



Stop being a YAML engineer, be a software engineer with CDK8s

Robert Hoffmann
Senior Solutions Architect @ AWS

Twitter: @robhoffmax



CDK For Kubernetes (CDK8s)

An open source multi-language software development framework for modeling Kubernetes resources as reusable components



Go from Code to Config

Define Kubernetes applications and architectures using familiar programming languages.



Cut Copy & Paste

Turn best practices into code libraries and share easily.



Run Everywhere

cdk8s runs locally and generates YAML you can deploy to any cluster, anywhere.



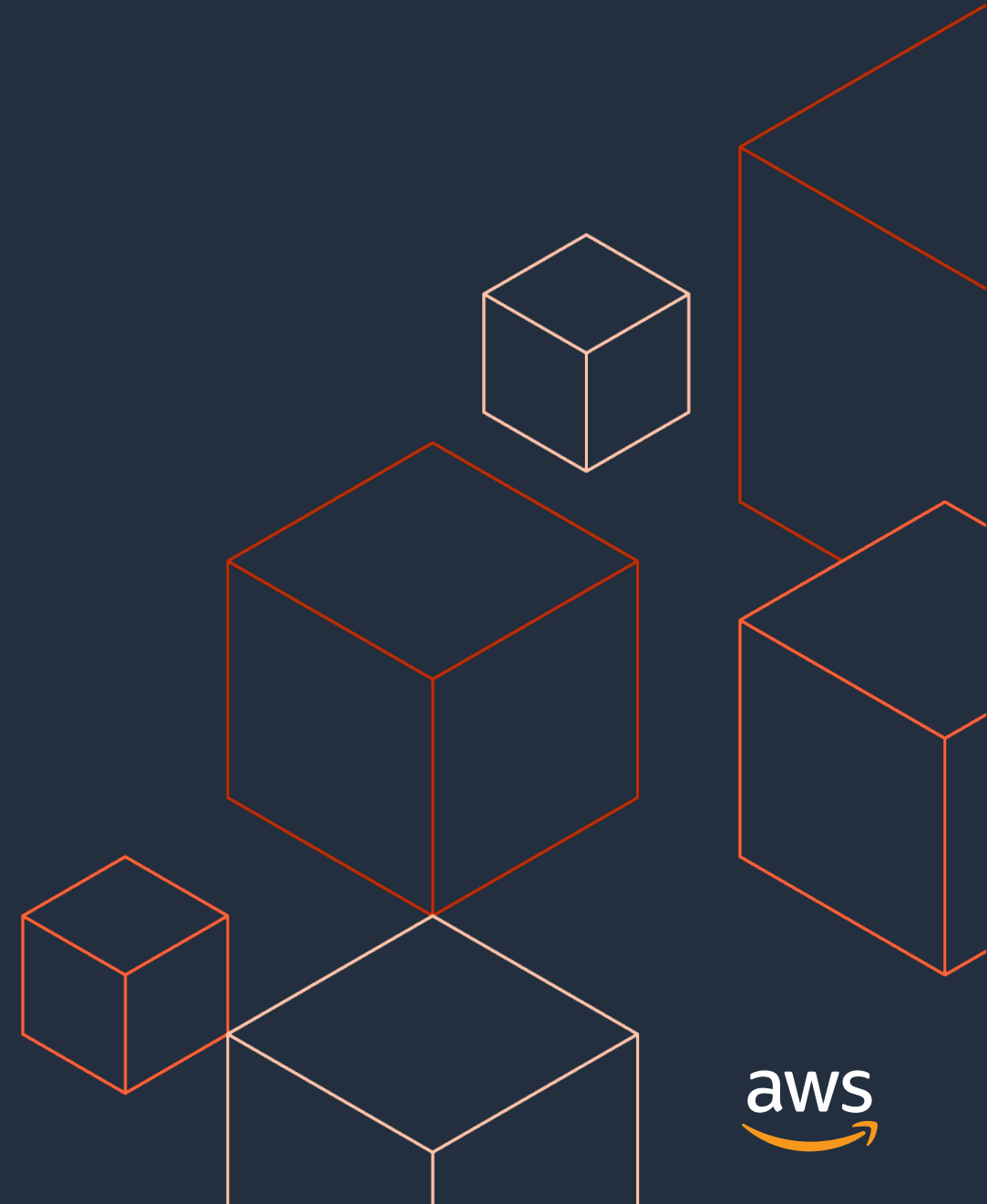
YAML



YAML:

- Y: Yelling
- A: At
- M: My
- L: Laptop

Let's engineer some YAML



Http Echo Server – with YAML

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: echo-deployment
spec:
  selector:
    matchLabels:
      app: echo
  template:
    metadata:
      labels:
        app: echo
    spec:
      containers:
        - args:
            - -text
            - hello
          image: hashicorp/http-echo
          name: main
          ports:
            - containerPort: 5678
```

```
apiVersion: v1
kind: Service
metadata:
  name: echo-service
spec:
  ports:
    - port: 5678
      targetPort: 5678
  selector:
    app: echo
```

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: echo-ingress
spec:
  rules:
    - http:
        paths:
          - backend:
              service:
                name: echo-service
                port:
                  number: 5678
            path: /hello
            pathType: Prefix
```

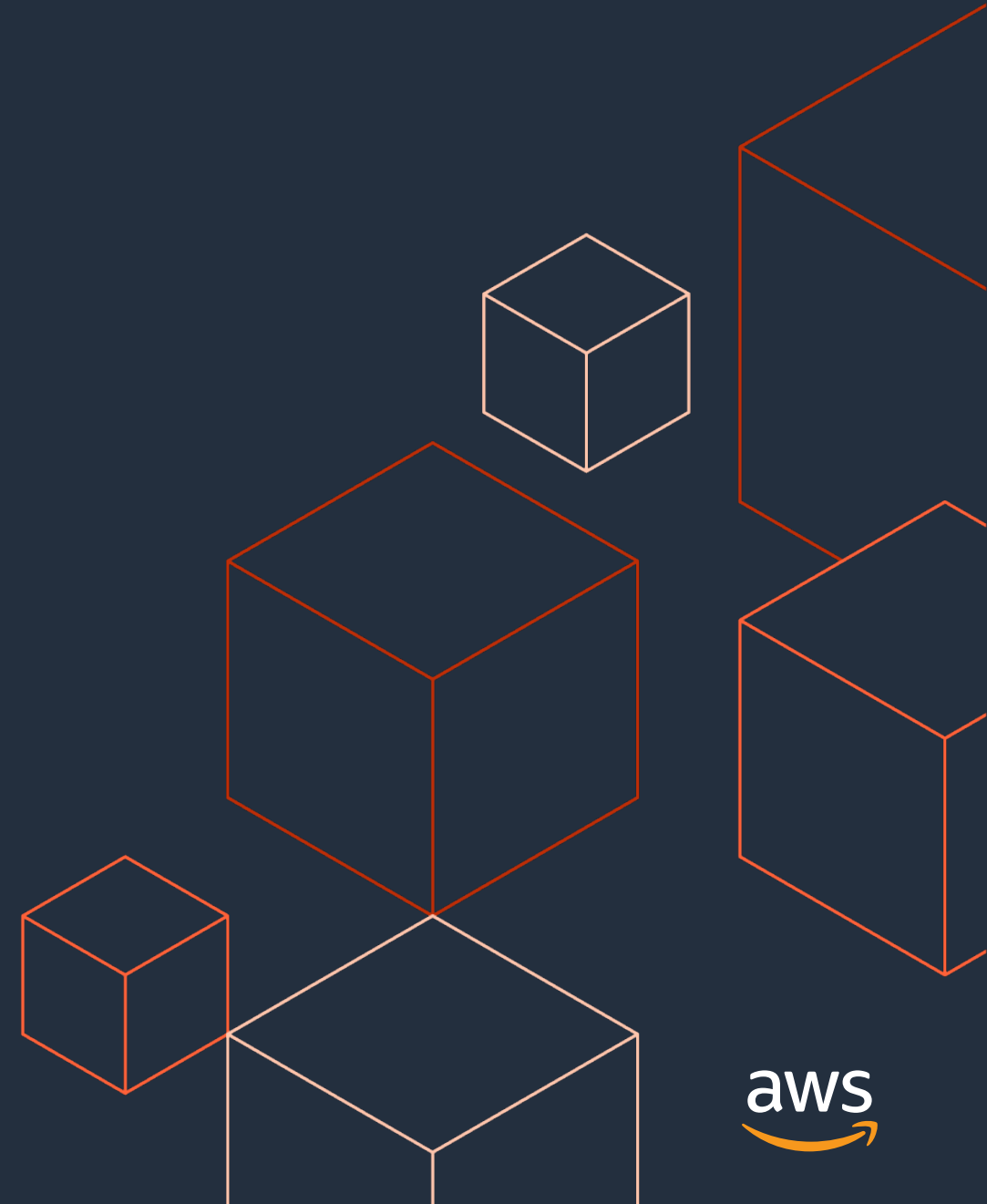
~ 45 lines of YAML

YAML

- Static
- Promotes Copy/Paste
- Hard to customize and share

In addition, manifests carry a fairly high cognitive load.

How CDK8s can help



Motivation

Authoring Kubernetes manifests can be a rather complicated task.

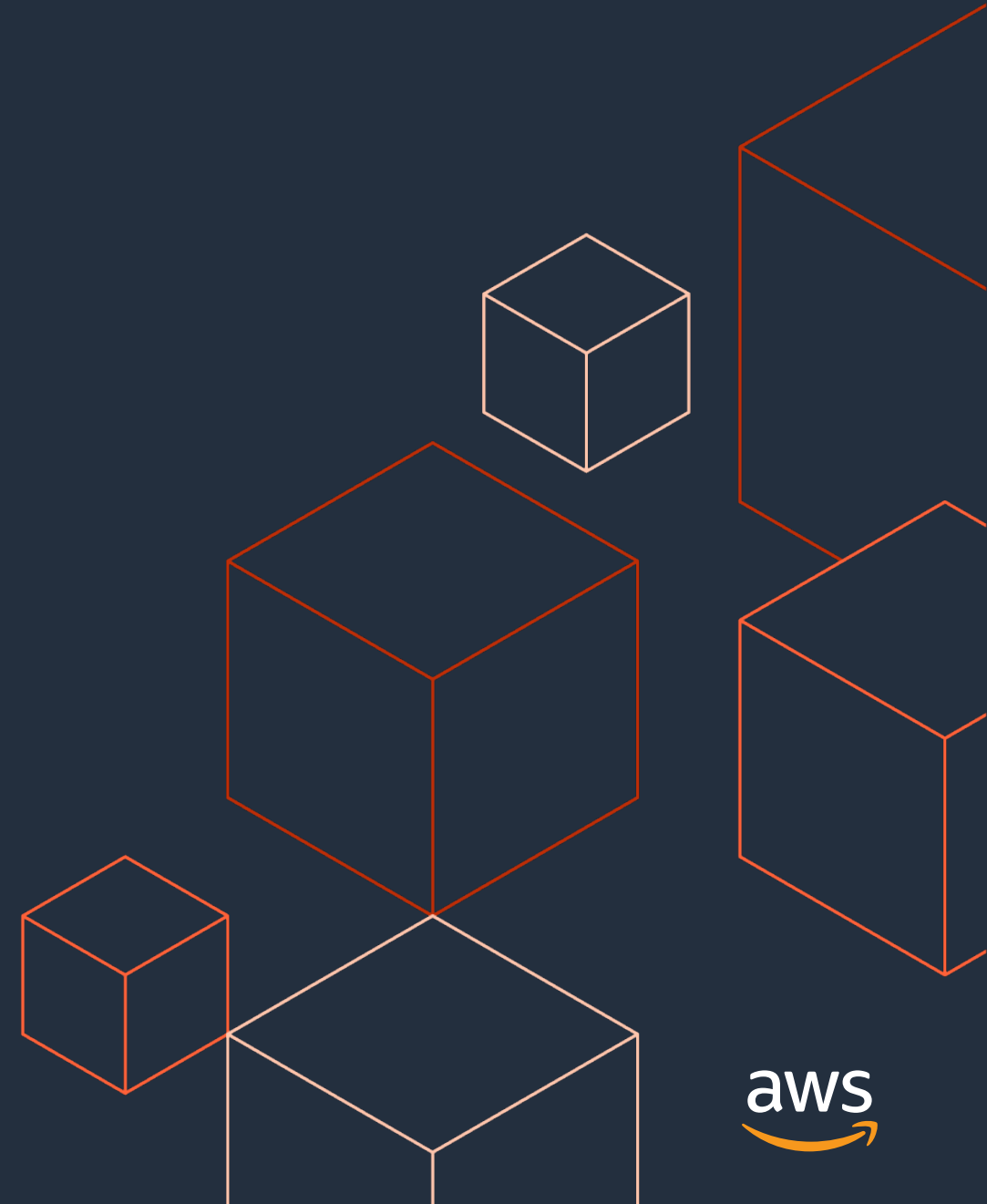
CDK8s aims to make authoring Kubernetes manifests simpler, and more robust.

How? By leveraging general purpose programming languages and standard software engineering practices.

General purpose programming languages

- Dynamic
- DRY
- Standard package managers
- Abstractions

Working with CDK8s



Workflow: Initialize your project



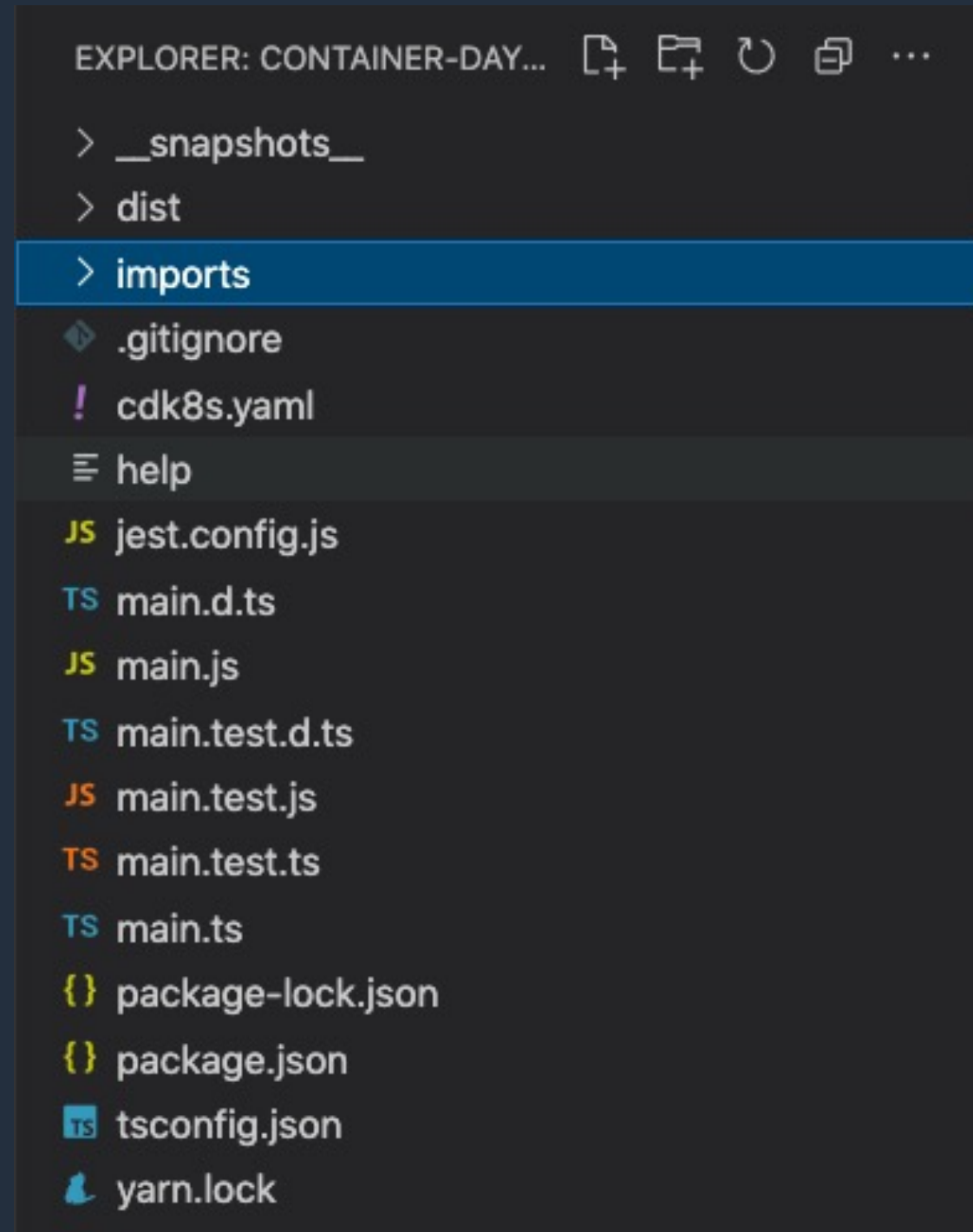
`npm install -g cdk8s-cli`

Project website: <https://cdk8s.io/>

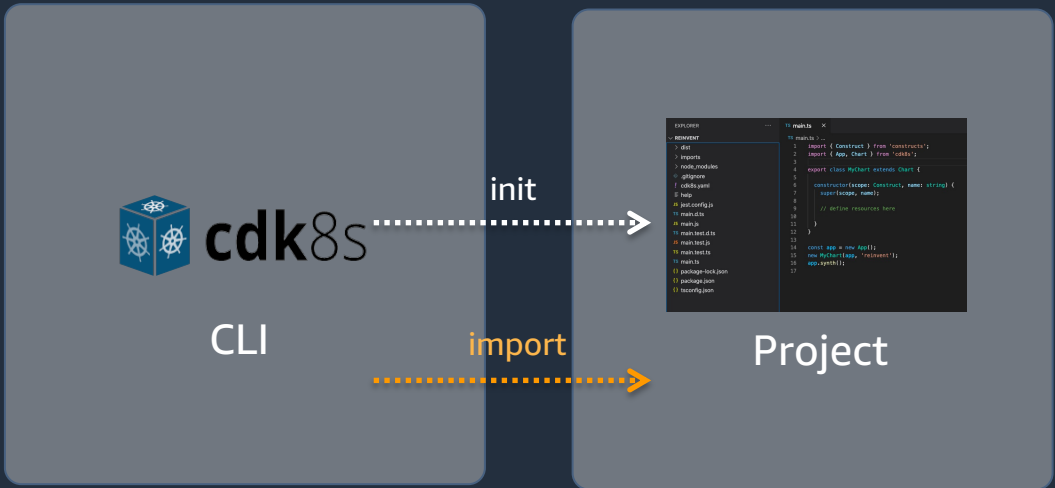
CDK8s CLI: Initialize your project

```
cdk8s init typescript-app  
go-app  
java-app  
python-app
```

Create a new cdk8s project from a template.



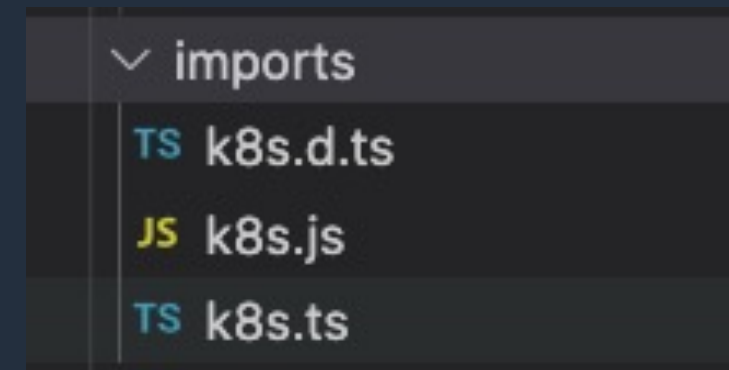
Workflow: Import API objects



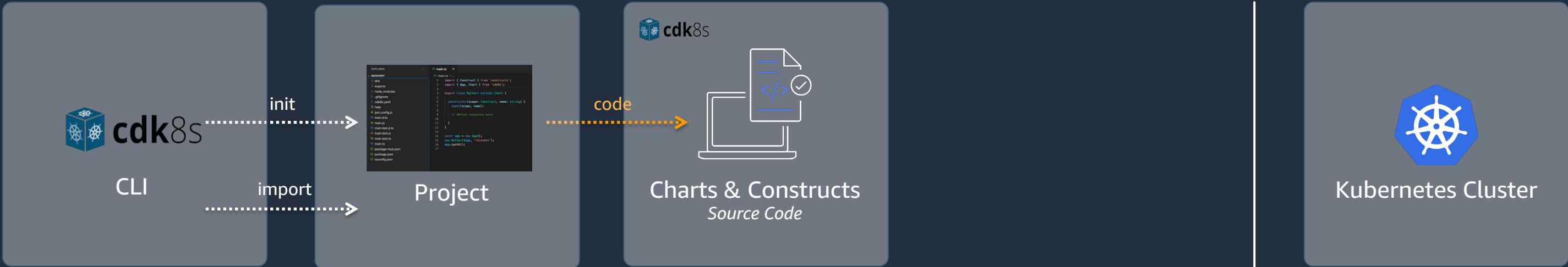
CDK8s CLI: Import API objects

```
cdk8s import k8s@1.20.0  
             jenkins.io_jenkins_crd.yaml
```

Imports API objects to your app by
generating constructs.



Workflow: Model your k8s objects with code



Code completion and inline documentation

```
new k8s.KubeDeployment(this, 'Deployment', {  
  spec: {  
    selector (property) DeploymentSpec.selector: k8s.Labe...  
    template  
    minReadySeconds?  
    paused?  
    progressDeadlineSeconds?  
    replicas?  
    revisionHistoryLimit?  
  }  
})
```

```
(property) DeploymentSpec.selector: k8s.LabelSelector  
Label selector for pods. Existing ReplicaSets whose pods are selected by this will be the  
ones affected by this deployment. It must match the pod template's labels.  
  
@schema — io.k8s.api.apps.v1.DeploymentSpec#selector  
readonly selector: LabelSelector;  
new  
  sp selector: {}  
  }  
})
```


Http Echo Server – with CDK8s

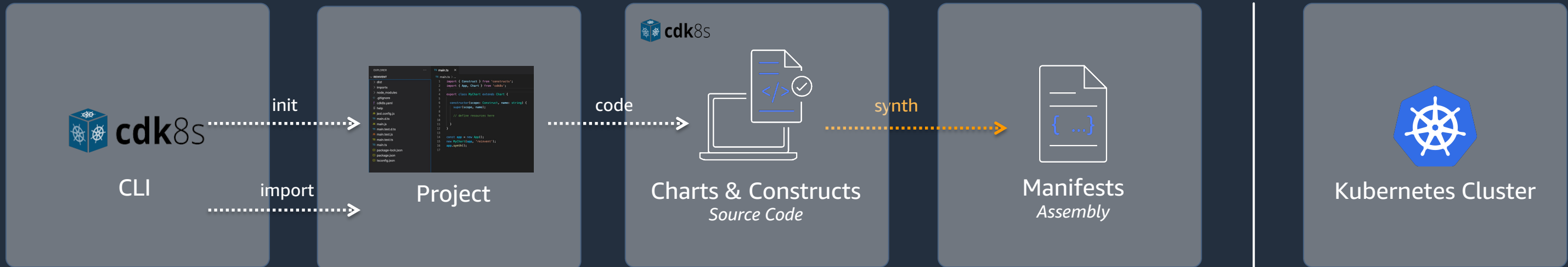
```
const port = 5678
const labels = { app: 'echo' }
```

```
new k8s.KubeDeployment(this, 'Deployment', {
  spec: {
    selector : {matchLabels: labels},
    template: {
      metadata: {labels : labels},
      spec: {
        containers: [{
          name: 'main',
          image: 'hashicorp/http-echo',
          args: ['-text', 'hello'],
          ports: [{containerPort: port}],
        }]
      }
    }
  }
})
```

```
const service = new k8s.KubeService(this, 'Service', {
  spec: {
    ports: [{
      port: port,
      targetPort: k8s.IntOrString.fromNumber(port)
    }],
    selector: labels
  }
})
```

```
new k8s.KubeIngress(this, 'Ingress', {
  spec: {
    rules: [{
      http: {
        paths: [{
          pathType: 'Prefix',
          path: '/hello',
          backend: {
            service: {
              name: service.name,
              port: {
                number: port
              }
            }
          }
        }]
      }
    }]
  }
})
```

Workflow: Synthesize the k8s manifests

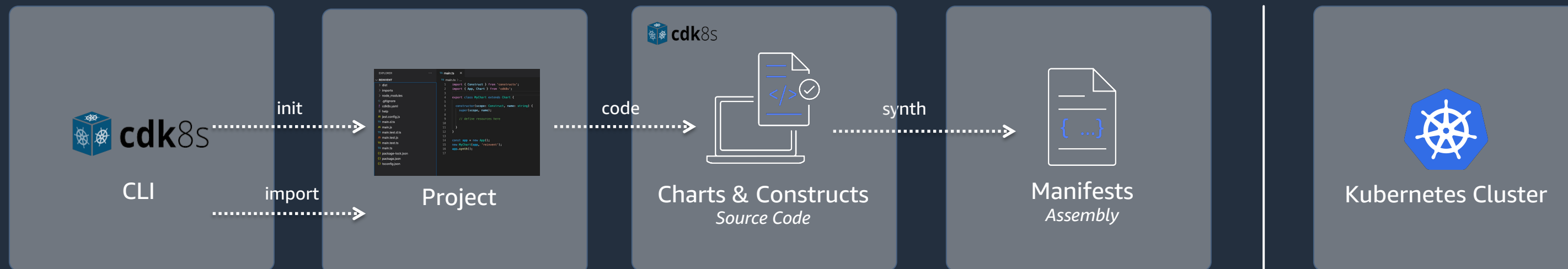


cdk8s synth

Synthesizes Kubernetes manifests for all charts in your app.

Workflow: Deploy the manifests

Kubectl/GitOps

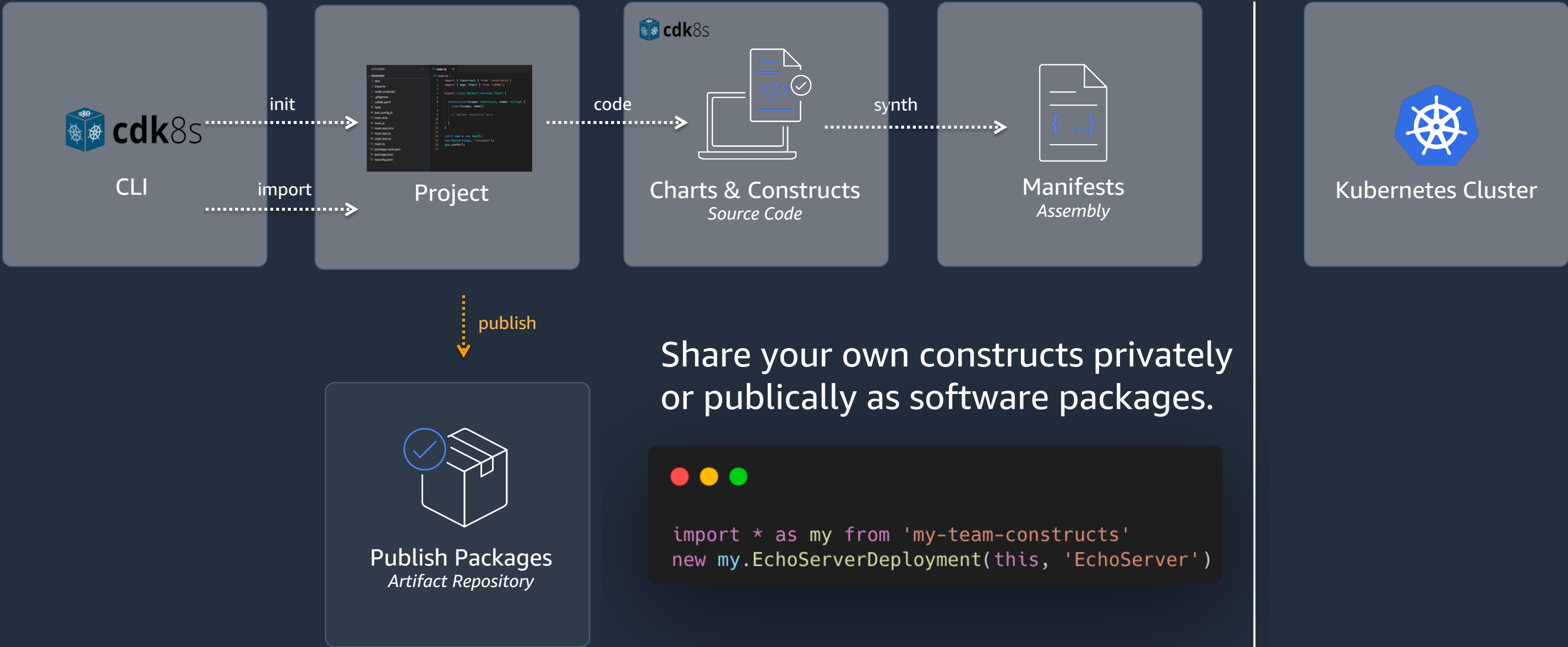


```
kubectl apply -f /dist/manifest.yaml
```

Deploy the Kubernetes manifests with your favorite tools.

Publish your code as a re-usable package

Kubectl/GitOps



How CDK8s+ can help even more



Construct Levels

L2

`kplus.Deployment`

Hand crafted high level APIs

L1

`k8s.KubeDeployment`

Automatically generated

L0

`cdk8s.ApiObject`

Common Functionality

Http Echo Server – with CDK8s+

```
const port = 5678

const deployment = new kplus.Deployment(this, 'Deployment', {
  containers: [
    {
      image: 'hashicorp/http-echo',
      args: [ '-text', 'hello' ],
      port: port
    }
  ]
})

deployment.exposeViaIngress('/hello', {
  pathType: kplus.HttpIngressPathType.PREFIX
})
```

~ 13 lines of code

Declare your intent!
Write for humans, not machines.

Clean Code

Code is clean if it can be understood easily – by everyone on the team.

Clean code can be read and enhanced by a developer other than its original author.

With understandability comes readability, changeability, extensibility and maintainability.

– Robert C. Martin, and others


```

const port = 5678

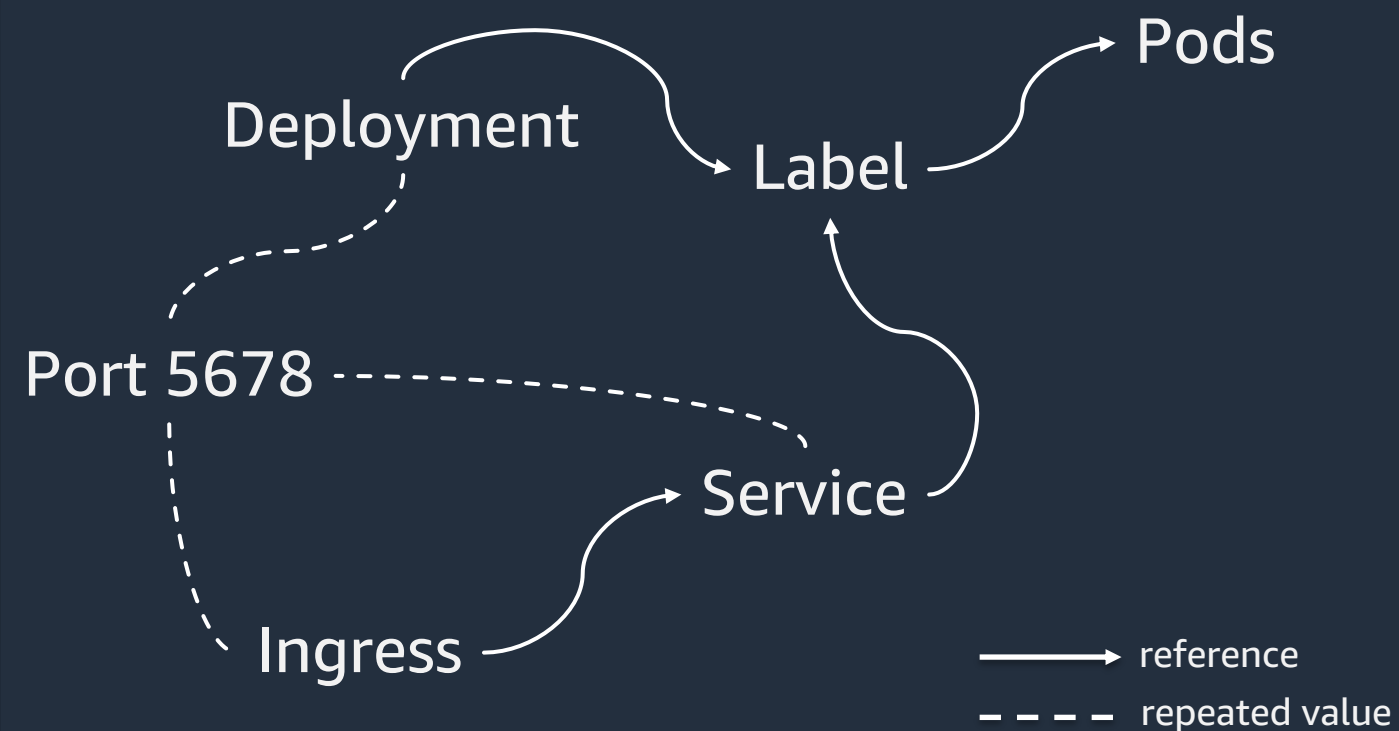
const deployment = new kplus.Deployment(this, 'Deployment', {
  containers: [
    {
      image: 'hashicorp/http-echo',
      args: [ '-text', 'hello' ],
      port: port
    }
  ]
})

deployment.exposeViaIngress('/hello', {
  pathType : kplus.HttpIngressPathType.PREFIX
})

```

The story:

“I want to create a deployment and expose it via ingress.”



Under the hood:

CDK8s creates and wires multiple resources for you, taking care of technical details if possible.

Why should I code my infrastructure?

- Familiar programming languages
- Great IDE / tool support
- Powerful abstractions with CDK8s+
- Software engineering practices: Clean Code!

Why not?

- You want to keep your existing tools (you could mix & match)
- You don't like coding
- You are the author of a Helm chart

Resources

Getting started with CDK8s

- Project website: <https://cdk8s.io/>
- Find constructs published by the open-source community: <https://constructs.dev/>

Interact with the community

- CNCF Sandbox project website: <https://www.cncf.io/projects/cdk-for-kubernetes-cdk8s/>
- Slack: <https://cdk.dev/>
- The community event for the whole CDK ecosystem: <https://www.cdkday.com/>

Thank you!

Robert Hoffmann

Twitter: [@robhoffmax](https://twitter.com/robhoffmax)