KubeCon | CloudNativeCon

North America 2017

Templating K8s
Easily Managing Applications via Common Configuration

Don Steffy, Sr Software Engineer, Oath
Anubhav Aeron, Software Engineer, Oath

Challenge

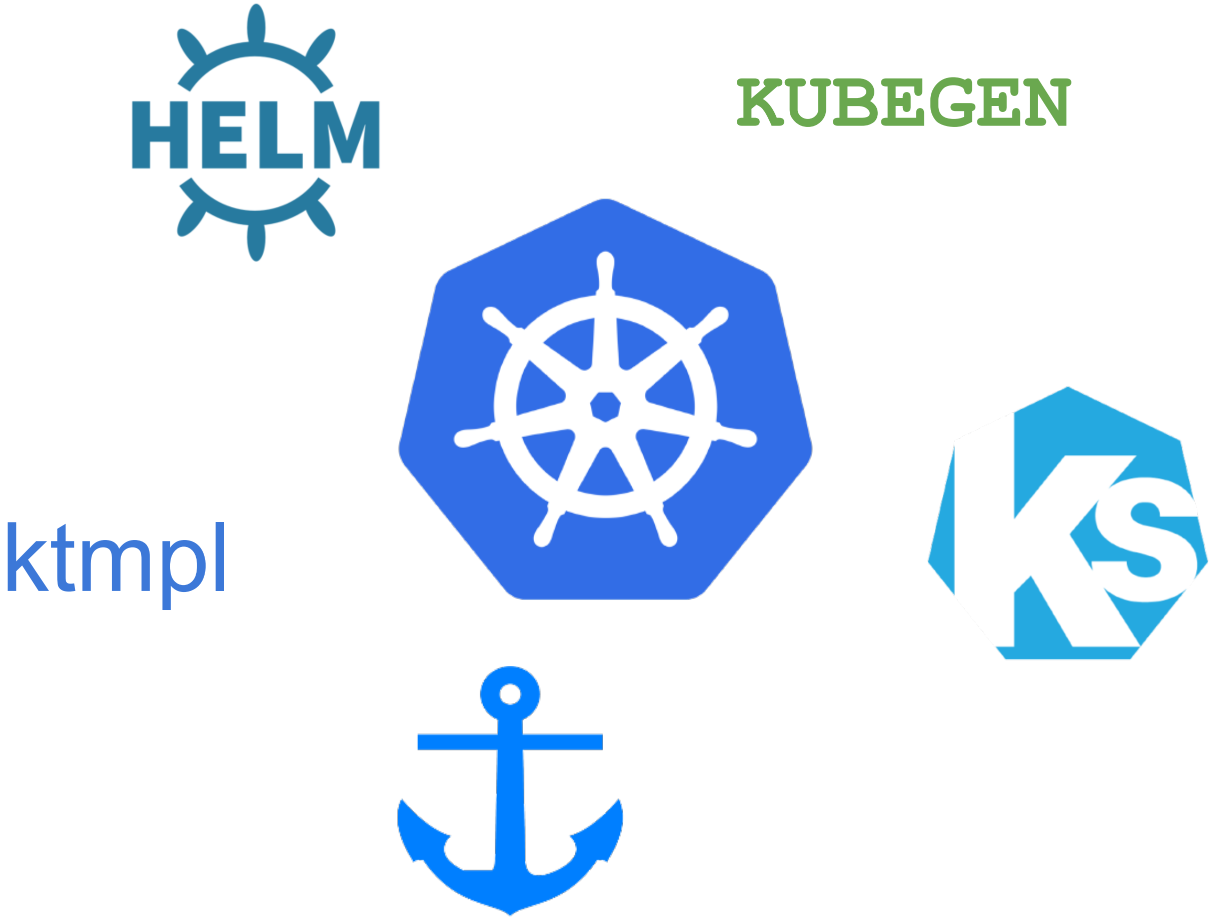
- Matrix Hell
  - 100 apps to migrate to k8s
  - Each with multiple environments
  - Each with multiple yaml files
    - deployment
    - service
    - autoscale
    - ingress

Requirements

- Simplicity
  - Simple yaml interface
- Flexibility
  - Ability to override settings
- Stability
  - Test all changes to the system

Templating Options

- helm
- ksonnet/kubecfg
- kubegen
- ktmpl
- kedge
- and more…



Simplicity: Simple Yaml Interface

- Settings clearly describe use cases

application.yaml

common settings

```
template: nodejs-app:stable
appName: foo-app
maintainer: foo-team@oath.com
baseimage: nodejs:6.0.170725-3
routing:
  ports: 4080
sidecars:
  athenz
    domain: oath.foo
  splunk:
    instance: foo.splunk:443
  metrics:
    namespace: foo
```

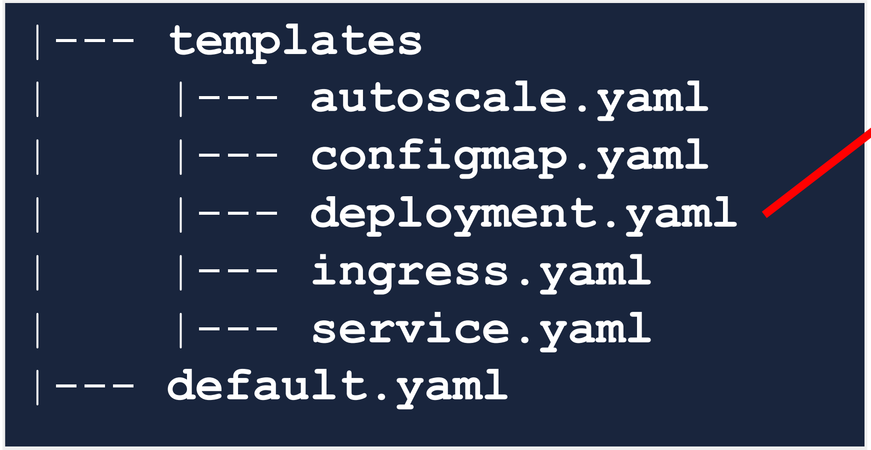
environment-specific settings

```
jobs:
  stage:
    environment: stage
    colo: west-1
  production-east-1:
    environment: production
    colo: east-1
  production-west-1:
    environment: production
    colo: west-1
```

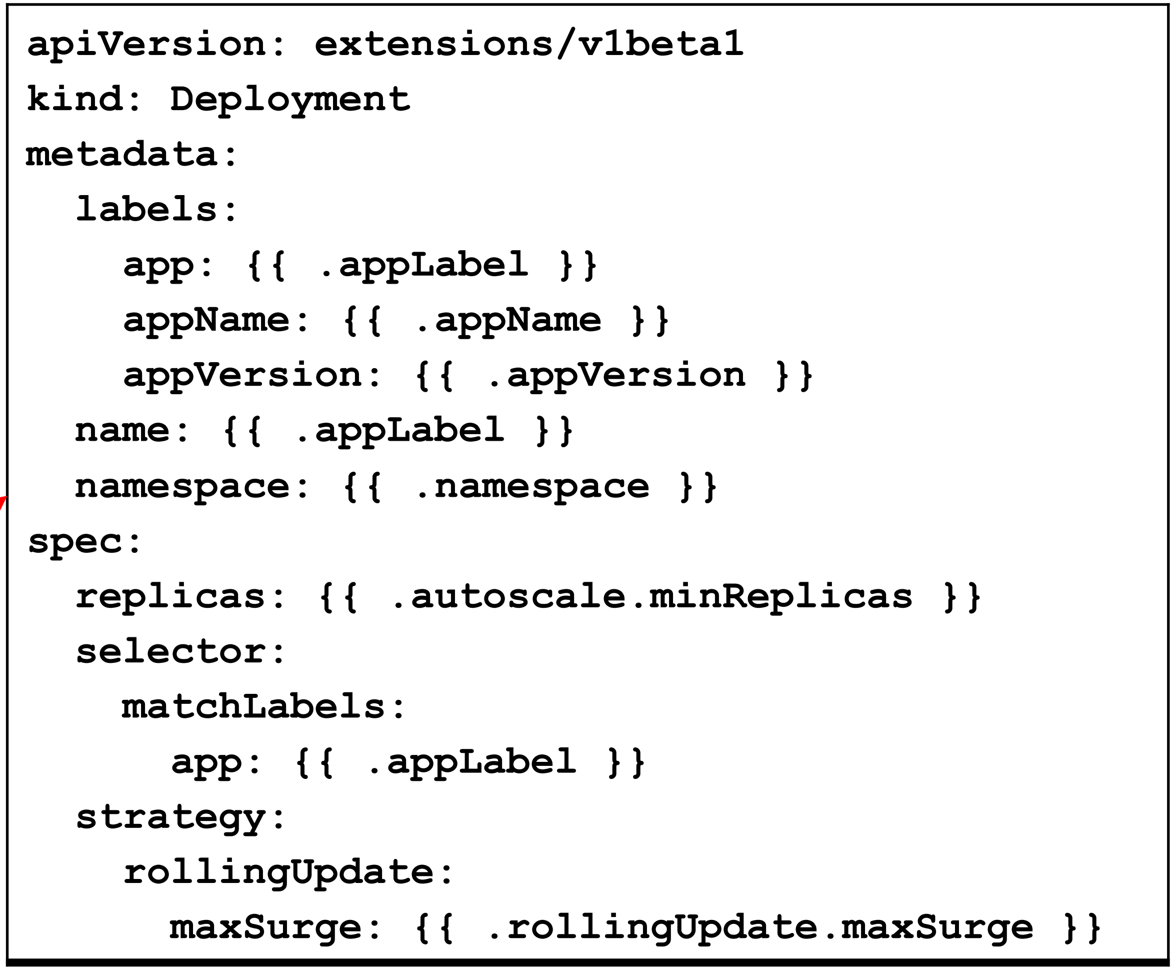
Simplicity: Shared Templates

- Templates owned by central ops team

Template Shared Across Applications



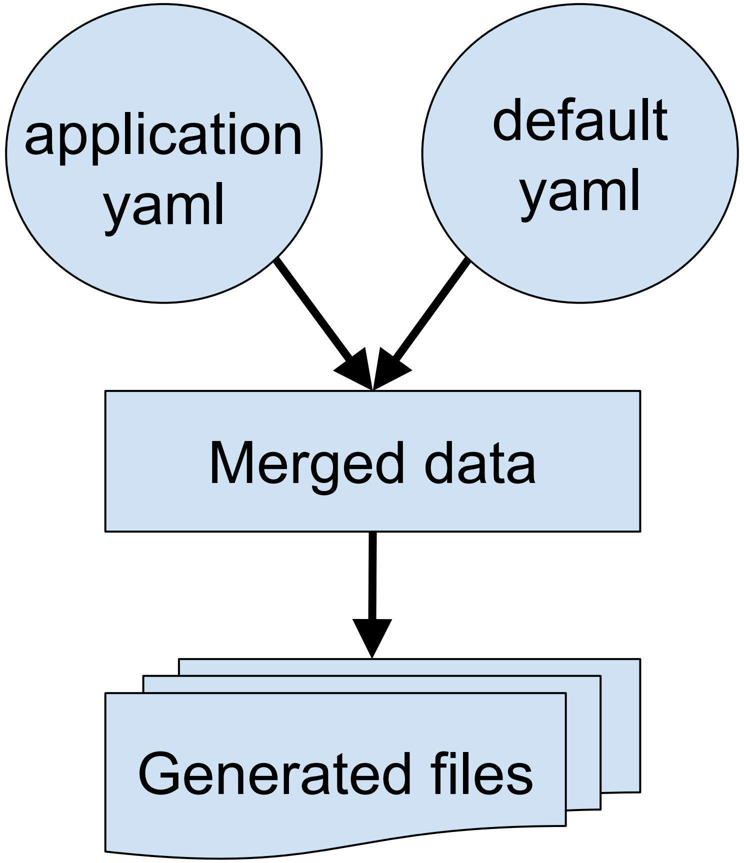
```
|--- templates
|     |--- autoscale.yaml
|     |--- configmap.yaml
|     |--- deployment.yaml
|     |--- ingress.yaml
|     |--- service.yaml
|--- default.yaml
```



```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  labels:
    app: {{ .appLabel }}
    appName: {{ .appName }}
    appVersion: {{ .appVersion }}
  name: {{ .appLabel }}
  namespace: {{ .namespace }}
spec:
  replicas: {{ .autoscale.minReplicas }}
  selector:
    matchLabels:
      app: {{ .appLabel }}
  strategy:
    rollingUpdate:
      maxSurge: {{ .rollingUpdate.maxSurge }}
```

Generation Flow

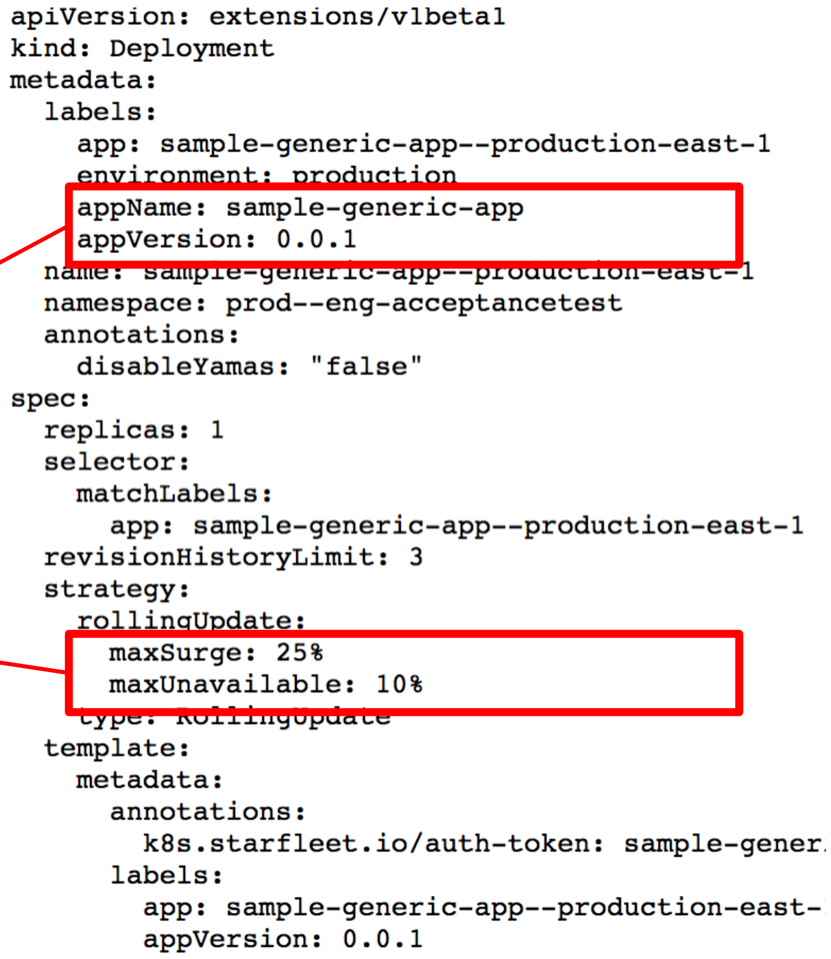
- Application's CD pipeline generates Kubernetes yaml files
  - **configgen -** CLI tool that generates files
  - **application.yaml** - user's settings
  - **default.yaml -** settings from template
    - maxSurge/maxUnavailable
    - cpu/memory



Generated Kubernetes Yaml Files

foo-app-1.0.345.tar

```
generated
|--- Dockerfile
|--- stage
|     |--- autoscale.yaml
|     |--- configmap.yaml
|     |--- deployment.yaml
|     |--- ingress.yaml
|     |--- service.yaml
|--- production-east-1
|     |--- autoscale.yaml
|     |--- configmap.yaml
|     |--- deployment.yaml
|     |--- ingress.yaml
|     |--- service.yaml
|--- production-west-1
|     |--- autoscale.yaml
|     |--- configmap.yaml
|     |--- deployment.yaml
|     |--- ingress.yaml
|     |--- service.yaml
```



```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  labels:
    app: sample-generic-app--production-east-1
    environment: production
    appName: sample-generic-app
    appVersion: 0.0.1
  name: sample-generic-app--production-east-1
  namespace: prod--eng-acceptancetest
  annotations:
    disableYamas: "false"
spec:
  replicas: 1
  selector:
    matchLabels:
      app: sample-generic-app--production-east-1
  revisionHistoryLimit: 3
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 10%
    type: RollingUpdate
  template:
    metadata:
      annotations:
        k8s.starfleet.io/auth-token: sample-gener
      labels:
        app: sample-generic-app--production-east-
        appVersion: 0.0.1
```

Flexibility

- Users can override default settings

application.yaml

```
# shared settings
template: nodejs-app:stable
appName: foo-app
maintainer: foo-team@oath.com
baseimage: ynodejs:6.0.170725-3
routing:
  ports: 4080
athensDomain: foo.domain
sidecars:
  splunk:
    instance: foo.splunk:443
  metrics:
    namespace: foo
```

```
# environment-specific settings
jobs:
  stage:
    environment: stage
    colo: gq1
  production-east-1:
    environment: production
    colo: bf1
    autoscale:
       minReplicas: 10
       maxReplicas: 25
  production-west-1:
    environment: production
    colo: gq1
```

Stability

- Stable Tags for Templates
- Acceptance Tests
- Controlled Configurations
- UI For Validating

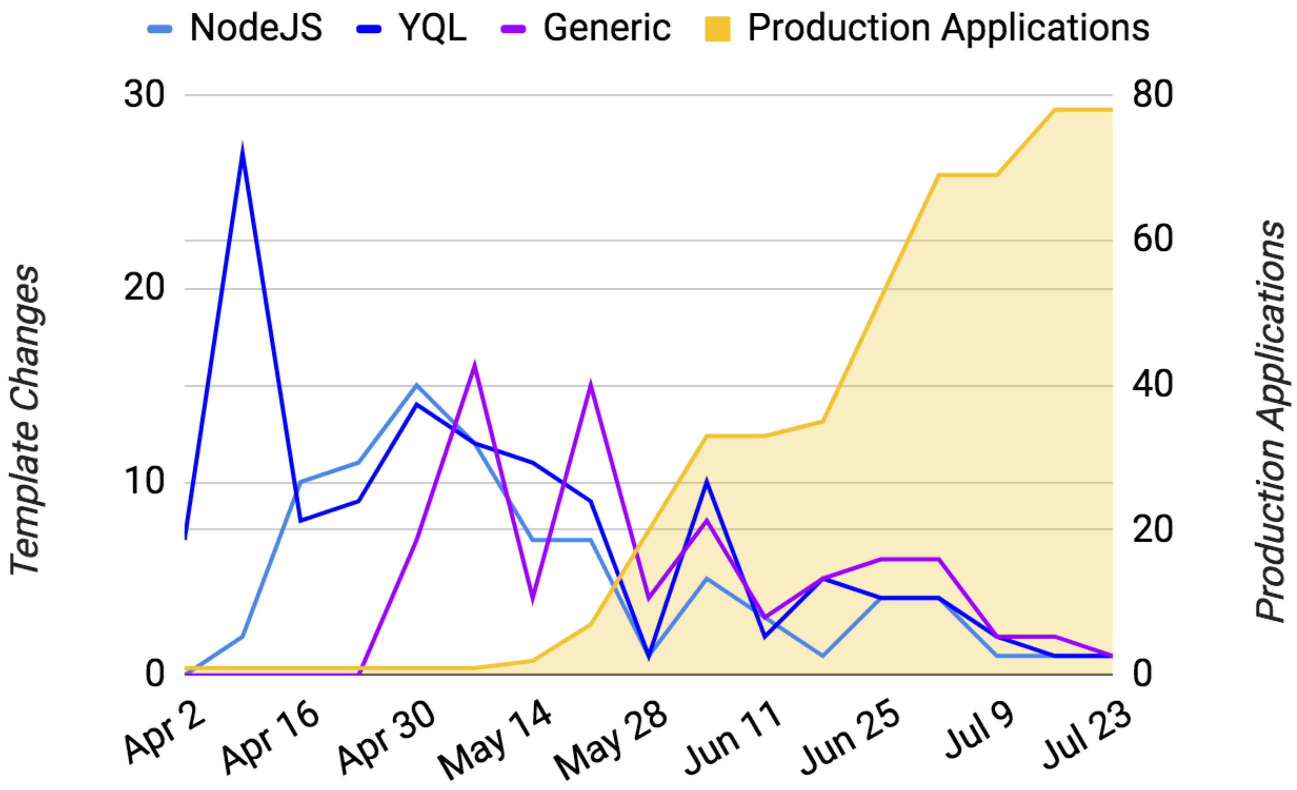
  application.yaml

Application onboarding

- CLI to create application.yaml
- Consistent Configuration
  - Easy to troubleshoot and fix issues
  - Global Fix

Results

- Stability Despite Changes
  - Applications not disrupted with templates changes



Results

- Rapid Onboarding
  - Total time to build system and onboard apps: **3 months**
  - Applications: **~100**
  - Production Peak RPS: **185k**
- Avoided Matrix Hell
  - Total files eliminated: **~30 per app, ~3000 total**
  - Config files: **1 per app**

Do you have any questions?