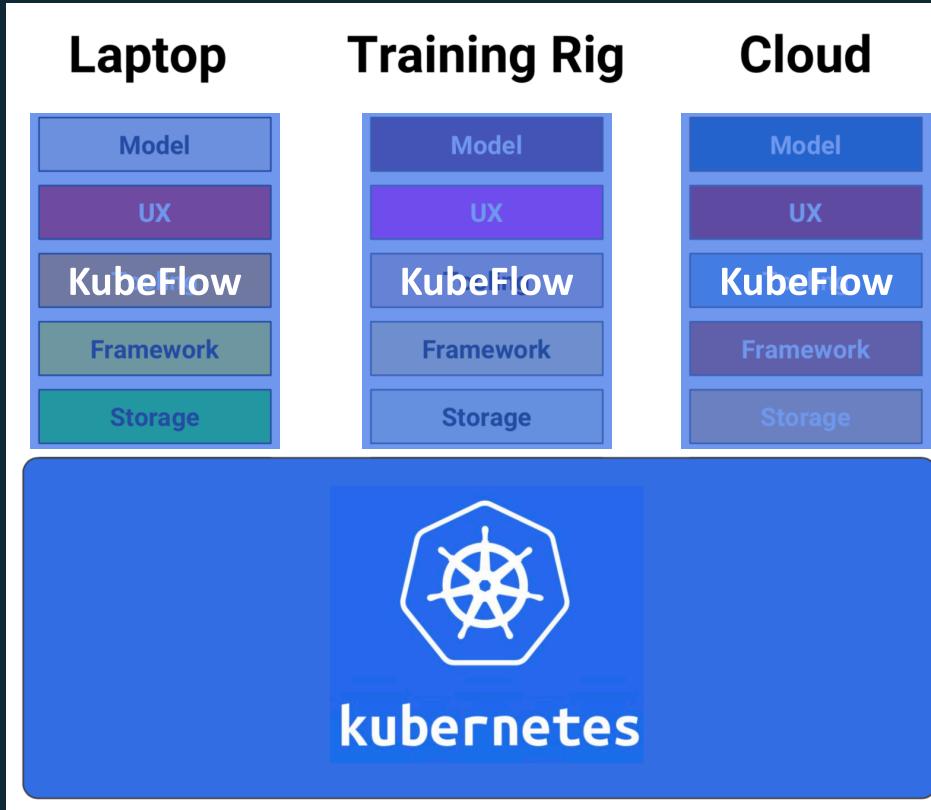


@aronchik

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



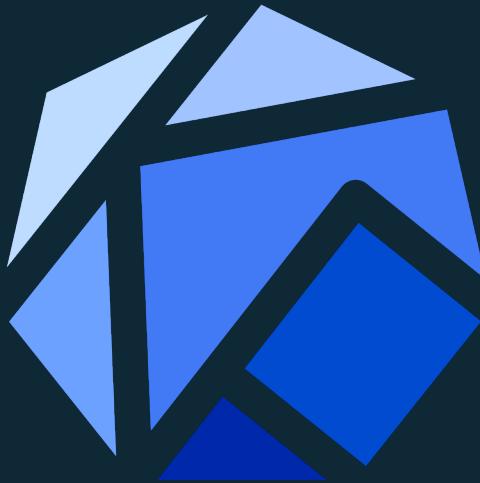
# Machine Learning on K8s: With KubeFlow



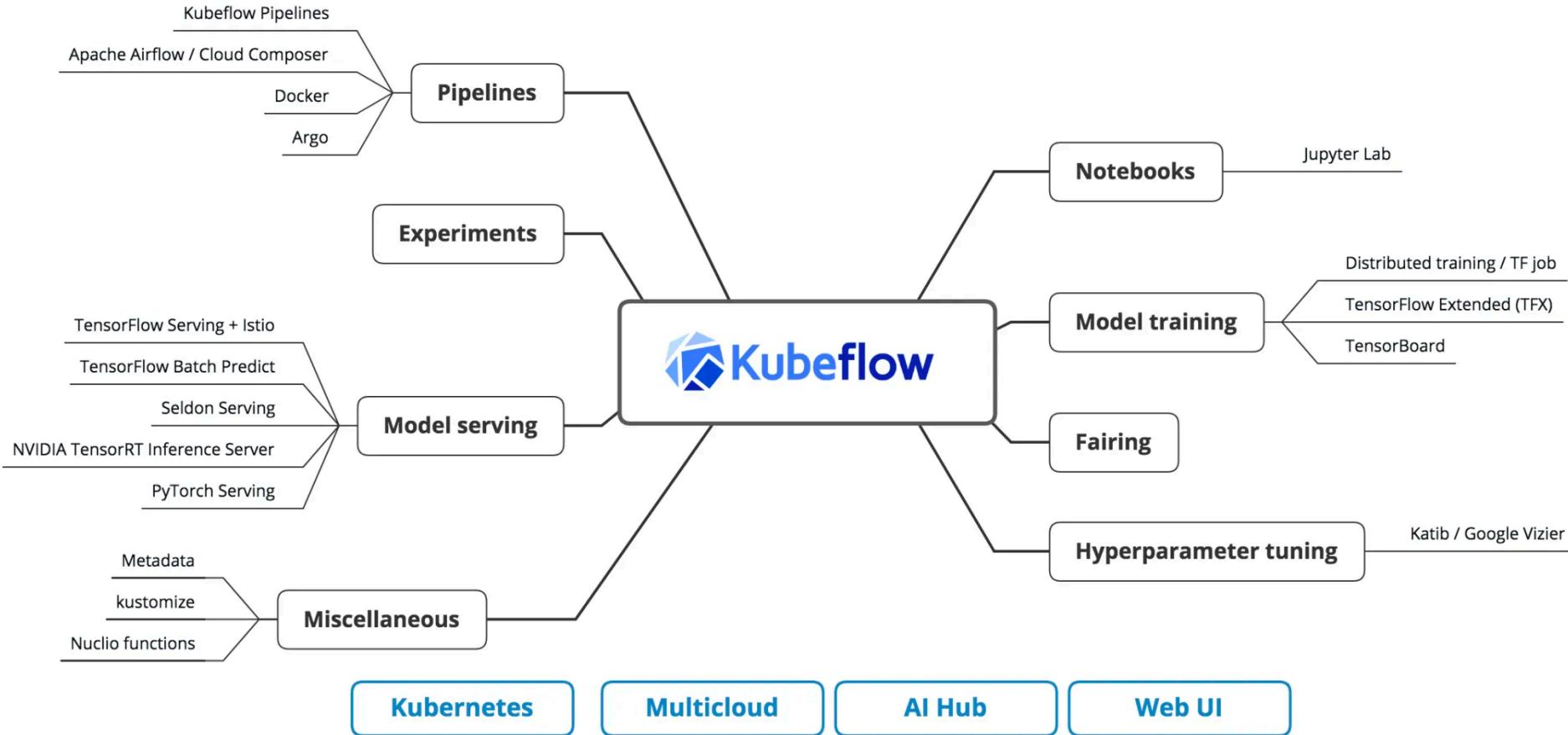
@aronchik

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.





# Kubeflow



# Amazon EKS: run Kubernetes in cloud

Managed Kubernetes control plane, attach data plane

---

Native upstream Kubernetes experience

---

Platform for enterprises to run production-grade workloads

---

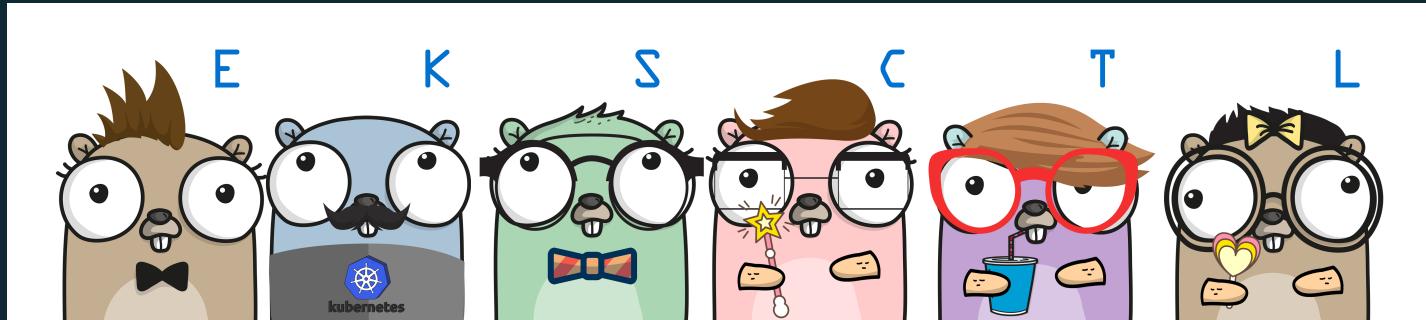
Integrates with additional AWS services

# Getting started with Amazon EKS

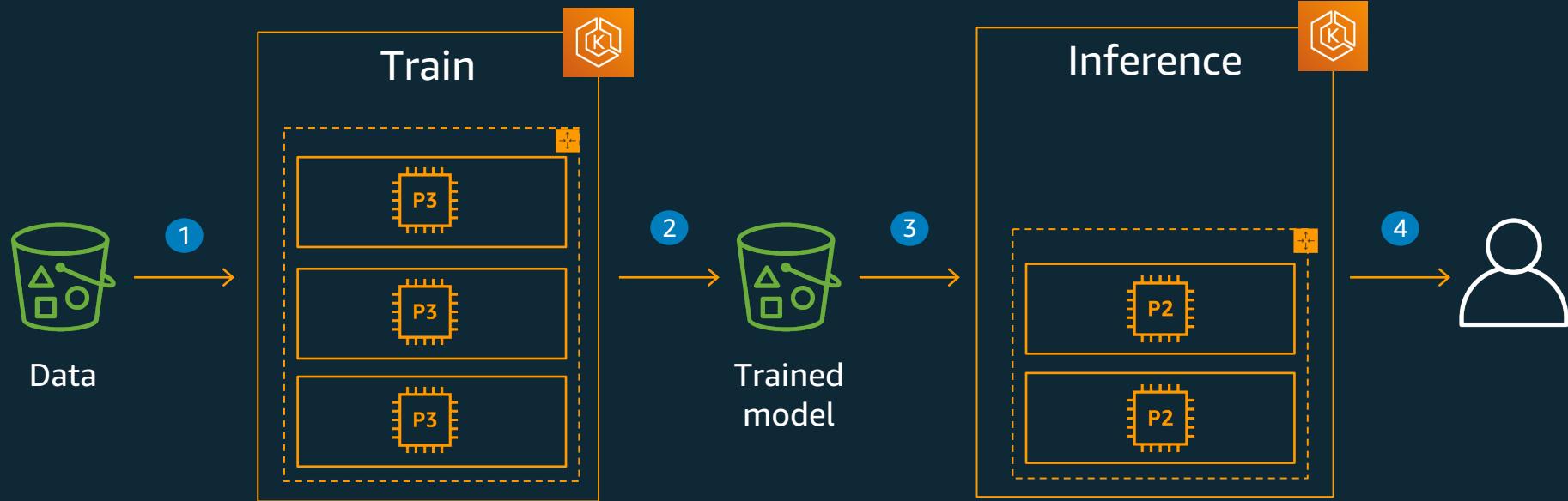
eksctl CLI—create Amazon EKS clusters ([eksctl.io](https://eksctl.io))

---

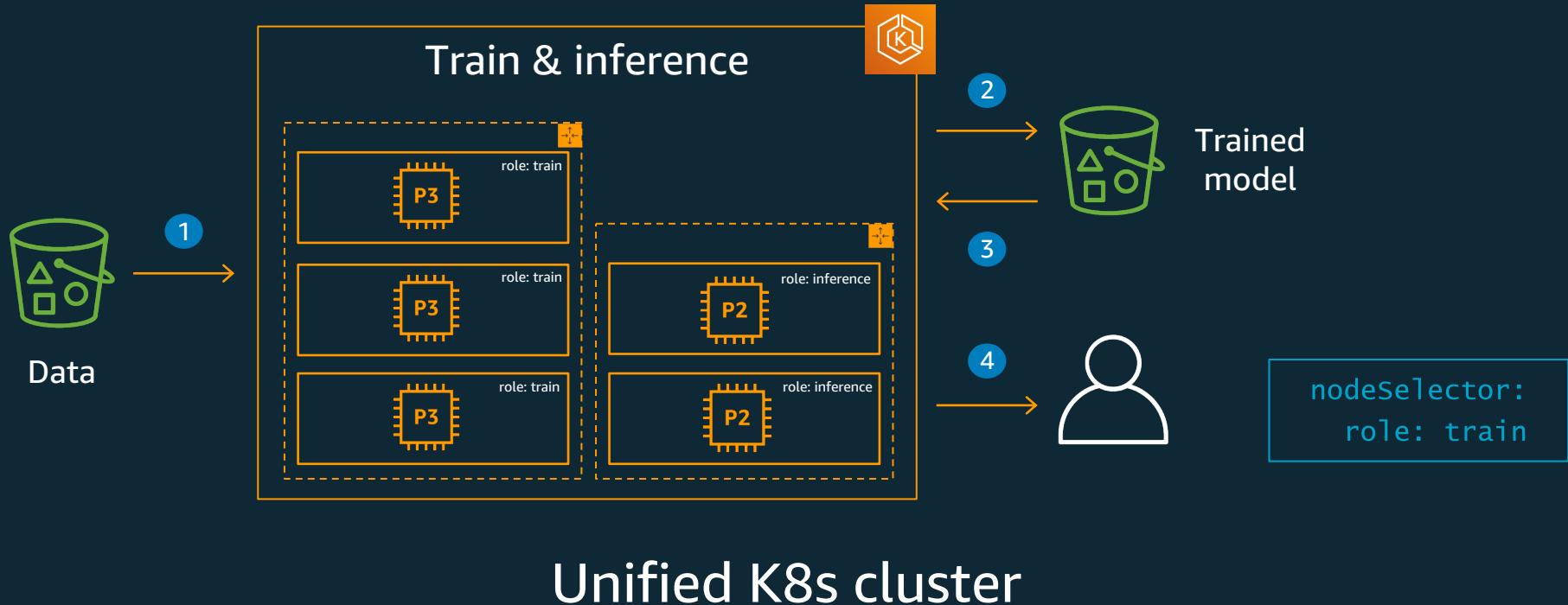
Creates all resources needed for the cluster



# Set up K8s for ML: Option 1

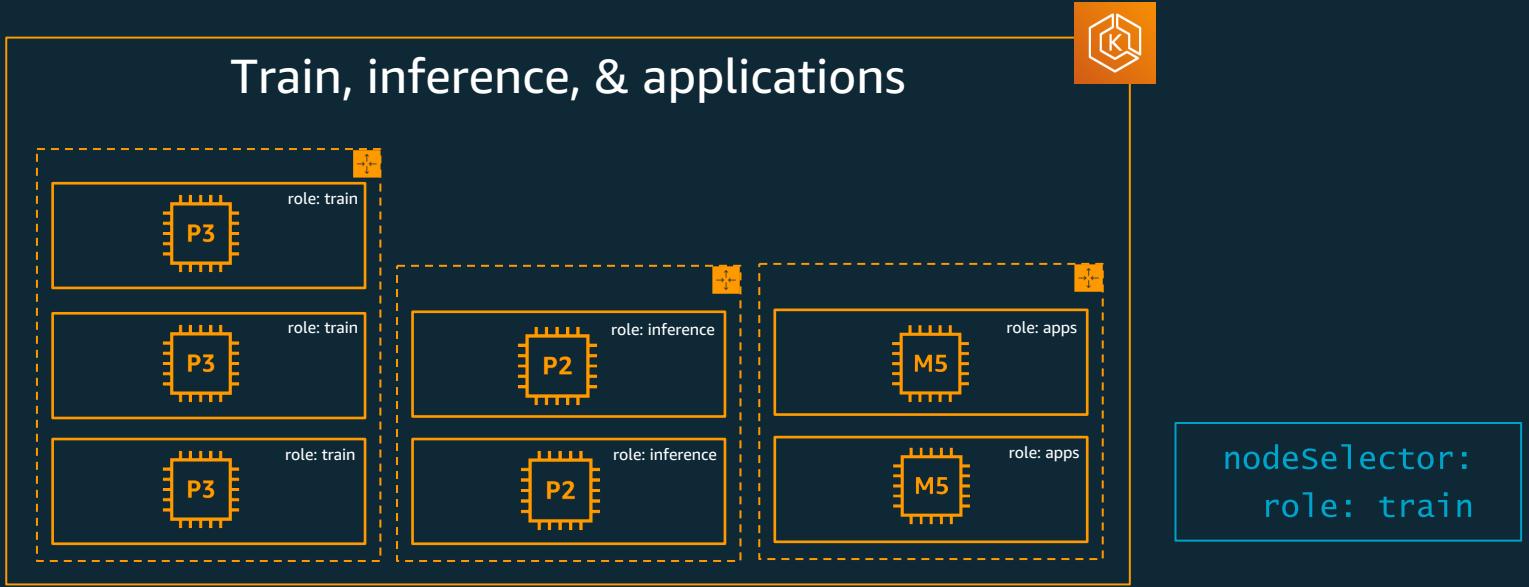


# Set up K8s for ML: Option 2a



Unified K8s cluster

# Set up K8s for ML: Option 2b



## Unified K8s cluster

# Scaling the cluster

## CLUSTER AUTOSCALER

Burst-able workloads

Aggressively move pods for utilization,  
can be configured for completion

Scale up based upon metrics

Takes over desired instance knob of auto-scaling group

Run them in same cluster with different node groups

## ESCALATOR

Batch or job-based workloads

Wait for the jobs to be completed

Aggressively scale up to reduce  
wait-time for pods

# Kubeflow Requirements

4 CPU, 12 GB memory, 50 GB storage

Kubernetes Versions	Kubeflow 0.4	Kubeflow 0.5	Kubeflow 0.6	Kubeflow 0.7
1.11	<b>compatible</b>	<b>compatible</b>	incompatible	incompatible
1.12	<b>compatible</b>	<b>compatible</b>	incompatible	incompatible
1.13	<b>compatible</b>	<b>compatible</b>	incompatible	incompatible
1.14	<b>compatible</b>	<b>compatible</b>	<b>compatible</b>	<b>compatible</b>
1.15	incompatible	<b>compatible</b>	<b>compatible</b>	<b>compatible</b>
1.16	incompatible	incompatible	incompatible	incompatible

[kubeflow.org/docs/started/k8s/overview/](https://kubeflow.org/docs/started/k8s/overview/)

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



# Kubeflow on Desktop

**MiniKF:** Local Kubeflow deployment using VirtualBox and Vagrant

- Minikube -> Kubernetes
- MiniKF -> Kubeflow (includes minikube)

Runs on macOS, Linux, and Windows

Does not require k8s-specific knowledge

# Kubeflow on Cloud

## Major cloud providers supported Choices on Amazon Web Services

- Self-managed k8s on EC2: Kops, CloudFormation, Terraform
- Amazon EKS

# Getting Started with Kubeflow on Amazon EKS



Kubeflow

# Jupyter Notebook



Web application to build, deploy, and train ML models

Create and share documents that contain live code,  
equations, visualizations, and narrative text

40+ programming languages

Use cases

- data cleaning and transformation

- numerical simulation

- data visualization

- machine learning

# Training using Jupyter Notebook



Kubeflow

# Kubeflow Fairing

Python SDK to build, train, and deploy ML models remotely

## Goals:

- Easily package ML training jobs
- Train ML models in the cloud
- Streamline the process of deploying a trained model

## Setup Kubeflow Fairing for training and prediction

```
from kubeflow import fairing
from kubeflow.fairing import TrainJob
from kubeflow.fairing.backends import KubeflowAWSBackend

from kubeflow import fairing

FAIRING_BACKEND = 'KubeflowAWSBackend'

AWS_ACCOUNT_ID = fairing.cloud.aws.guess_account_id()
AWS_REGION = 'us-west-2'
DOCKER_REGISTRY = '{}.dkr.ecr.{}.amazonaws.com'.format(AWS_ACCOUNT_ID, AWS_REGION)
S3_BUCKET = 'kubeflow-pipeline-data'

import importlib

if FAIRING_BACKEND == 'KubeflowAWSBackend':
    from kubeflow.fairing.builders.cluster.s3_context import S3ContextSource
    BuildContext = S3ContextSource(
        aws_account=AWS_ACCOUNT_ID, region=AWS_REGION,
        bucket_name=S3_BUCKET
    )

BackendClass = getattr(importlib.import_module('kubeflow.fairing.backends'), FAIRING_BACKEND)
```

[https://github.com/aws-samples/eks-kubeflow-workshop/blob/master/notebooks/02\\_Fairing/02\\_06\\_fairing\\_e2e.ipynb](https://github.com/aws-samples/eks-kubeflow-workshop/blob/master/notebooks/02_Fairing/02_06_fairing_e2e.ipynb)

## Train an XGBoost model remotely on Kubeflow

```
from kubeflow.fairing import TrainJob
train_job = TrainJob(HousingServe, input_files=['ames_dataset/train.csv', "requirements.txt"],
                     docker_registry=DOCKER_REGISTRY,
                     backend=BackendClass(build_context_source=BuildContext))
train_job.submit()
```

## Deploy the trained model to Kubeflow for prediction

```
from kubeflow.fairing import PredictionEndpoint
endpoint = PredictionEndpoint(HousingServe, input_files=['trained_ames_model.dat', "requirements.txt"],
                               docker_registry=DOCKER_REGISTRY,
                               service_type='ClusterIP',
                               backend=BackendClass(build_context_source=BuildContext))
endpoint.create()
```

## Call the prediction endpoint

```
# Wait service a while to be ready
# Example: !nc -vz fairing-service-srwh2.kubeflow-anonymous.svc.cluster.local 5000
!nc -vz <endpoint> 5000

# Get sample data and query endpoint
(train_X, train_y), (test_X, test_y) = read_input("ames_dataset/train.csv")

# PR https://github.com/kubeflow/fairing/pull/376
# Add `:5000/predict` to mitigate the issue.
endpoint.url='http://fairing-service-n8qv2.kubeflow-anonymous.svc.cluster.local:5000/predict'

endpoint.predict_nparray(test_X)
```

# Hyperparameter Tuning using Katib

**Hyperparameter** are parameters external to the model to control the training

e.g. Learning rate, batch size, epochs

**Tuning** finds a set of hyperparameters that optimizes an objective function

e.g. Find the optimal batch size and learning rate to maximize prediction accuracy

Katib enables hyperparameter tuning in Kubeflow  
Credits @Richard Liu @Johnu (Kubeflow slack)

# Katib Concepts

## Extensible

Framework agnostic: TensorFlow, PyTorch, MXNet, ...

Customizable algorithm backend

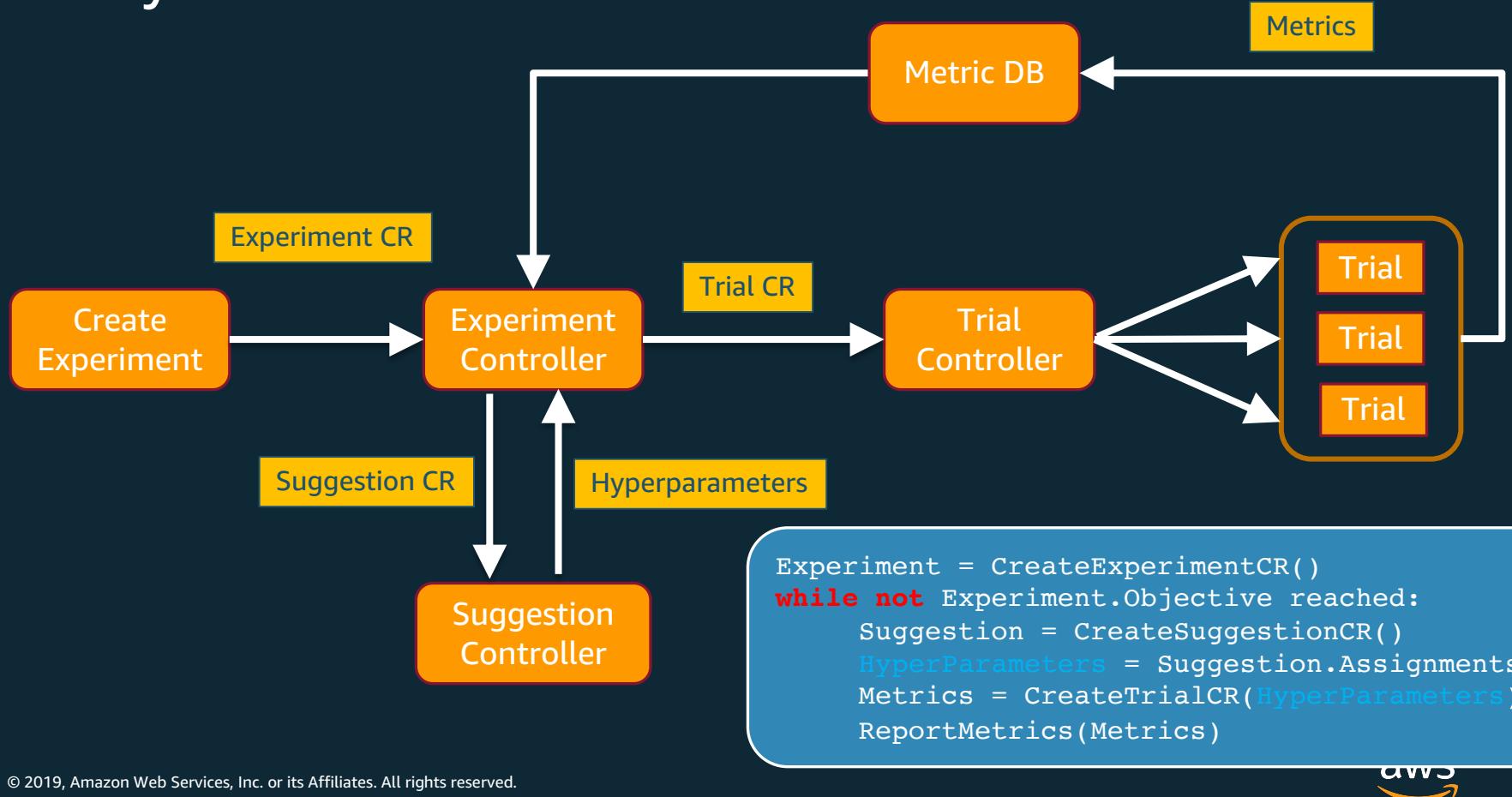
**Experiment:** “optimization loop” for some specific problem

**Suggestion:** a proposed solution to the problem

**Trial:** one iteration of the loop

**Job:** evaluate a trial and calculate objective value

# Katib System Architecture



## Experiment CR

```
2   kind: Experiment
3   metadata:
4     labels:
5       controller-tools.k8s.io: "1.0"
6     name: random-example
7   spec:
8     objective:
9       type: maximize
10      goal: 0.99
11      objectiveMetricName: Validation-accuracy
12      additionalMetricNames:
13        - accuracy
14     algorithm:
15       algorithmName: random
16     parallelTrialCount: 3
17     maxTrialCount: 12
18     maxFailedTrialCount: 3
```

## Hyperparameters

```
19   parameters:
20     - name: --lr
21       parameterType: double
22       feasibleSpace:
23         min: "0.01"
24         max: "0.03"
25     - name: --num-layers
26       parameterType: int
27       feasibleSpace:
28         min: "2"
29         max: "5"
30     - name: --optimizer
31       parameterType: categorical
32       feasibleSpace:
33         list:
34           - sgd
35           - adam
36           - ftl
```

[https://github.com/aws-samples/eks-kubeflow-workshop/blob/master/notebooks/08\\_Hyperparameter\\_Tuning/random-search-example.yaml](https://github.com/aws-samples/eks-kubeflow-workshop/blob/master/notebooks/08_Hyperparameter_Tuning/random-search-example.yaml)

## Trial template

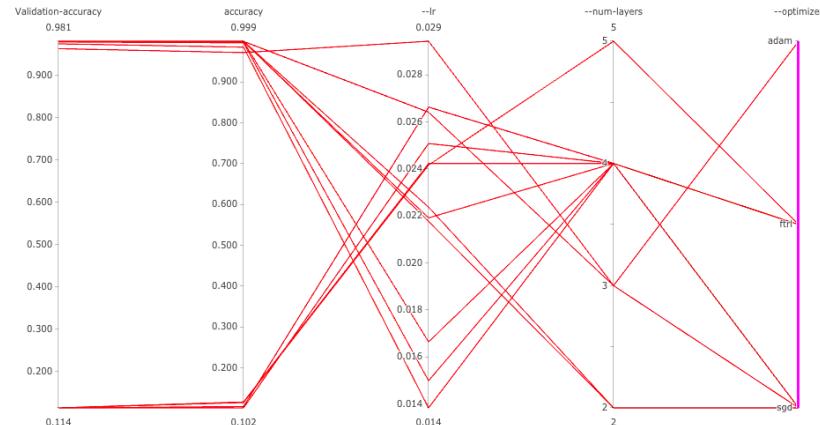
```
37     trialTemplate:  
38         goTemplate:  
39             rawTemplate: |-  
40                 apiVersion: batch/v1  
41                 kind: Job  
42                 metadata:  
43                     name: {{.Trial}}  
44                     namespace: {{.NameSpace}}  
45             spec:  
46                 template:  
47                     spec:  
48                         containers:  
49                             - name: {{.Trial}}  
50                               image: docker.io/katib/mxnet-mnist-example  
51                               command:  
52                                   - "python"  
53                                   - "/mxnet/example/image-classification/train_mnist.py"  
54                                   - "--batch-size=64"  
55                                   {{- with .HyperParameters}}  
56                                   {{- range .}}  
57                                   - "{{.Name}}={{.Value}}"  
58                                   {{- end}}  
59                                   {{- end}}  
60             restartPolicy: Never
```



## ☰ Katib

BACK

Experiment Name: random-experiment



trialName	Validation-accuracy	accuracy	--lr	--num-layers	--optimizer
random-experiment-rfwbnsd	0.974920	0.984844	0.013831565266960293	4	sgd
random-experiment-rfwbnsd	0.974920	0.984844	0.013831565266960293	4	sgd

# Kubeflow KFServing

Simple and pluggable platform for **ML inference**

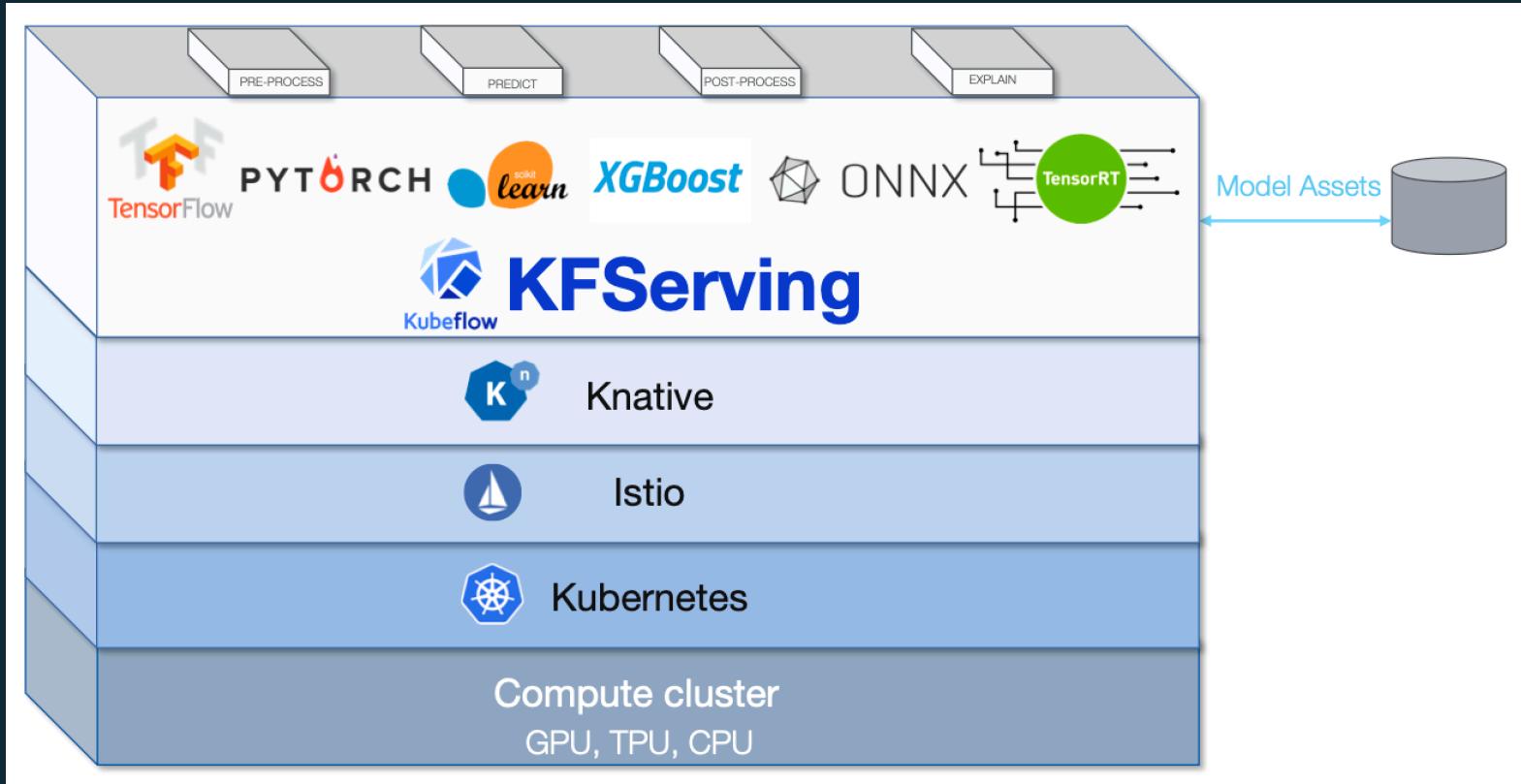
Intuitive and consistent experience

Serving models on arbitrary frameworks

e.g. TensorFlow, XGBoost, SciKitLearn

Encapsulates GPU auto-scaling, canary rollouts

Credits @ellis-bigelow (Kubeflow slack)



# KFServing Custom Resource

```
1 apiVersion: v1
2 kind: Secret
3 metadata:
4   name: mysecret
5 annotations:
6     serving.kubeflow.org/s3-endpoint: minio-service.kubeflow:9000 # repl
7     serving.kubeflow.org/s3-usehttps: "0" # by default 1, for testing wi
8 type: Opaque
9 data:
10    awsAccessKeyId: bWluaW8=
11    awsSecretAccessKey: bWluaW8xMjM=
12 ---
13 apiVersion: v1
14 kind: ServiceAccount
15 metadata:
16   name: sa
17 secrets:
18   - name: mysecret
```

```
1 apiVersion: "serving.kubeflow.org/v1alpha2"
2 kind: "InferenceService"
3 metadata:
4   name: "mnist-s3"
5 spec:
6   default:
7     predictor:
8       serviceAccountName: sa
9       tensorflow:
10        storageUri: "s3://mnist/v1/export"
```

S3 secret  
attached to  
Service  
Account

Trained  
model

[https://github.com/kubeflow/kfserving/blob/master/docs/samples/s3/tensorflow\\_s3.yaml](https://github.com/kubeflow/kfserving/blob/master/docs/samples/s3/tensorflow_s3.yaml)

# Pluggable Interface

```
apiVersion: "serving.kubeflow.org/v1alpha1"
kind: "InferenceService"
metadata:
  name: "sklearn-iris"
spec:
  default:
    sklearn:
      storageUri: "gs://kfserving-samples/models/sklearn/iris"
```

```
apiVersion: "serving.kubeflow.org/v1alpha1"
kind: "InferenceService"
metadata:
  name: "flowers-sample"
spec:
  default:
    tensorflow:
      storageUri: "gs://kfserving-samples/models/tensorflow/flowers"
```

```
apiVersion: "serving.kubeflow.org/v1alpha1"
kind: "KFService"
metadata:
  name: "pytorch-cifar10"
spec:
  default:
    pytorch:
      storageUri: "gs://kfserving-samples/models/pytorch/cifar10"
      modelClassName: "Net"
```



# KFServing Interface – Scikit Learn

```
apiVersion: "serving.kubeflow.org/v1alpha1"
kind: "KFService"
metadata:
  name: "sklearn-iris"
spec:
  default:
    sklearn:
      storageUri: "gs://kfserving-samples/models/sklearn/iris"
      serviceAccount: inferencing-robot
      minReplicas: 3
      maxReplicas: 10
      resources:
        requests:
          cpu: 2
          gpu: 1
          memory: 10Gi
  canaryTrafficPercent: 25
  canary:
    sklearn:
      storageUri: "gs://kfserving-samples/models/sklearn/iris-v2"
      serviceAccount: inferencing-robot
      minReplicas: 3
      maxReplicas: 10
      resources:
        requests:
          cpu: 2
          gpu: 1
          memory: 10Gi
```



# Distributed Training using Horovod



Created by Uber, hosted by LF AI Foundation

Distributed training framework for TensorFlow, Keras, PyTorch, and MXNet

Compared to distributed TensorFlow

- Far less code changes
- ~2x faster

Examples at Uber: Self-driving vehicles, fraud detection, and trip forecasting

Named after a traditional Russian dance

# Kubeflow Pipelines

Compose, deploy, and manage end-to-end ML workflows

- End-to-end orchestration
- Easy, rapid, and reliable experimentation
- Easy re-use

Built using Pipelines SDK

- `kfp.compiler`, `kfp.components`, `kfp.Client`

Uses Argo under the hood to orchestrate resources

# Kubeflow Pipelines Platform

**UI** for managing and tracking experiments, jobs, and runs

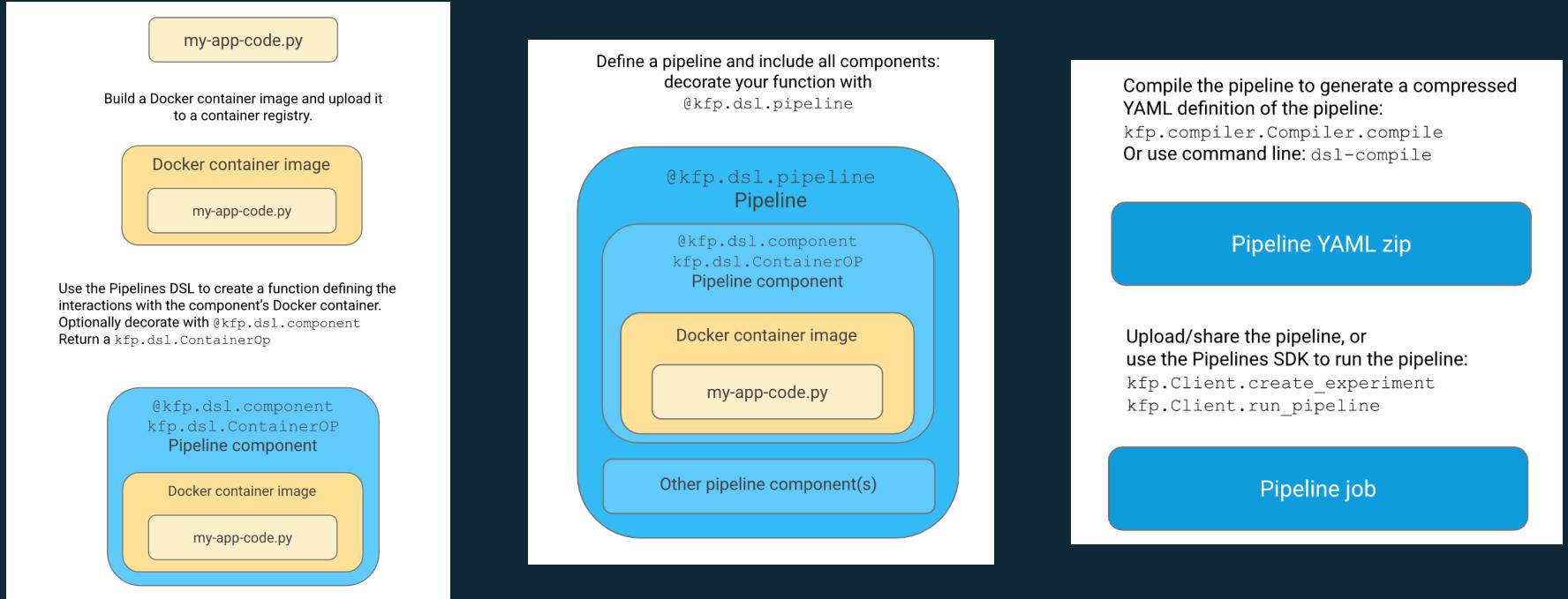
**Engine** for scheduling multi-step ML workflows

**SDK** for defining and manipulating pipelines and components

- [kubeflow-pipelines.readthedocs.io/en/latest/](https://kubeflow-pipelines.readthedocs.io/en/latest/)

**Notebooks** for interacting with the system using the SDK

# Creating Kubeflow Pipeline Components



# Consumer Loan Acceptance Scoring



## Objective

- Putting hundreds of data products live
- Single development -> deployment -> delivery environment
- First go batch, then real-time

## Analytics environment

- AWS
- High security and compliance with regulation

## Typical modeling context

- Structured data
- Supervised learning
- Internalizing interpretable models and hybrid pipelines

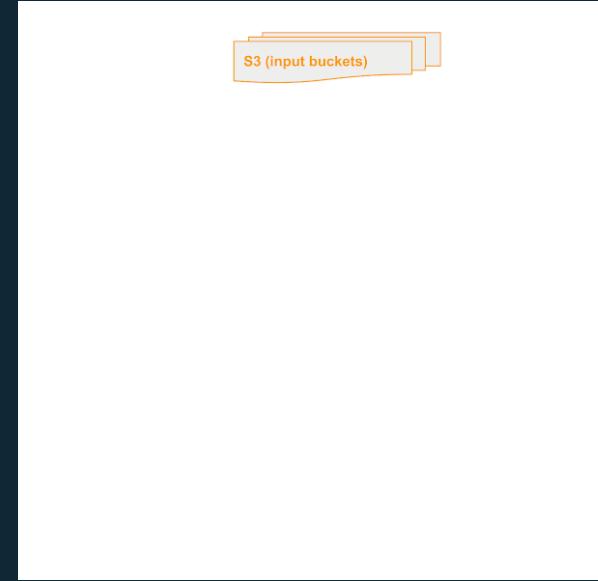
# Requirements

## Data Scientists

- Hybrid, integrated, cloud-based dev env
  - Python
  - PySpark (locally + remotely on Spark cluster)
  - R
  - SQL
- Version control (scripts & artifacts)

## ML DevOps

- Seamless deployment of hybrid pipelines
- Trigger-based scheduling & orchestration of runs
- Monitoring & dashboard
- Version control (runs & pipelines)



# Architecture

Custom notebook servers

Kubeflow 0.6  
Notebook dev environment  
Pipelines for dev & delivery

ElasticStack  
Dashboarding

Amazon EKS

Amazon ECR

AWS  
Infrastructure, connections, security  
S3, Spark cluster, VMs, ...

# Storage: FSx for Lustre

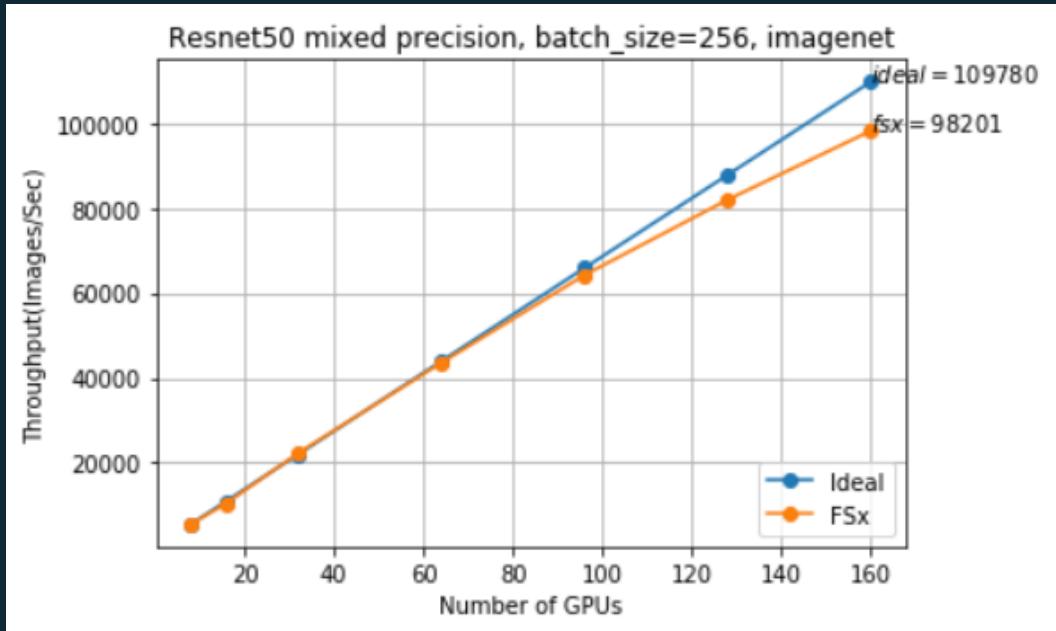
High performance file system for processing Amazon S3 or on-premises data

Low latency and high throughput

Works natively with Amazon S3

Container Storage Interface (CSI) driver

# Best Practices for Optimizing Distributed Deep Learning Performance on Amazon EKS



<https://aws.amazon.com/blogsopensource/optimizing-distributed-deep-learning-performance-amazon-eks/>

# Advantages of KubeFlow on AWS

[kubeflow.org/docs/aws](https://kubeflow.org/docs/aws)

EKS cluster provision with `eksctl`

---

External traffic with `AWS ALB Ingress Controller`

---

`Amazon FSx CSI driver` to manage Lustre file system

---

Centralized and unified K8s logs in `CloudWatch`

---

TLS and Auth with `AWS Certificate Manager` and `AWS Cognito`

---

Private access for your K8s API server endpoint

---

Detect GPU instance and install `Nvidia device plugin`

# Kubeflow components usage

Top 5 components:

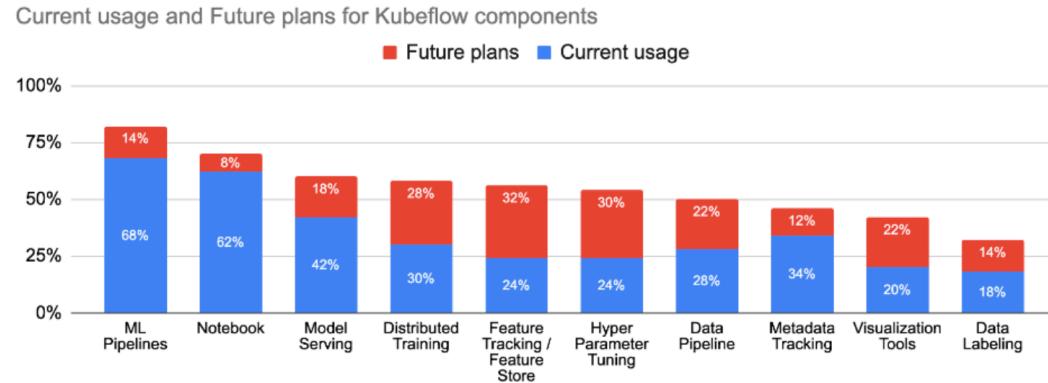
82% Pipelines

70% Notebooks

60% Model Serving

58% Distributed Training

56% Feature Tracking/Store



# Kanban Board

The Kanban board displays the status of 109 issues across three columns:

- To do (109 items):**
  - Add other libraries to tensorflow-notebook-image? (opened by Jeffwan, priority/p2)
  - upgrade model deployer component to remove ks (opened by SinaChavoshi, priority/p2)
  - Generate a link to Kubeflow Central Dashboard page with kfctl (opened by elviraux, priority/p3)
  - CSM integration (opened by lluunn, feature\_request, lifecycle/stale, platform/gcp)
- In progress (1 item):**
  - Prometheus/Telemetry metrics for Notebooks Manager UI (opened by abhi-g, area/jupyter, effort/2-weeks, kind/feature, priority/p1)
- Done (15 items):**
  - GCP credentials for notebook - new UI breaks user experience (opened by SinaChavoshi, area/jupyter, kind/feature, platform/gcp, priority/p1)
  - use a different service mesh (opened by rverma-nikhai, area/kustomize, kind/discussion, kind/feature, platform/aws, priority/p2)
  - Kubeflow identity management API to beta (opened by jlewi, area/enterprise\_re..., kind/feature, priority/p0)
  - Need confirmation on kubeflow on kubernetes 1.16 (opened by cyrilthank, area/testing, kind/feature, priority/p1, problems/k8s)

[github.com/orgs/kubeflow/projects/25](https://github.com/orgs/kubeflow/projects/25)

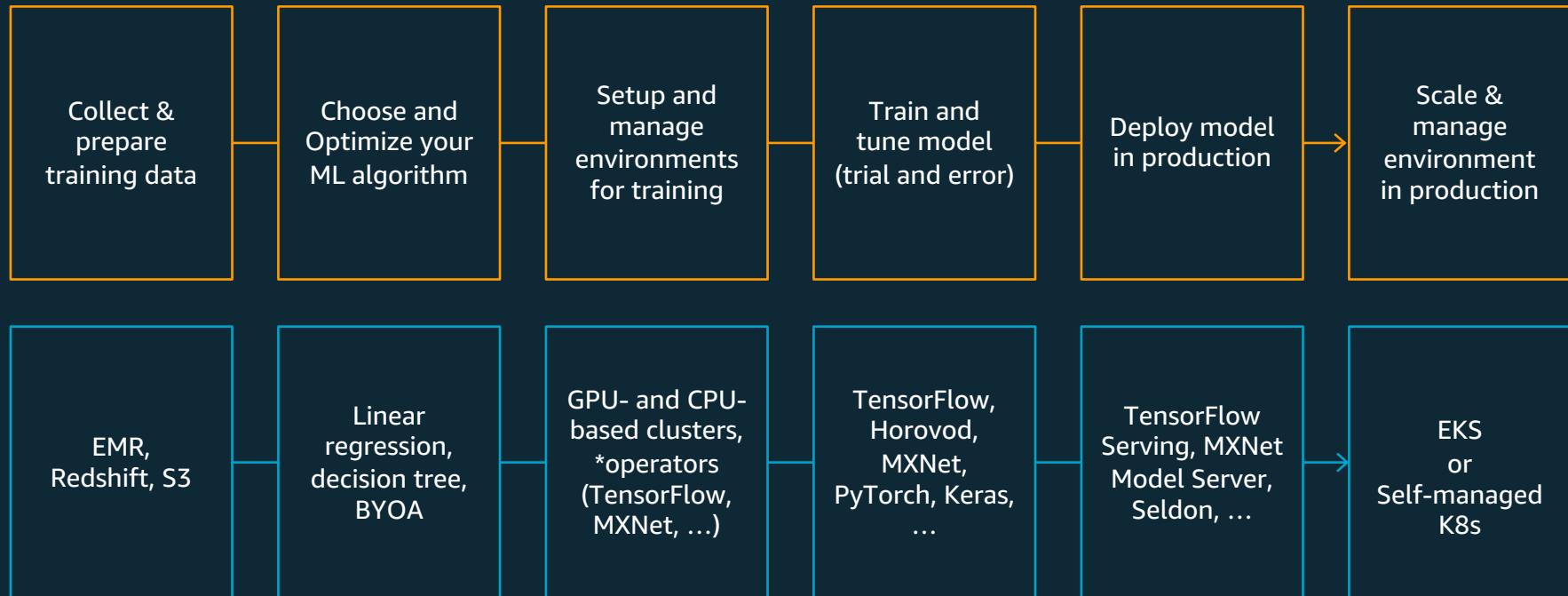
# Application Requirements 1.0

[https://github.com/kubeflow/community/blob/master/guidelines/application\\_requirements.md](https://github.com/kubeflow/community/blob/master/guidelines/application_requirements.md)

# Machine Learning pipeline



# Machine Learning pipeline for Kubernetes on AWS



# Thank You!

Chris Fregly, @cfregly

More Info...

<https://eksworkshop.com/kubeflow>

[https://distributed-training-workshop.go-aws.com/kubernetes\\_dist\\_training/workflow.html](https://distributed-training-workshop.go-aws.com/kubernetes_dist_training/workflow.html)

<https://github.com/aws-samples/eks-kubeflow-workshop>