Kubeflow's Mission

Make it Easy for Everyone to **Develop, Deploy** and **Manage** Portable, Distributed ML on Kubernetes
Argo CD

Declarative continuous delivery (CD) system for Kubernetes
ML on K8s Is Difficult

- Cobble together an ML platform out of 10's components
- ML bits
  - Jupyter
  - TFJob/PyTorch
- K8s bits
  - Networking(Ambassador, ISIO, CertManager)
  - GPU installers
- Cloud Bits
  - K8s cluster
  - Storage
Enable GitOps for ML

- Fully declarative
- `kfctl` is a two step process
  - Create configs
  - Apply configs
- GitOps reduces the toil of managing infrastructure
  - Automation (e.g. WeaveFlux/ArgoCD) keeps infrastructure up to date
Create Kubeflow Manifests In Git

kfctl.sh init ${KFAPP} --platform none
cd ${KFAPP}
${KUBEFLOW_REPO}/scripts/kfctl.sh generate k8s
git add .
git commit -m "My app"
git push origin
Install Argo CD

argocd cluster add CONTEXTNAME

export KUBEFLOW_REPO_URL='Replace with a ssh or https git endpoint'

argocd app create kubeflow --repo $KUBEFLOW_REPO_URL --path ks_app --env default

argocd app sync kubeflow
Sync Resources

argocd app sync kubeflow
Keep Your Cluster Up To Date

- Push changes to Git
- Argo CD detects difference between manifests and what's running
- Sync manually or automatically
More Info

- Kubeflow Docs for GitOps
- www.kubeflow.org
- kubeflow.slack.com
- Argo CD
- Weaveworks Flux
Thank you

- Danny Thomson (Argo & Intuit)
- Edward Lee (Argo & Intuit)
- Jesse Suen (Argo & Intuit)
- Mukulika Kapas (Argo & Intuit)