KubeCon | CloudNativeCon

OPEN SOURCE SUMMIT

China 2023

KubeCon | CloudNativeCon

OPEN SOURCE SUMMIT

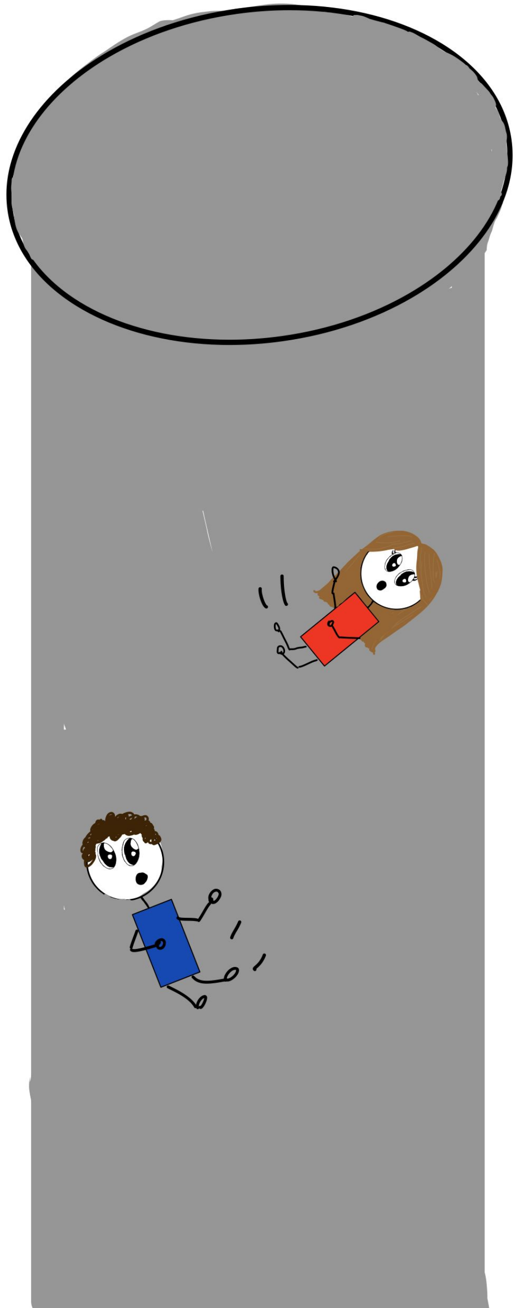
China 2023

Kubernetes Wonderland: Adventures in Platform Building

Alexa Griffith - Bloomberg
Mauricio Salatino - Diagrid

Agenda

- Platforms on top of Kubernetes
    - What do application development teams need?
    - What do data scientist need?
- Shared concerns and platform building
- Takeaways



Who are we?

Mauricio Salatino

OSS Software Engineer

Diagrid / Knative / Dapr

Alexa Griffith

Software Engineer

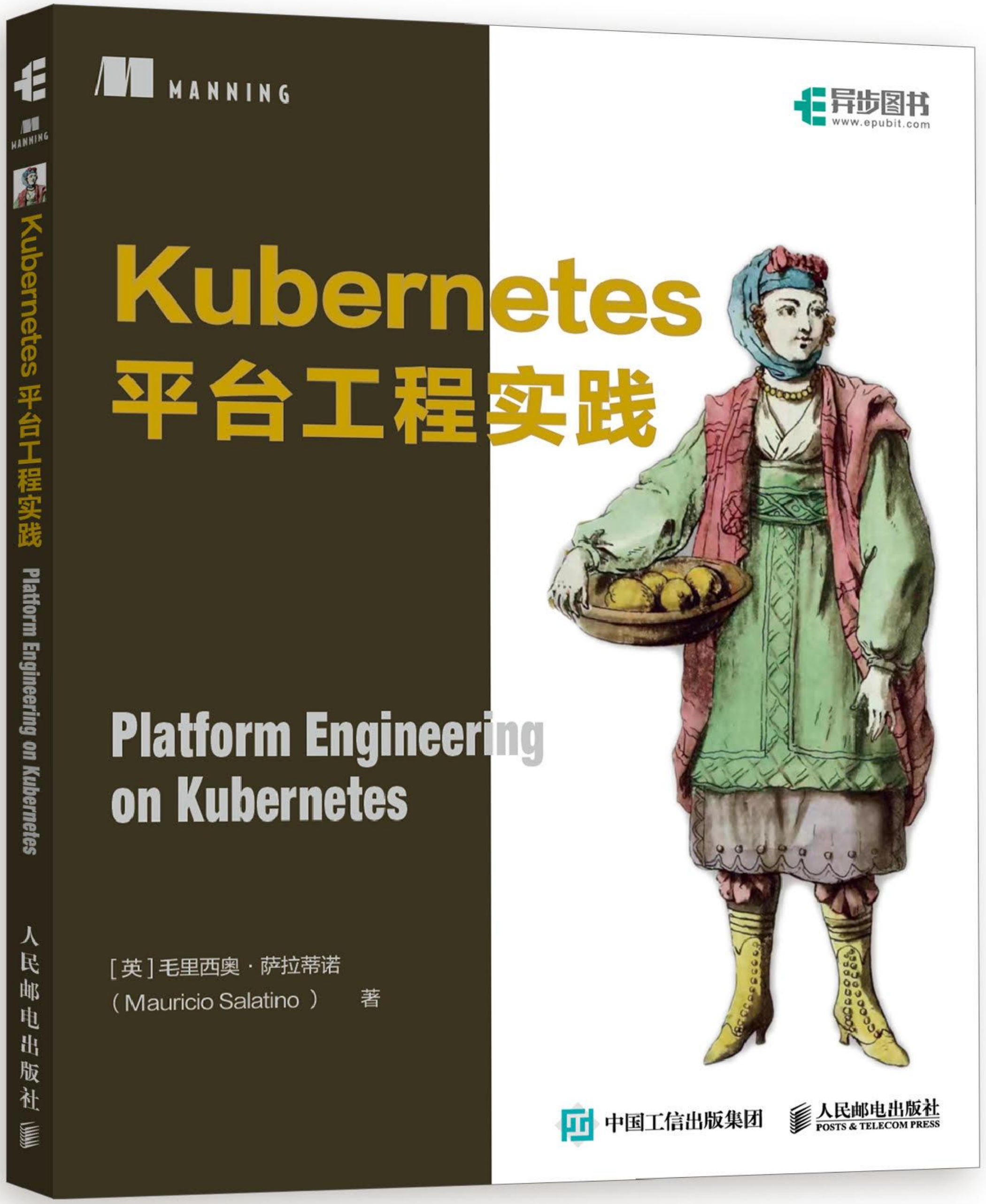
Bloomberg / KServe

Platform Engineering on Kubernetes

- Combining tools to enable teams to be productive
- Using Open Source and Cloud-Native tools
  - Dapr, Knative, Argo CD, Crossplane, Tekton, Dagger, OpenFeature, among others

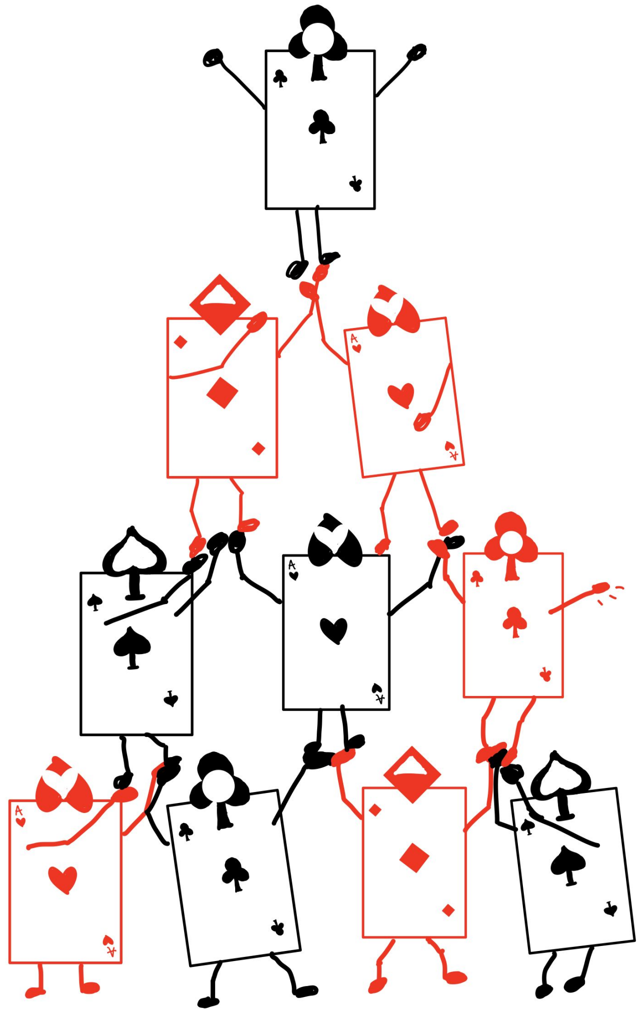
- Translated into Chinese in 2024 https://www.epubit.com/

- Thanks @dustise for the Chinese translations on the tutorials 🇨🇳🥳 https://github.com/salaboy/platforms-on-k8s

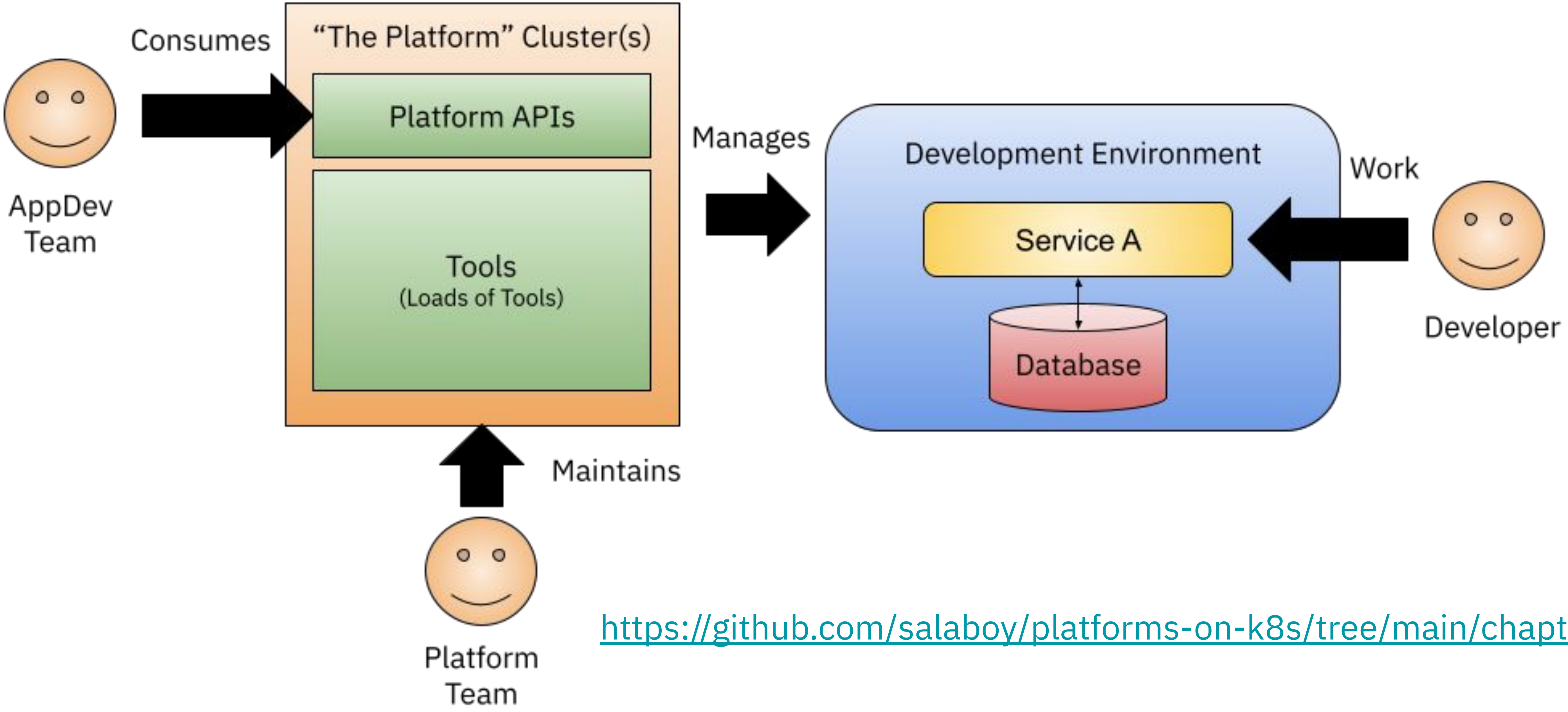


Platforms on top of Kubernetes

- Feels like an adventure
  - Scaling up your teams expertise
  - Avoiding making your teams' life more complicated
  - Avoiding decision paralysis
- Our platforms should provide teams with self-service APIs



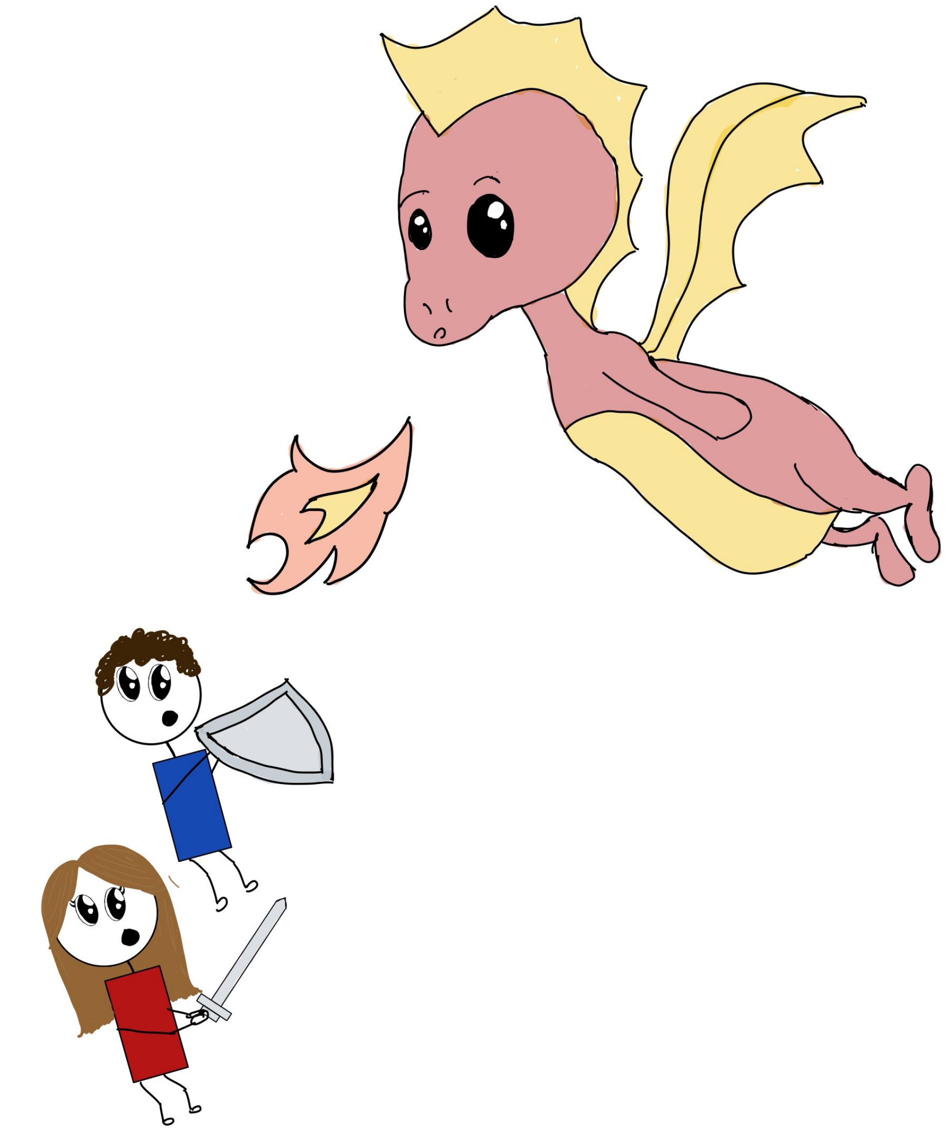
The shape of our adventure



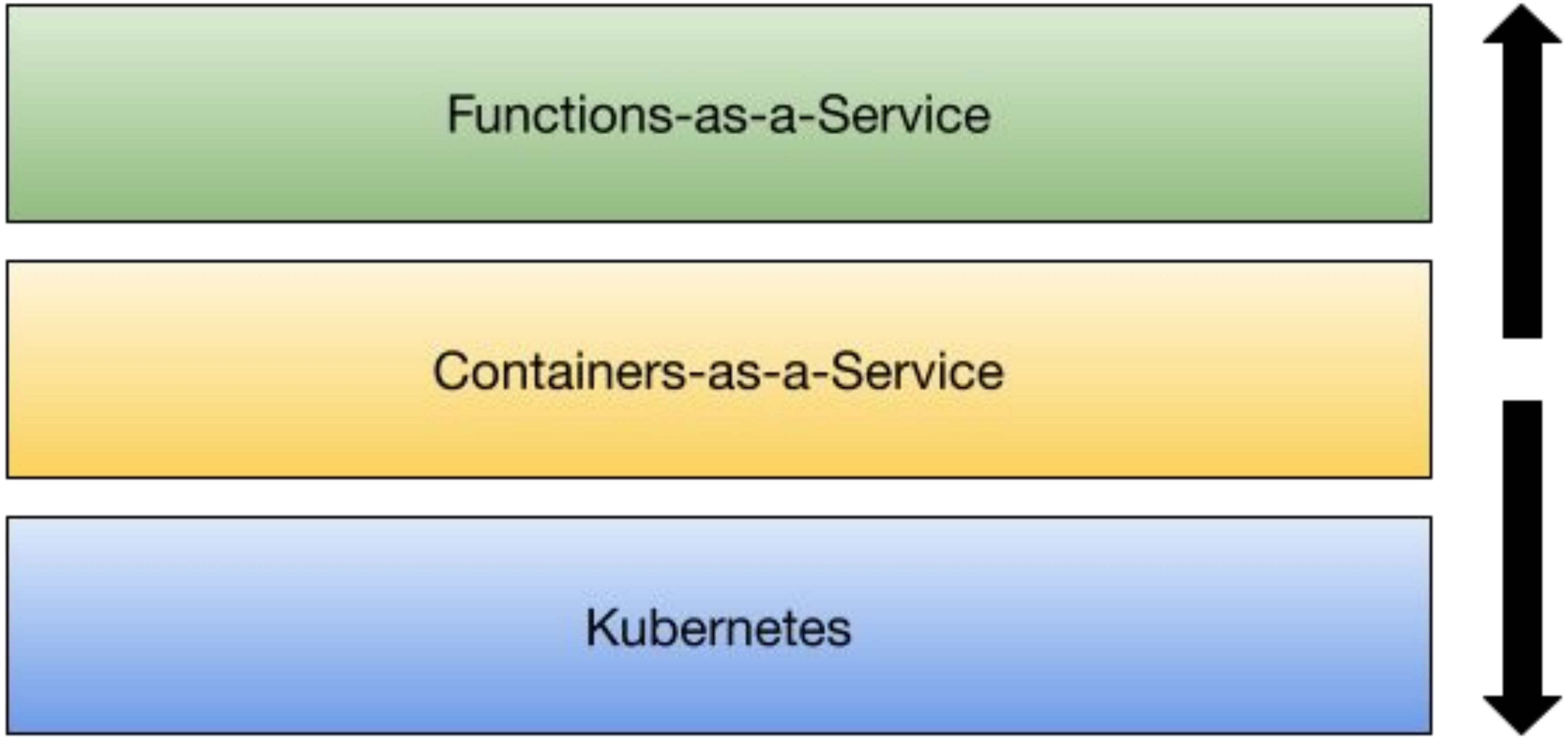
https://github.com/salaboy/platforms-on-k8s/tree/main/chapter-6

Different approaches

- Containers as a Service (Google Cloud Run, AWS App Runner)
- Functions as a Service (Alibaba Function Compute, Google Cloud Functions, AWS Lambdas)
- Standard APIs to hook into the infrastructure



Common Patterns



Functions-as-a-Service

Containers-as-a-Service

Kubernetes

More opinionated, less control, low cognitive load

Require teams to containerise applications

Less opinionated, more control, high cognitive load

Knative - CaaS & scale-to-zero

```
apiVersion: serving.knative.dev/v1
kind: Service
metadata:
  name: frontend
spec:
  template:
    spec:
      containers:
      - image: salaboy/frontend:v2.0.0
  traffic:
   <Traffic Rules>
```

Istio

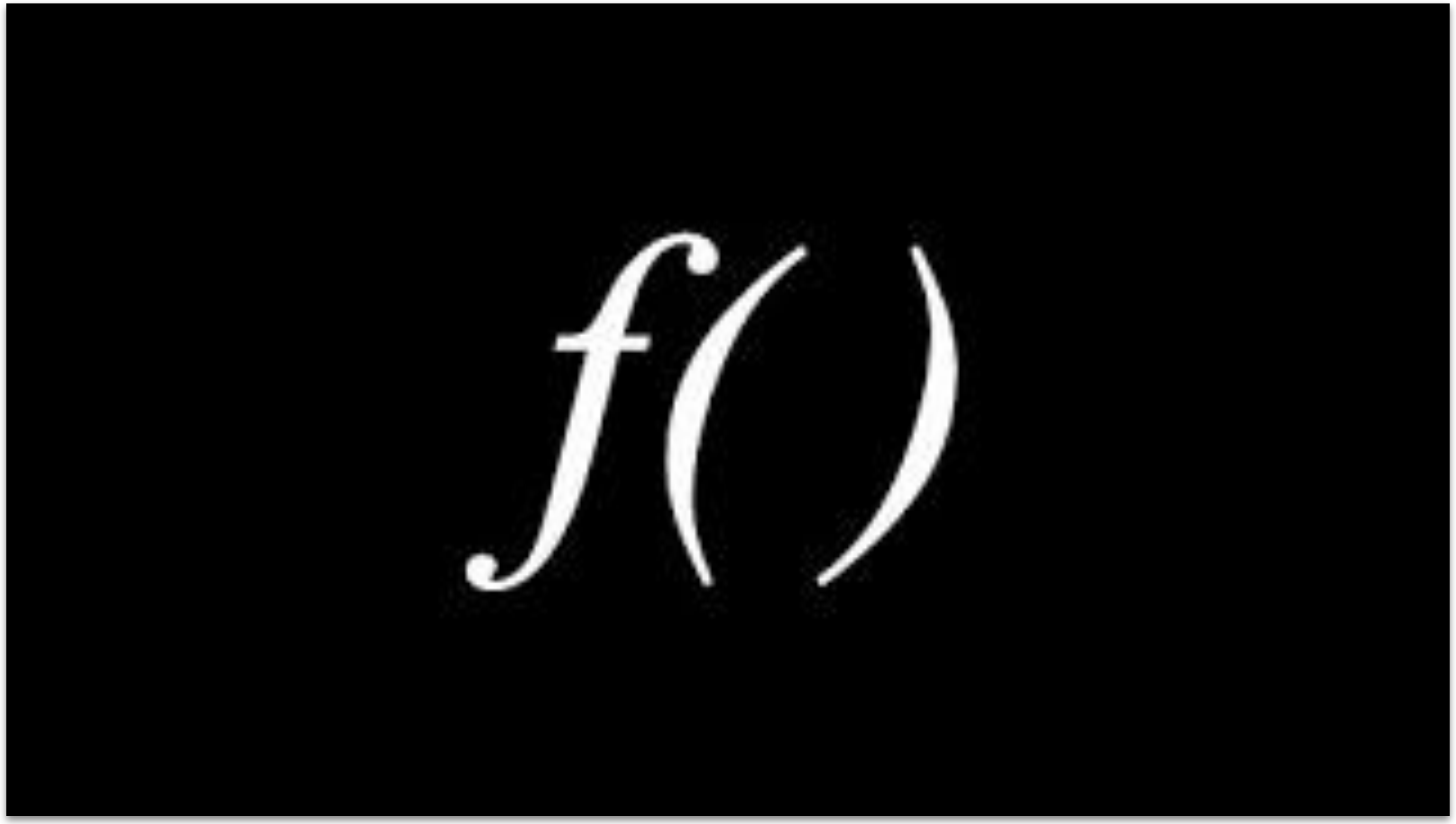
- Provide advanced traffic management and routing that Knative can expose to its users
- Provides mTLS and observability
- Knative abstract away the complexity of using Istio and provide a simple way to implement release strategies
- Traffic control
    - Ingress regulates who can access the resource/service
    - Egress checks if a principal identity is authorized to access the external service

https://github.com/salaboy/platforms-on-k8s/blob/main/chapter-8/knative/README.md

Knative Functions

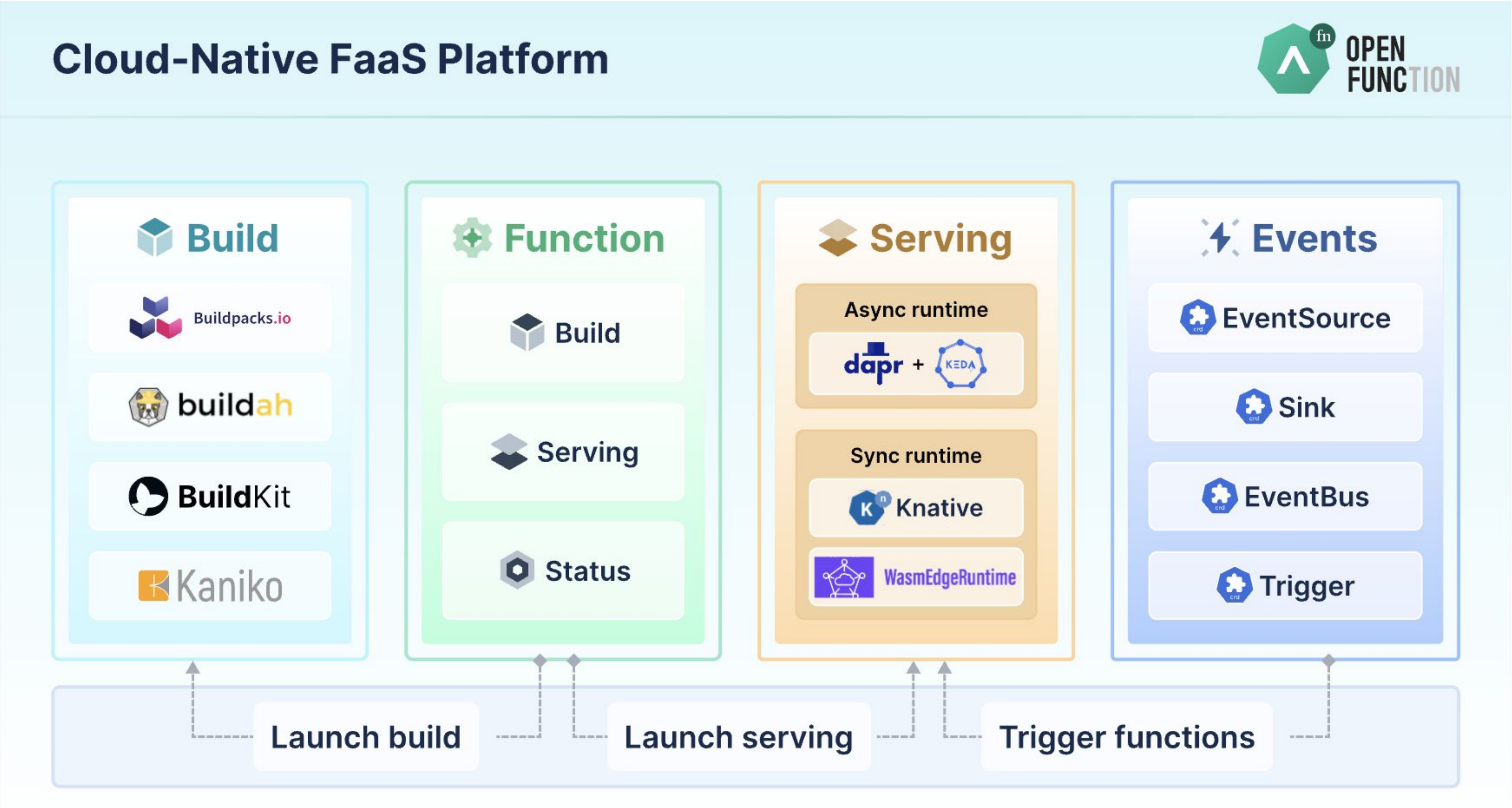
- https://github.com/knative/func
- Functions CLI

```
> func create -l go
> func deploy
```



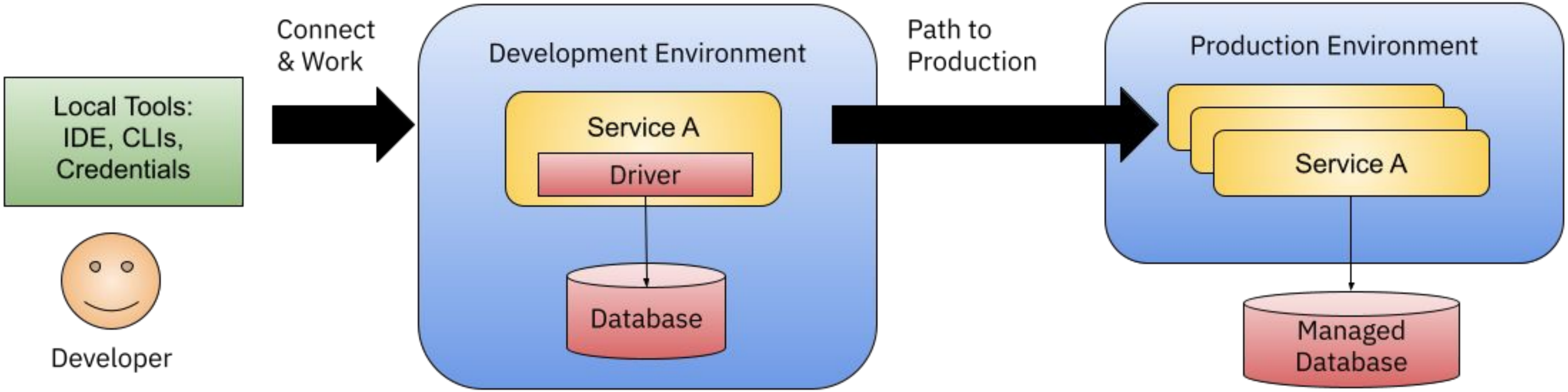
OpenFunction.dev

- https://openfunction.dev

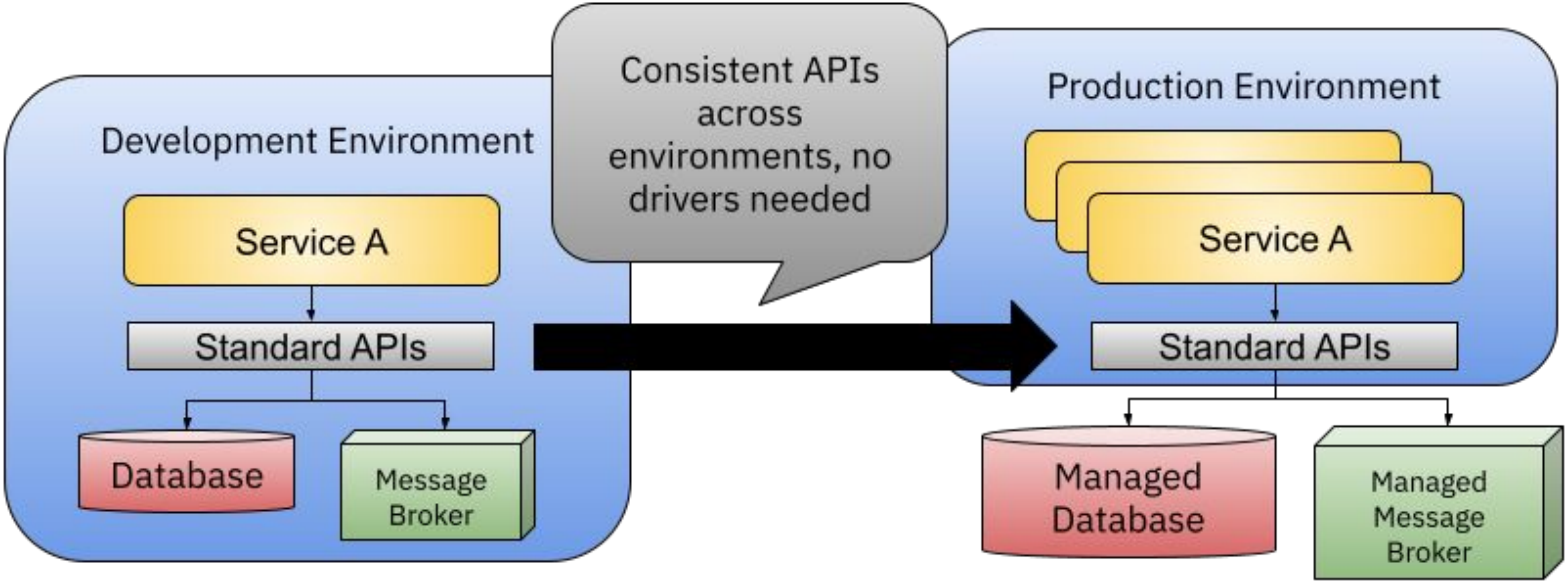


Cloud-Native FaaS Platform

But things gets complicated



APIs between apps and infrastructure



Dapr for Standard APIs

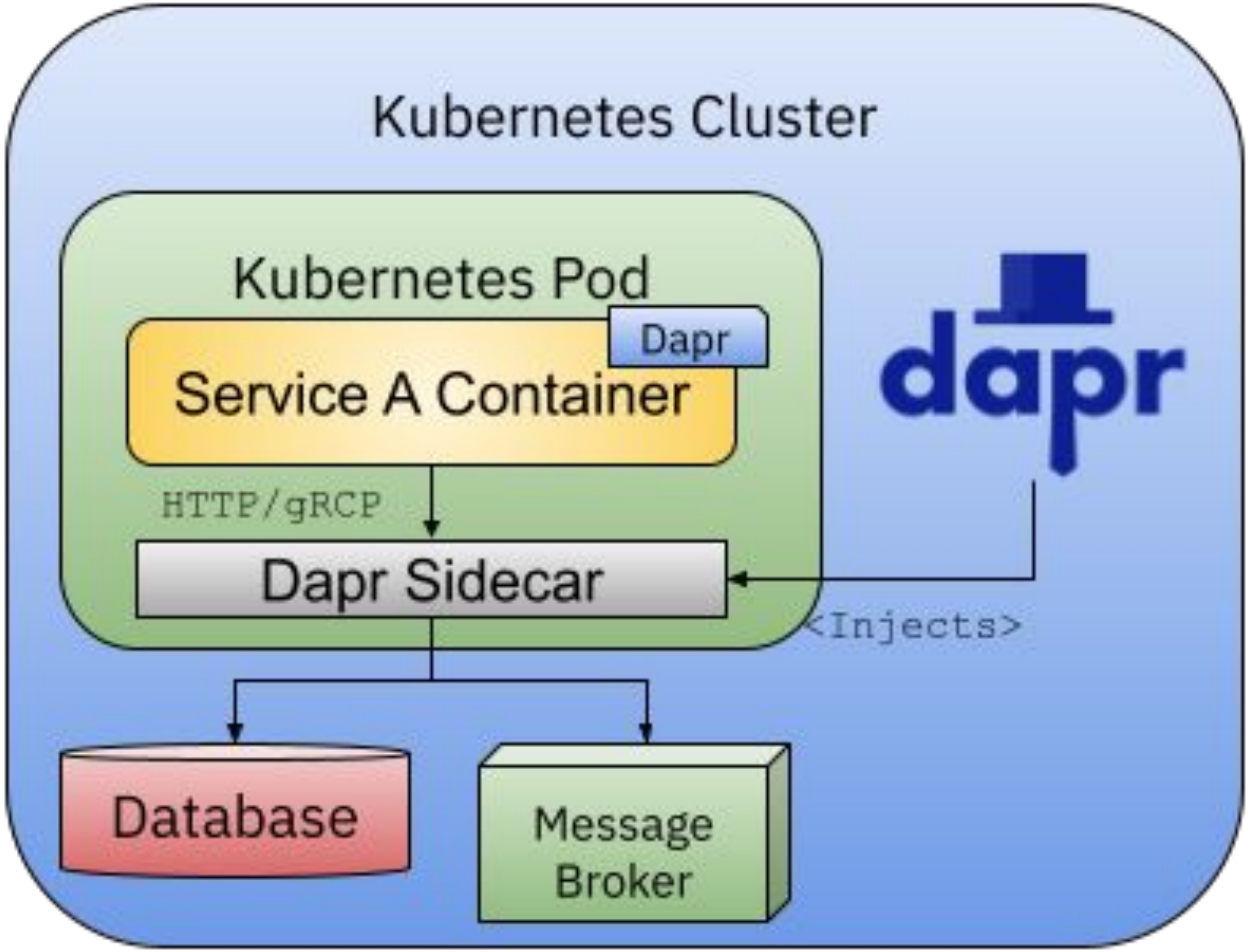
- https://dapr.io
- Application level APIs to solve distributed application challenges
- Dapr Building Blocks APIs
  - Statestore
  - PubSub
  - Configuration / Secrets
  - Resiliency Policies

https://blog.crossplane.io/crossplane-and-dapr/
https://blog.dapr.io/posts/2021/03/19/how-alibaba-is-using-dapr/
https://github.com/salaboy/platforms-on-k8s/tree/main/chapter-7

Knative + Dapr

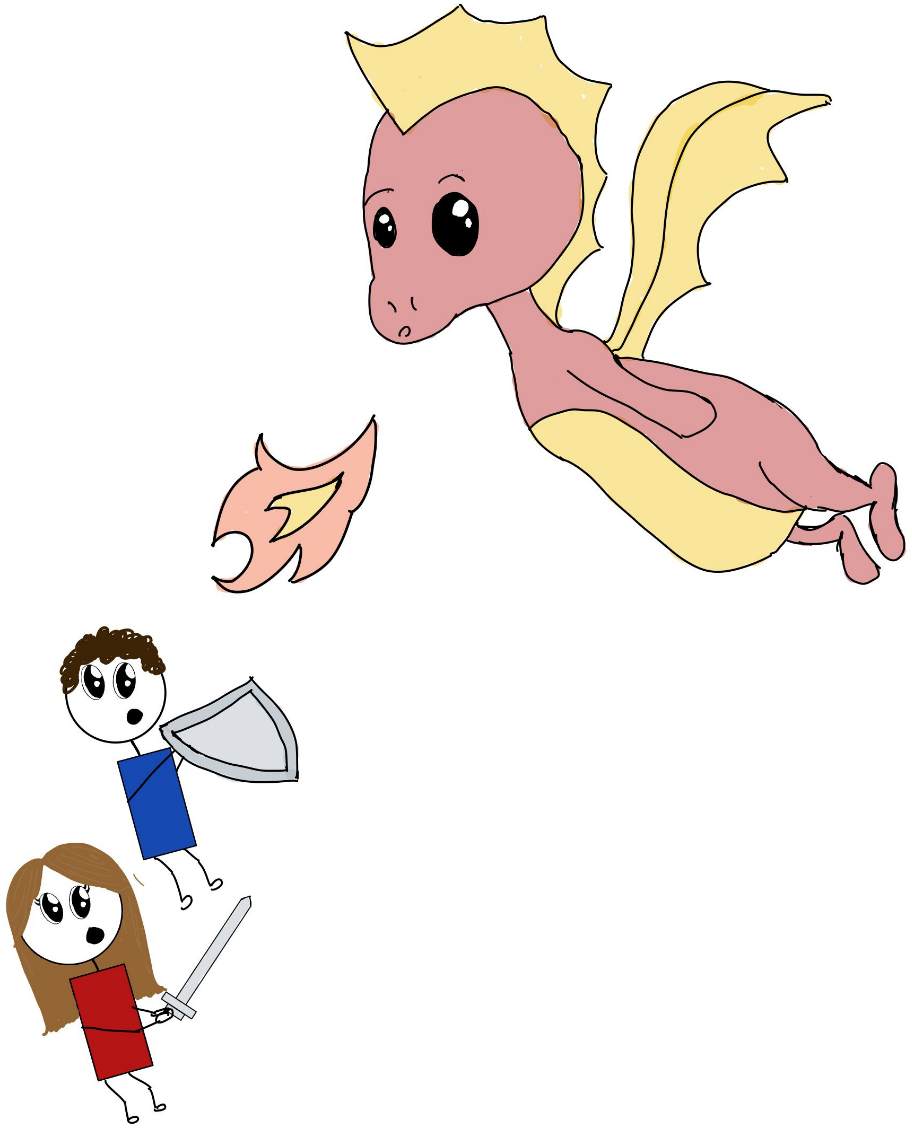
```
apiVersion: serving.knative.dev/v1
kind: Service
metadata:
  name: frontend
spec:
  template:
    metadata:
      annotations:
        dapr.io/app-id: frontend
        dapr.io/app-port: "8080"
        dapr.io/enabled: "true"
    spec:
      containers:
      - image: salaboy/frontend:v2.0.0
```

Dapr on Kubernetes



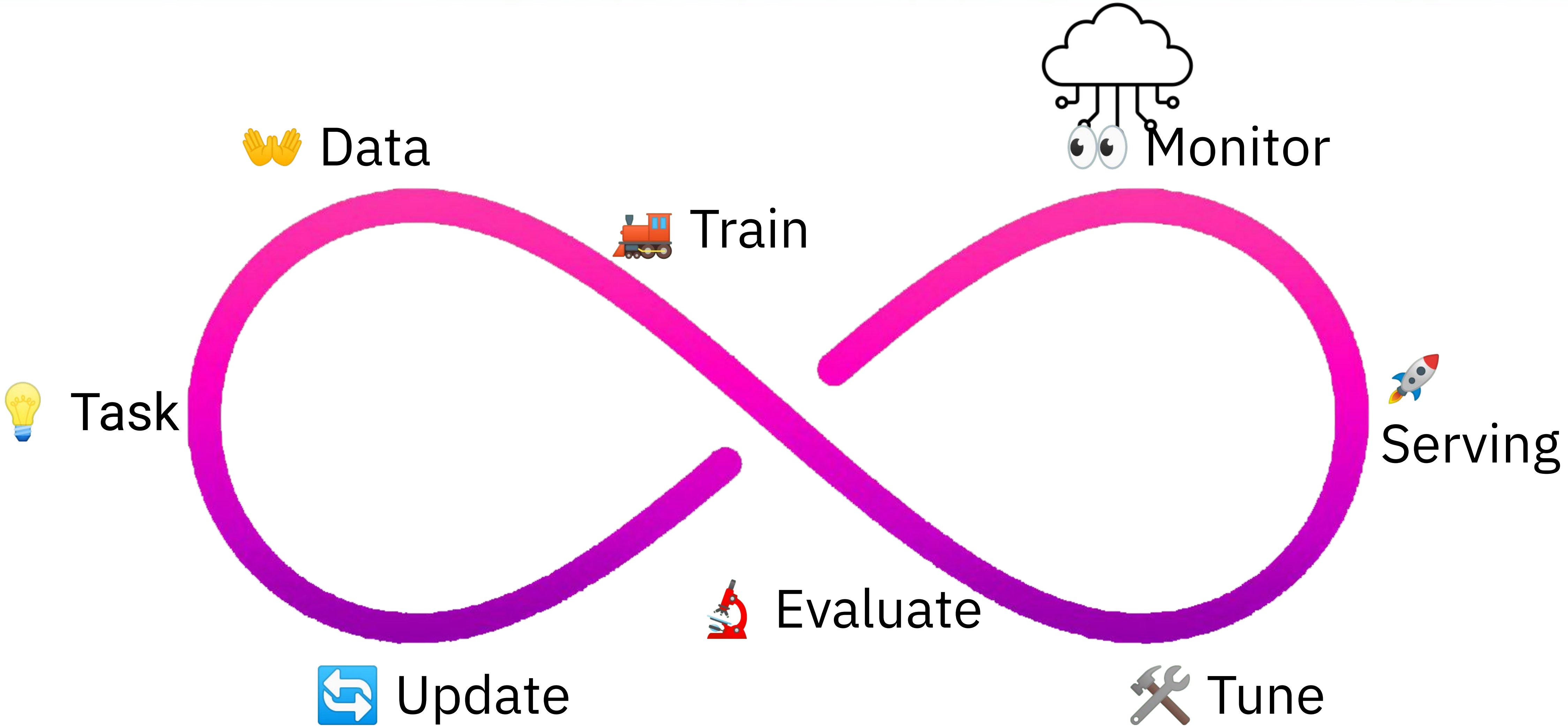
Machine Learning on Kubernetes

- Training & Inference workflows benefit from standard APIs
- Tools like KServe, Kubeflow, Buildpacks, etc. allow for quick development on top of Kubernetes



Model Development Life Cycle (#MDLC)

1. 💡 Task
2. 👐 Data
3. 🚂 Train
4. 🔬 Evaluate
5. 🛠️ Tune
6. 🚀 Serving
7. 👀 Monitor
8. 🔄 Update



Data Science Platform Portfolio

Data Access & Exploration

Jupyter Notebooks

Data Access Libraries

Credential Management
(Identities, Secrets, IDX)

Cataloguing & Discovery

Dataset Onboarding

Model Training

ML Frameworks
(TensorFlow, PyTorch, Deepspeed, MPI)

High Performance Compute
(GPU, Infiniband)

Monitoring & Debugging
(Grafana)

Resource Management
(CPU, GPU, RAM, NVMe)

Experiment Management

Developer Console
(UI)

Model Metrics

Reproducible Representations of ML Tasks
(YAMLs, Blueprints, Custom Forms)

Code Tracking
(Buildpacks)

Model Serving

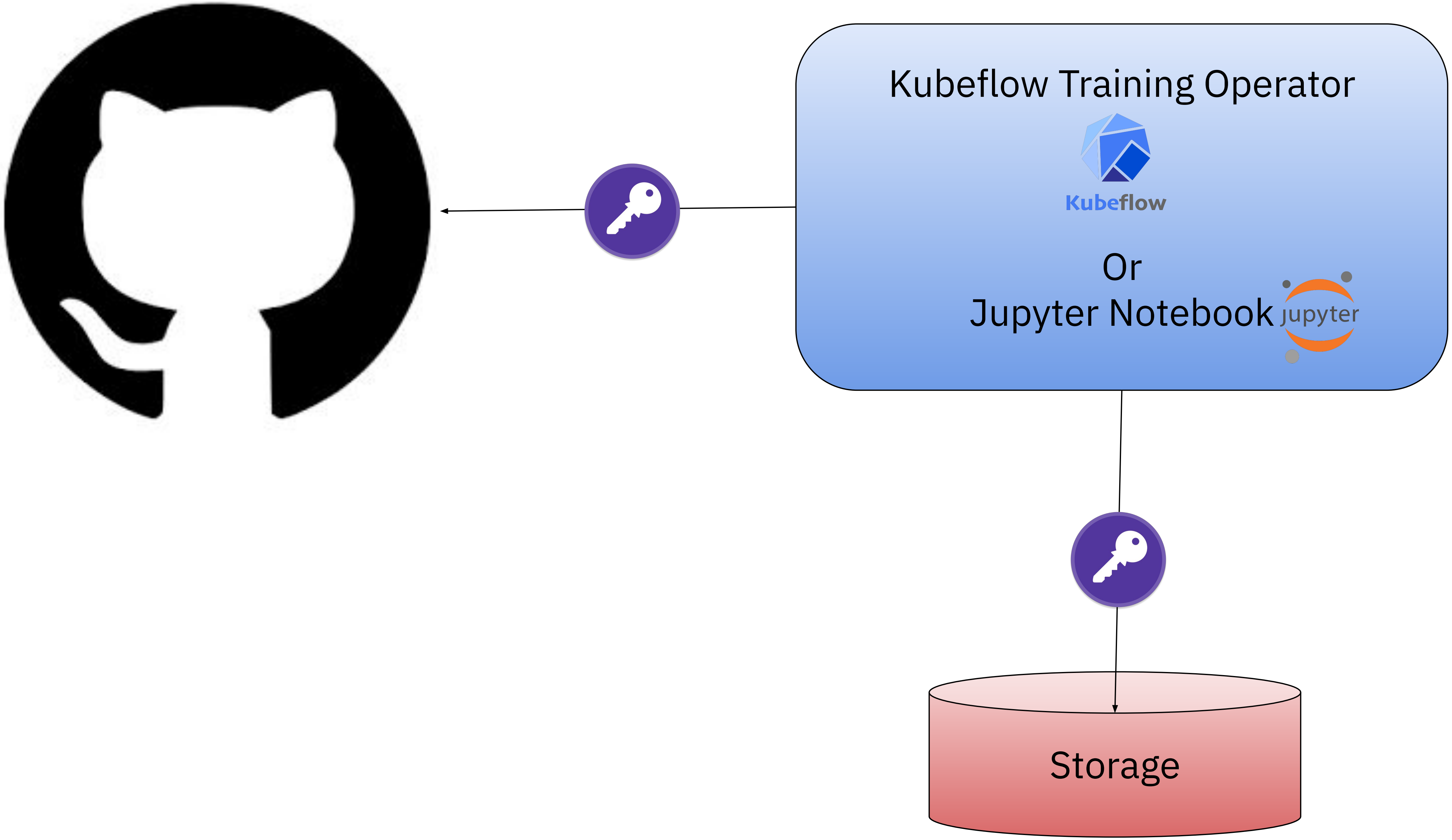
Inference API
Streaming & Request-Response
(KServe)

Deployment Workflow

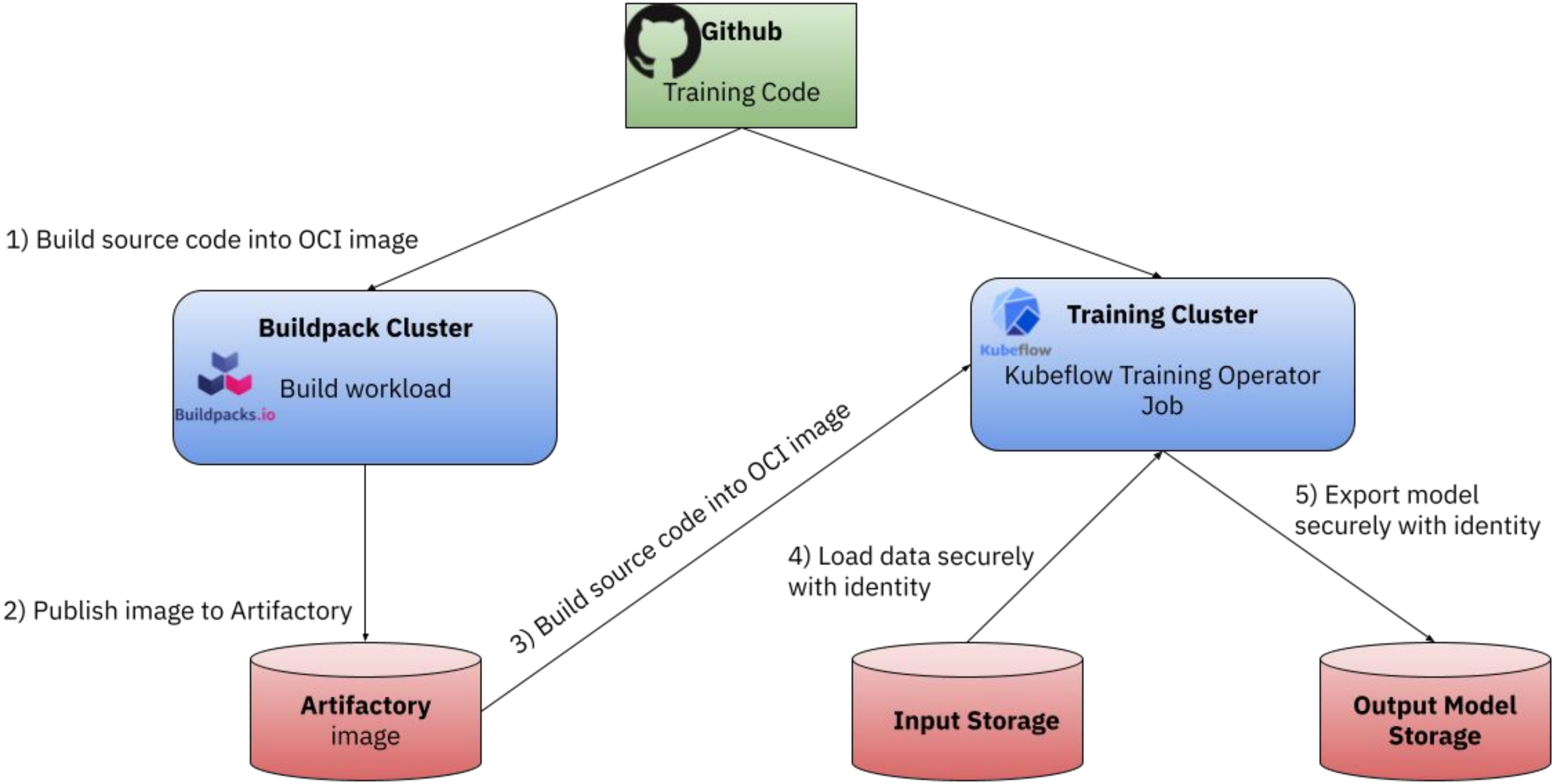
Service Monitoring
(UI, Grafana)

Hardware Performance
(Scale-to-Zero, GPUs)

Training Platform Offerings

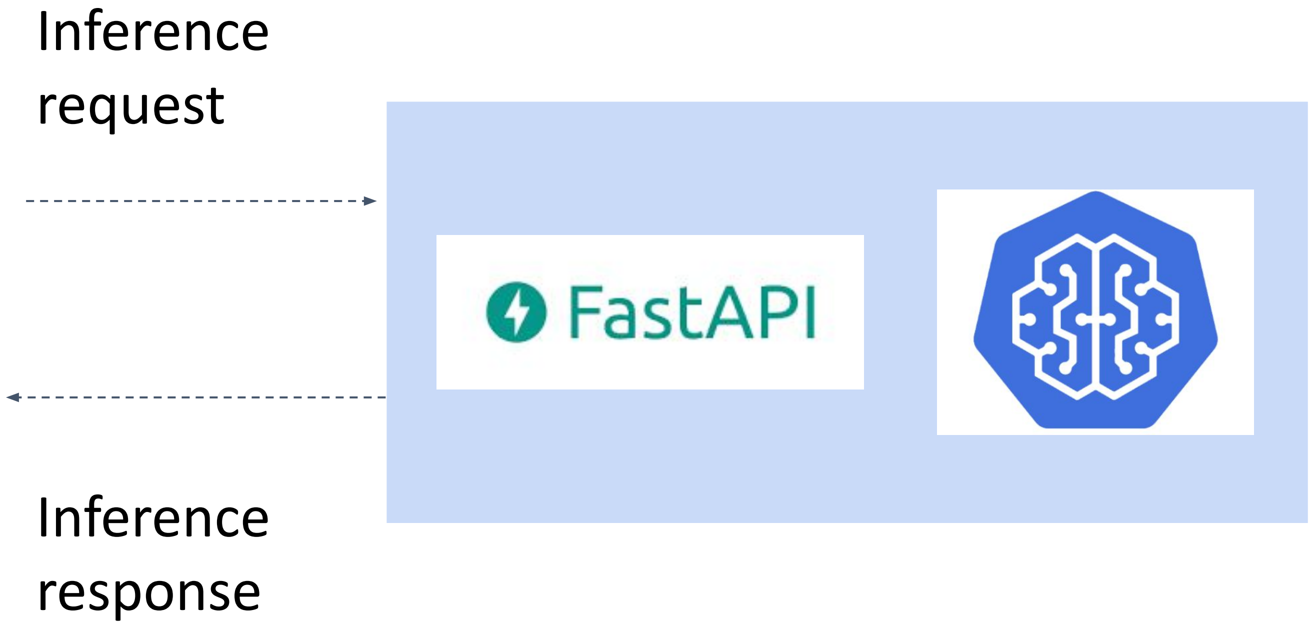


Training Lifecycle


Github
Training Code
1) Build source code into OCI image
Buildpack Cluster
Build workload
Buildpacks.io
Training Cluster
Kubeflow
Kubeflow Training Operator Job
2) Publish image to Artifactory
3) Build source code into OCI image
4) Load data securely with identity
5) Export model securely with identity
Artifactory image
Input Storage
Output Model Storage

Model Deployment (Inference) Platform

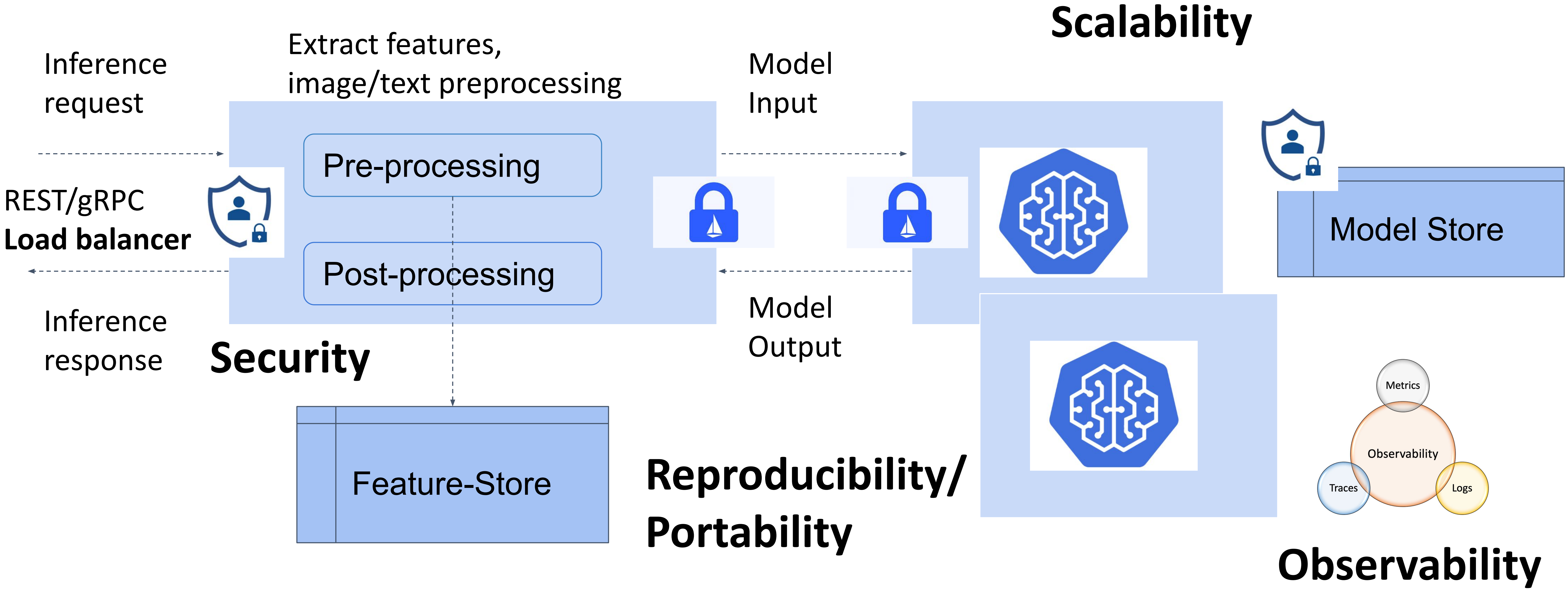
“Launching AI application pilots is deceptively easy, but deploying them into production is notoriously challenging.”



The State & Future of Cloud Native Model Serving - https://www.youtube.com/watch?v=786VaGAfm6I

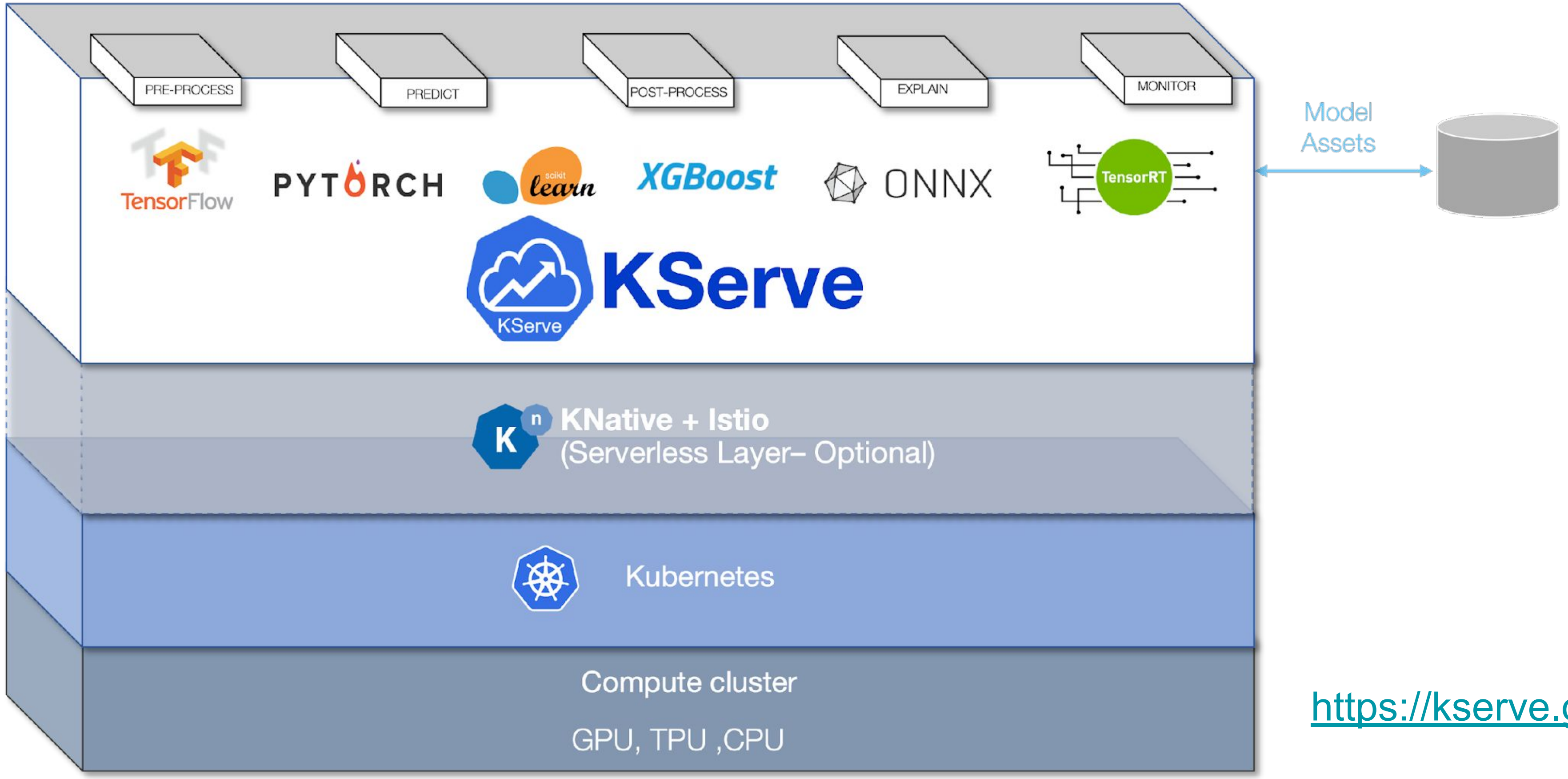
Model Deployment (Inference) Platform

"Launching AI application pilots is deceptively easy, but deploying them into production is notoriously challenging."



KServe

- **KServe** is a **highly scalable** and **standards-based cloud-native model inference platform** on Kubernetes for Trusted AI that encapsulates the complexity of deploying models to production.
- KServe can be deployed **standalone** or as an **add-on component** with **Kubeflow** in the **cloud** or **on-premises** environment.



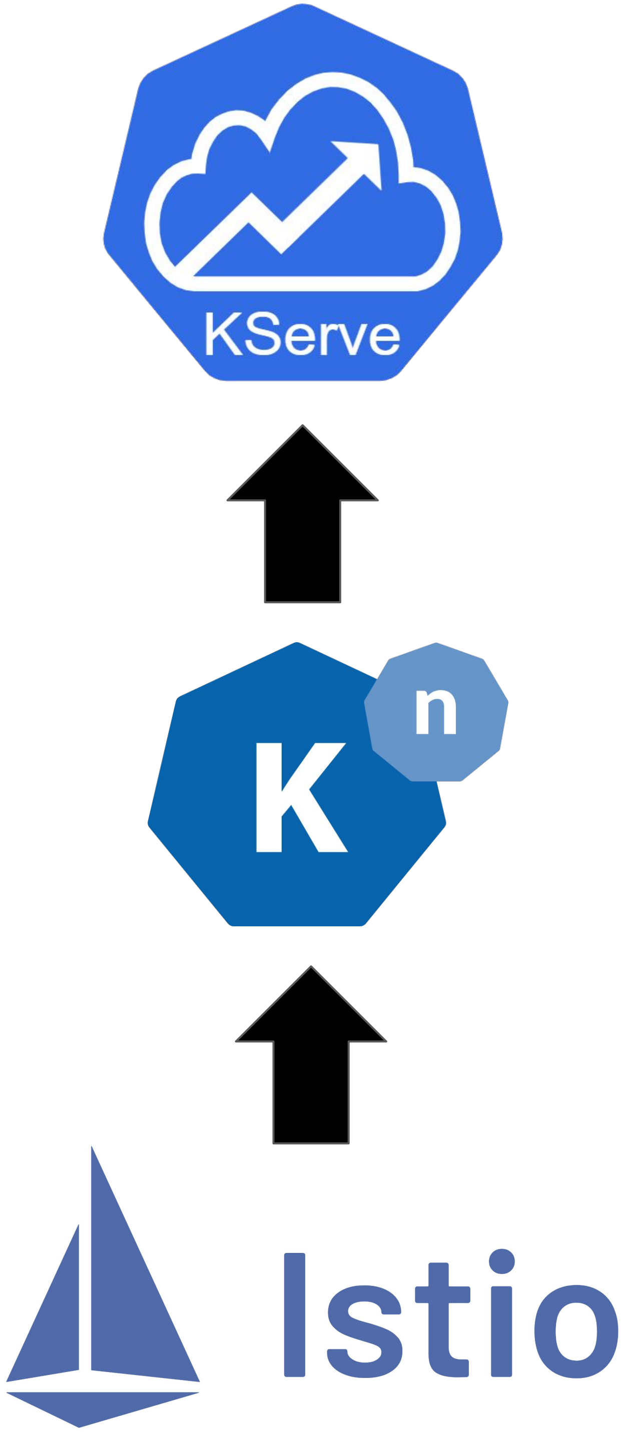
https://kserve.github.io/website/0.11/

KServe Open Inference Protocol

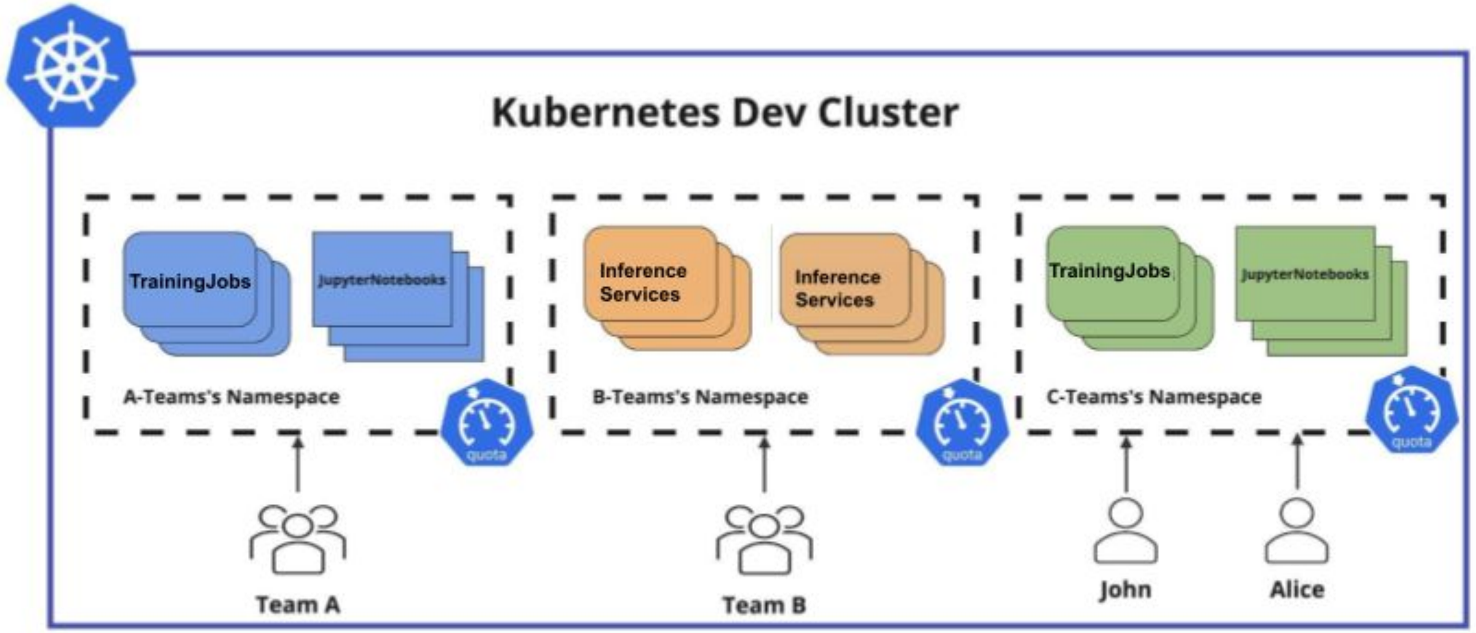
REST	gRPC
GET v2/health/live	rpc ServerLive(ServerLiveRequest) returns (ServerLiveResponse)
GET v2/health/ready	rpc ServerReady(ServerReadyRequest) returns (ServerReadyResponse)
GET v2/models/{model_name}/ready	rpc ModelReady(ModelReadyRequest) returns (ModelReadyResponse)
GET v2/models/{model_name}	rpc ModelMetadata(ModelMetadataRequest) returns (ModelMetadataResponse)
POST v2/models/{model_name}/infer	rpc ModeInfer(ModelInferRequest) returns (ModelInferResponse)

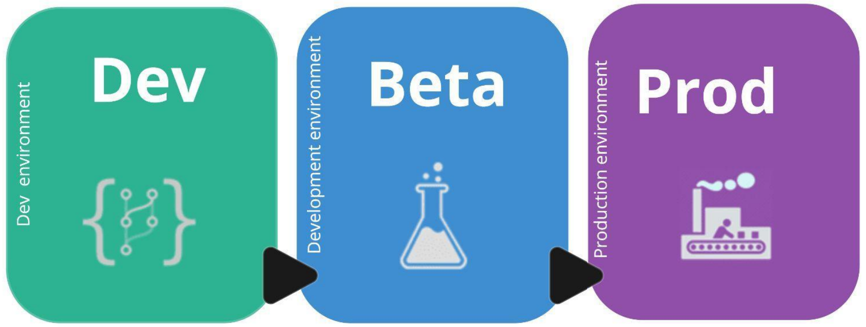
KServe + Knative + Istio

```yaml
apiVersion: "serving.kserve.io/v1beta1"
kind: "InferenceService"
metadata:
  name: "example-inference-svc"
spec:
  transformer:
    containers:
    - image: kserve/image-transformer:latest
      name: kserve-container
  predictor:
    model:
      modelFormat:
        name: pytorch
      storageUri: "gs://path-to-model/pytorch/v1"
```



Platform Features





- Both training and inference platforms offer standard APIs to users that allow them to choose among a variety of tooling for their services.

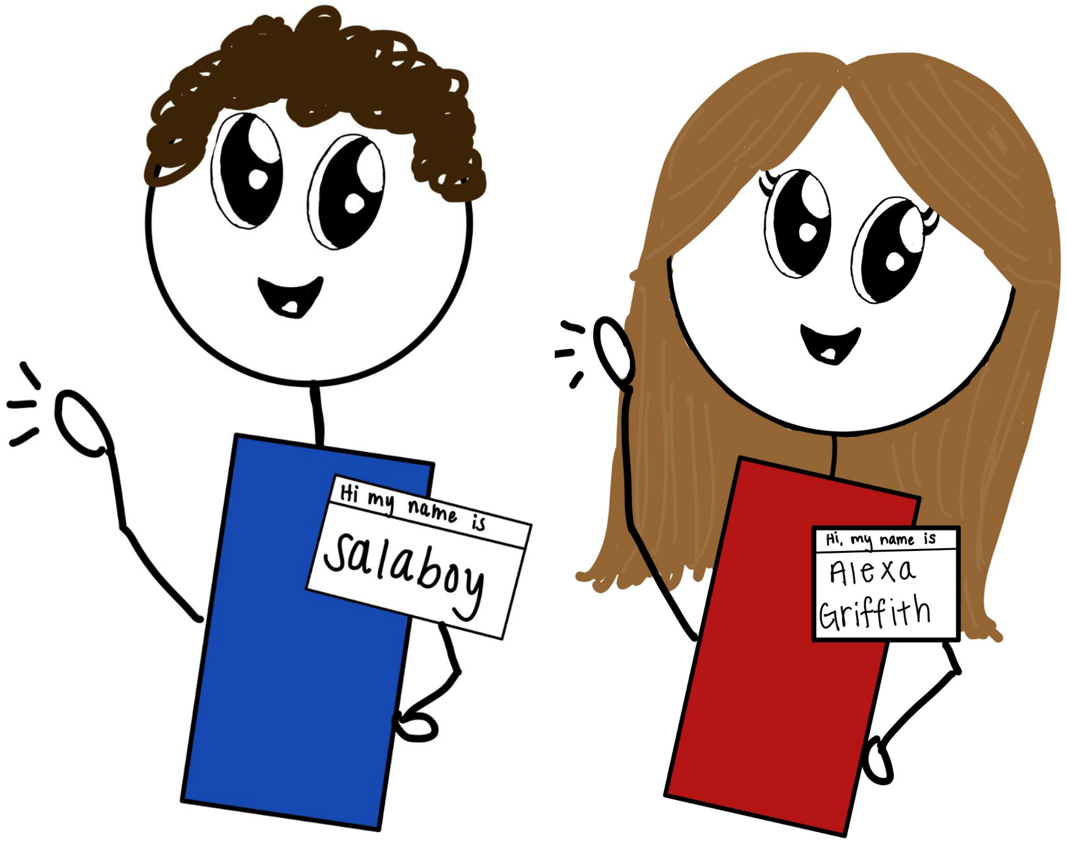
Demo

https://github.com/salaboy/kubecon-china-2023/

Takeaways

- Using software development skills to enable and scale up teams
- Focusing on APIs enable Platform teams to provide a self-service approach for teams to have access to the tools they need
- The same principles can be applied to development teams, data scientist, product teams, operations, etc.
- Adopting Open Source solutions require expertise. Open Standards can help your teams avoid “decision paralysis”

Thank you!



Follow us on Twitter!
@lexal0u
@salaboy

Learn more about us and our work

https://www.TechAtBloomberg.com
https://www.bloomberg.com/engineering
https://www.bloomberg.com/careers

References

- TAG App Delivery Platforms White Paper
  https://tag-app-delivery.cncf.io/whitepapers/platforms/
- Free step-by-step tutorials (Chinese translations thanks to @dustise 🥳)
  https://github.com/salaboy/platforms-on-k8s/
- Building Bloomberg's ML Inference Platform Using KServe
  https://www.bloomberg.com/company/stories/the-journey-to-build-bloombergs-ml-inference-platform-using-kserve-formerly-kfserving/
- Provisioning and consuming Multi Cloud Infrastructure
  https://blog.crossplane.io/crossplane-and-dapr/
- Dapr and Alibaba Cloud
  https://blog.dapr.io/posts/2021/03/19/how-alibaba-is-using-dapr/
- Red Light, Green Light: Traffic Security in the Service Mesh wi... Alexa Nicole Griffith & Zhenni Fu
  https://www.youtube.com/watch?v=f6jMix46ZD8
- Exploring ML Model Serving with KServe (with fun drawings) - Alexa Nicole Griffith, Bloomberg
  https://www.youtube.com/watch?v=FX6naJLaq2Y
- The State & Future of Cloud Native Model Serving
  https://www.youtube.com/watch?v=786VaGAfm6I

KubeCon | CloudNativeCon

OPEN SOURCE SUMMIT

China 2023