Argo Worklows

> A open source container-native workflow engine

- Defines workflows where each step is a container
- Multi-step Directed Acyclic Graph (DAG)
- CNCF Graduated Project

Why use Argo Workflows

- Most popular workflow execution engine for Kubernetes
- Light-weight and scaleable
- Designed for Containers
- Cloud Agnostic

Argo Workflows Vs K8s Jobs

Workflow

```yaml
apiVersion: argoproj.io/v1alpha1
kind: Workflow
metadata:
  generateName: hello-world-
  labels:
    workflows.argoproj.io/archive-strategy: "false"
  annotations:
    workflows.argoproj.io/description: |
      This is a simple hello world example.
spec:
  entrypoint: whalesay
  templates:
  - name: whalesay
    container:
      image: docker/whalesay:latest
      command: [cowsay]
      args: ["hello world"]
```

Jobs

```yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: pi
spec:
  template:
    spec:
      containers:
      - name: pi
        image: perl:5.34.0
        command: ["perl",  "-Mbignum=bpi", "-wle", "print bpi(2000)"]
      restartPolicy: Never
  backoffLimit: 4
```

Why use Workflows over Jobs?

- Argo Provides a higher level of abstraction
- Argo Allows for the coordination of multiple dependent steps

Example Time: (Argo Workflows)

```
# Initialize Cluster:
# - install argocd, workflows, argo-events
$ make init

# Bring up Argo Workflows UI (demo mode - no auth)
$ make argo-workflows-ui

# Deploy basic workflow example:
$ make demo-workflow
```

Workflow DAG

```
# Deploy Dag Application:
$ make demo-dag
```

Workflow Cron

Similar to K8s CronJobs

```
# Deploy Cron App:
$ make demo-cron
```

More examples:

- https://github.com/argoproj/argo-workflows/tree/main/examples

Going Deeper in the Argo Ecosystem

Arogo Events

What is it and why do we need it?

- Event Driven workflow framework
- Allows us to trigger k8s objects from many different events

Main Components

- Event Source
  - Event to consume (AWS, Webhooks, GIT,...)
- Sensor - event dependencies and triggers
- Event Bus - Transport layer
- Trigger - workload to execute

Demo Time - Webhook

```
# Apply the webhook demo
$ make demo-webhook

# Hit the webhook
$ make demo-webhook-run
```