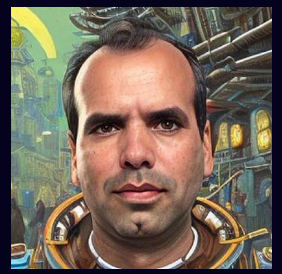
Control Planes on Kubernetes and Policy Validation



Carlos Santana

Sr. Specialist Solutions Architect, Kubernetes

csantanapr

@csantanapr

Amazon EKS is the most trusted and secure way to run Kubernetes

Amazon **EKS**

- EKS runs vanilla Kubernetes. EKS is upstream and certified conformant version of Kubernetes (with backported security fixes)
- EKS supports 5 versions of Kubernetes, giving customers time to test and roll out upgrades.
- EKS provides a managed Kubernetes experience for performant, reliable, and secure Kubernetes.
- EKS makes Kubernetes operations, administration, and management simple and boring.

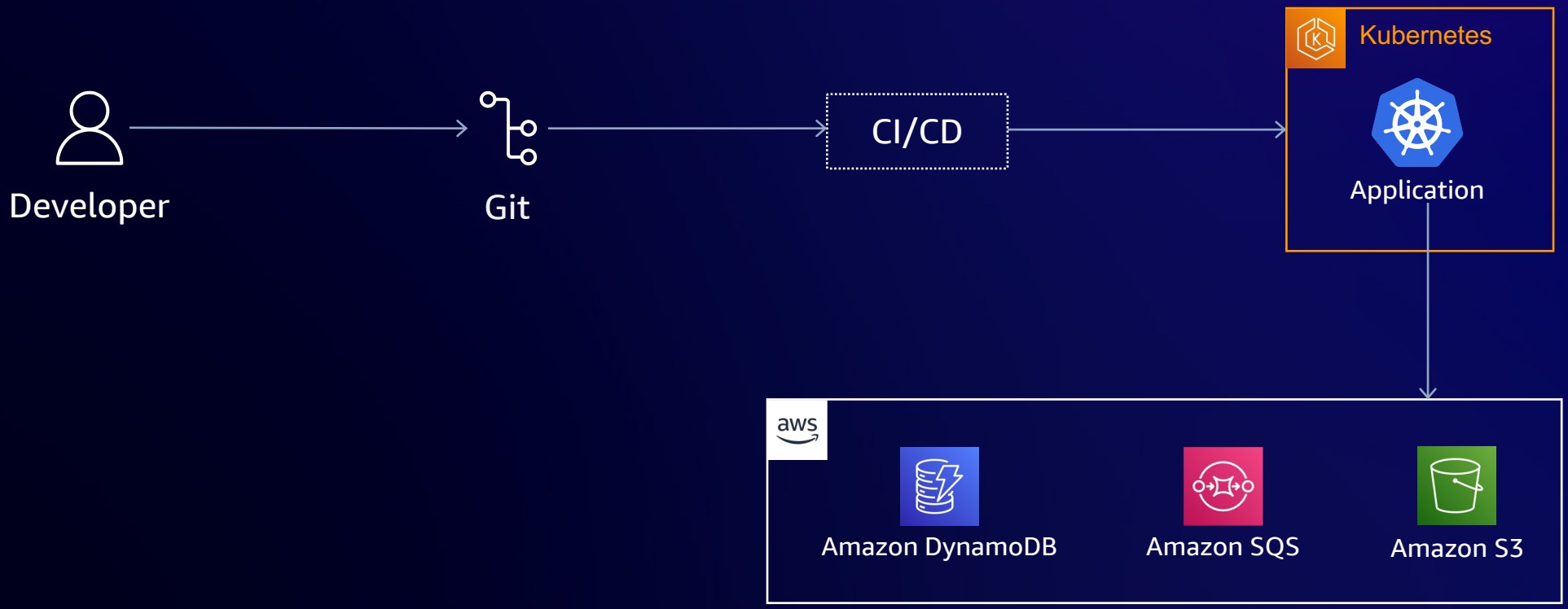
AWS is the best place to run Kubernetes. 65% of organizations choose AWS to run their containers.

CNCF State of Cloud Native Development

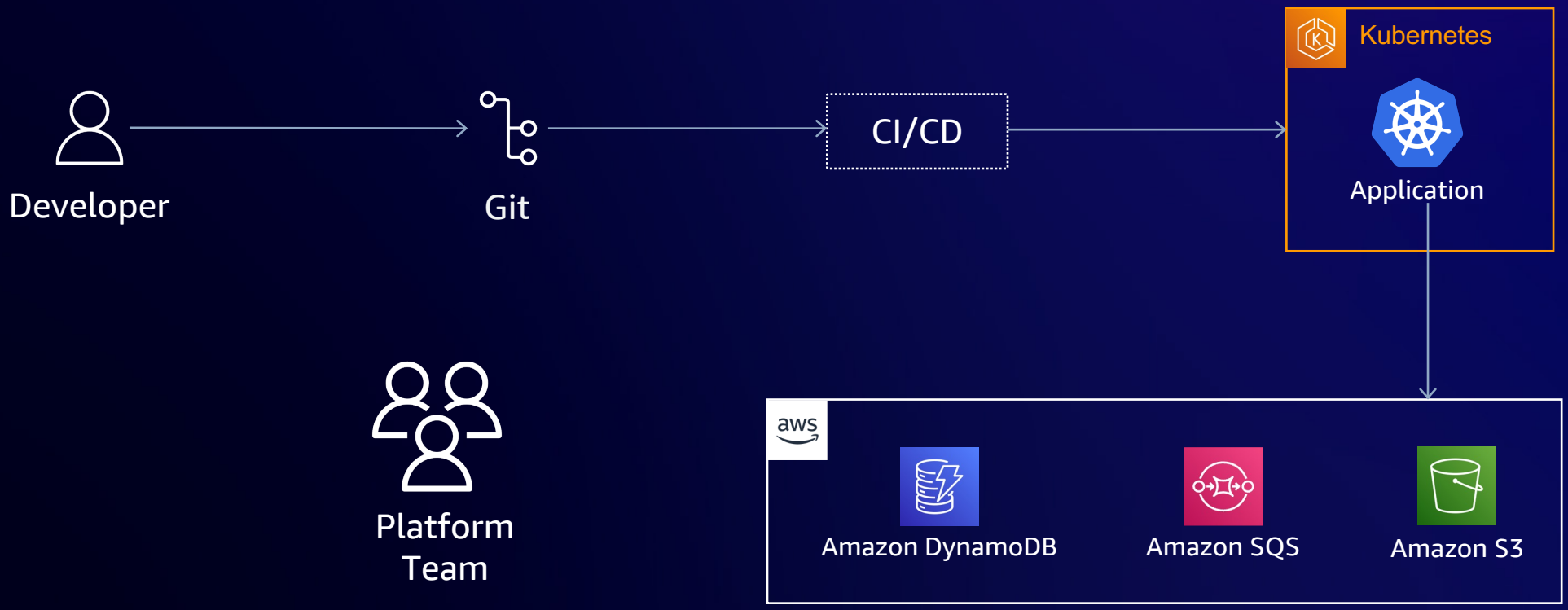
aws

Deploy Kubernetes Applications

Deploy Kubernetes Applications



Deploy Kubernetes Applications



Separation of Concerns

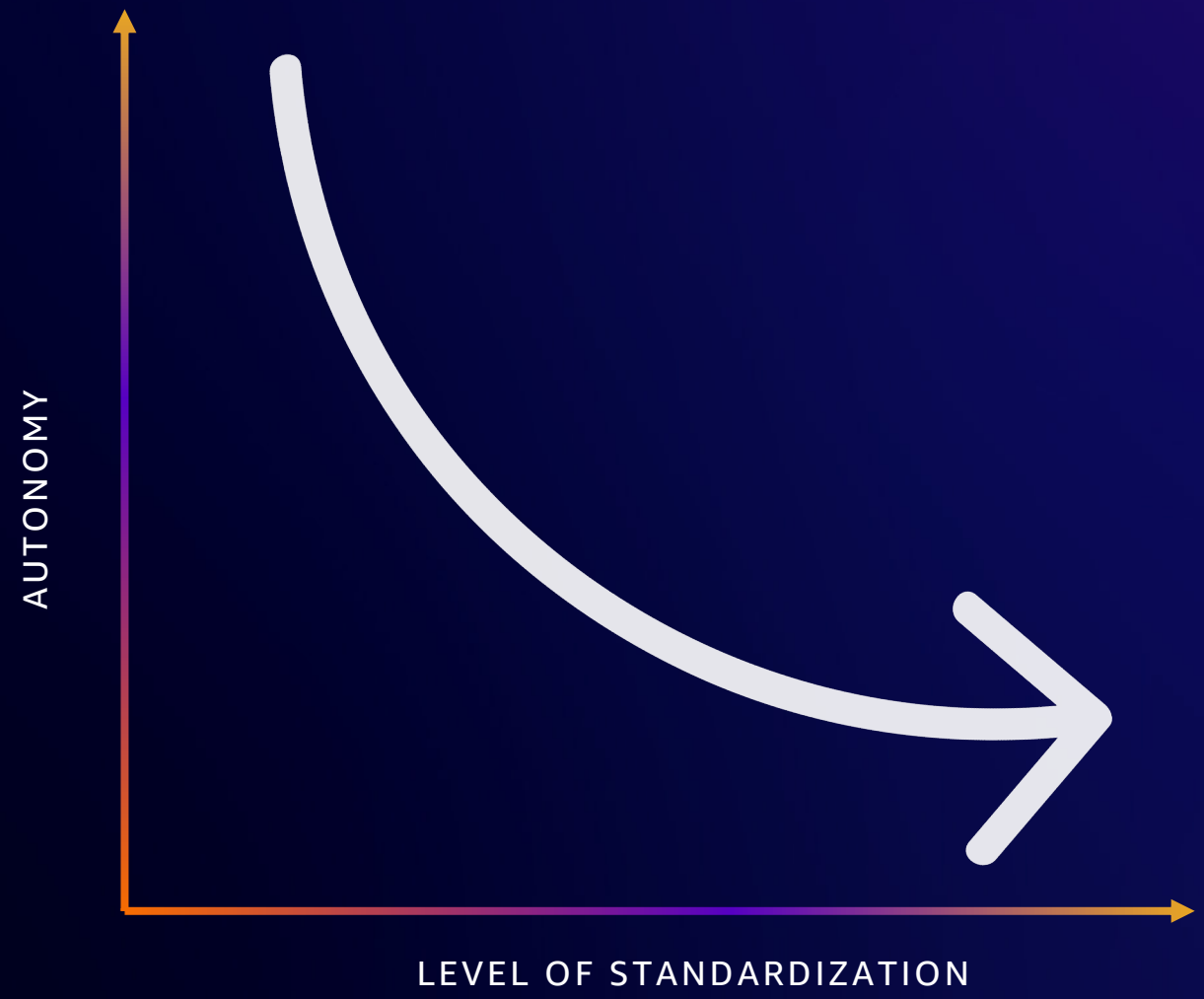
Platform Team

I would like to **standardize** the deployment process for application teams while **enforcing** organizational standards for **cloud**.

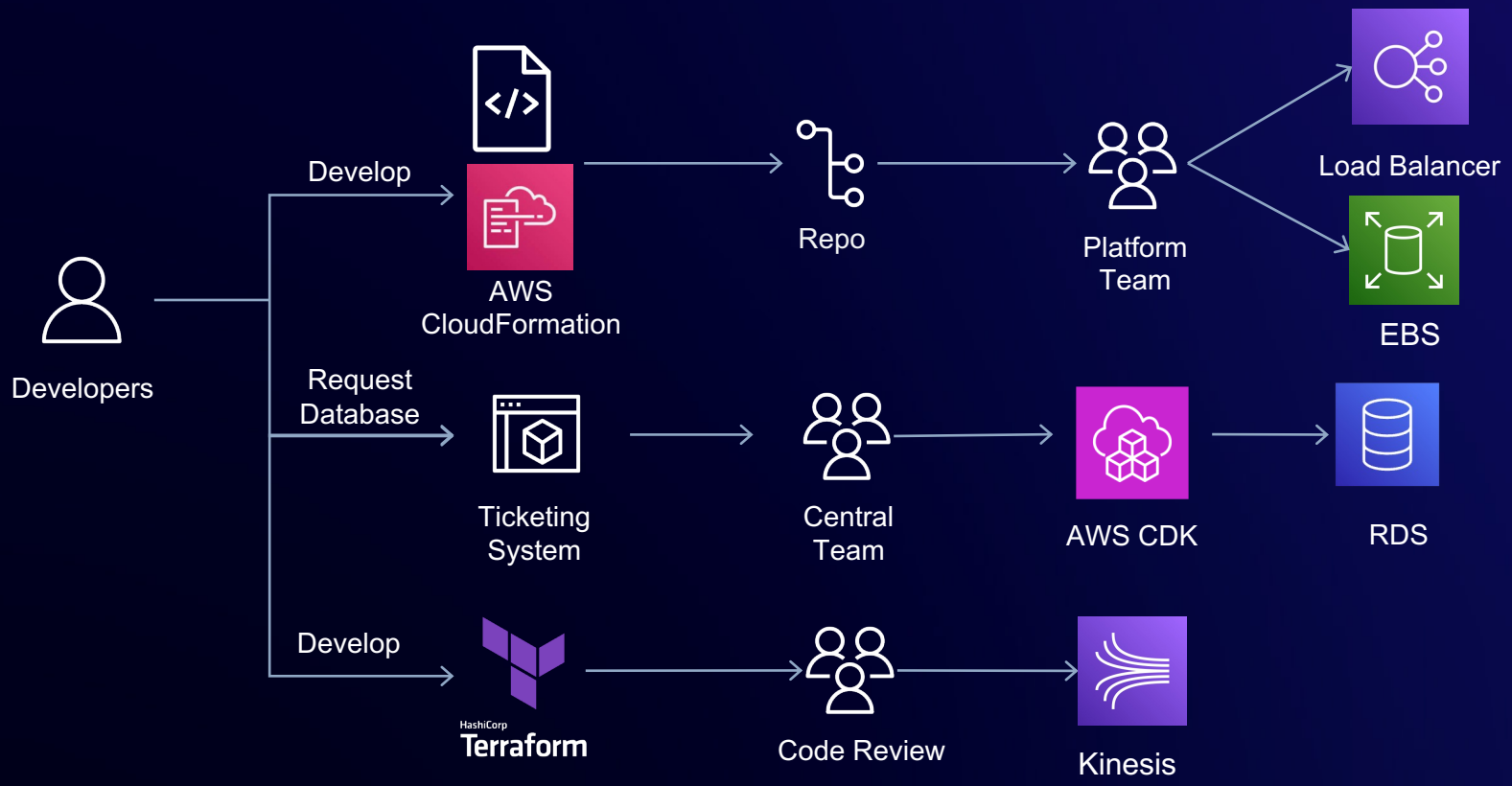
Application Teams

I would like to have **full autonomy** of **applications** and its **cloud dependencies** deployment lifecycle.

Autonomy vs Standardization



Disjointed Workflows Adds Developer Overhead



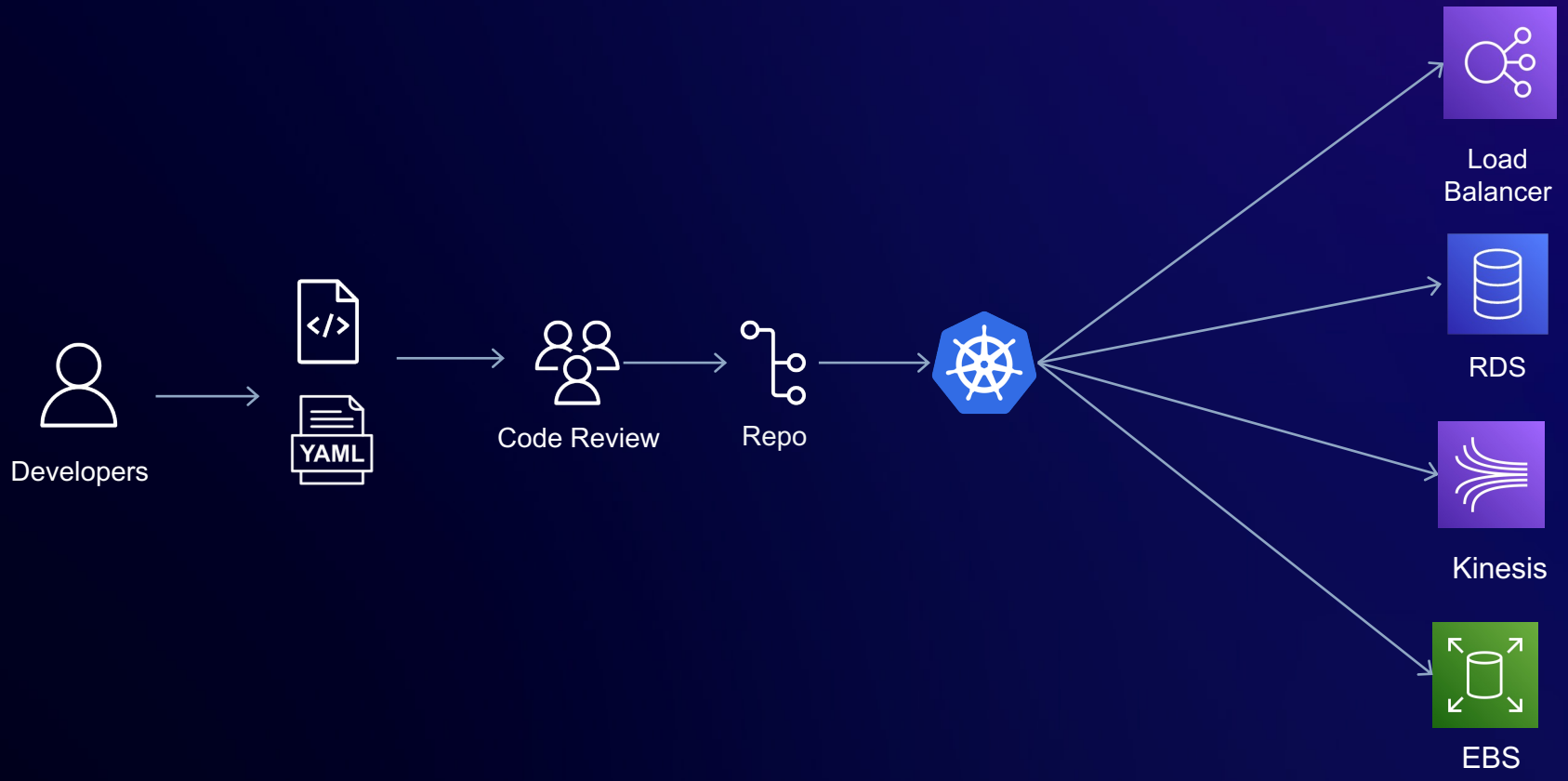
How do you get the state of the entire system?

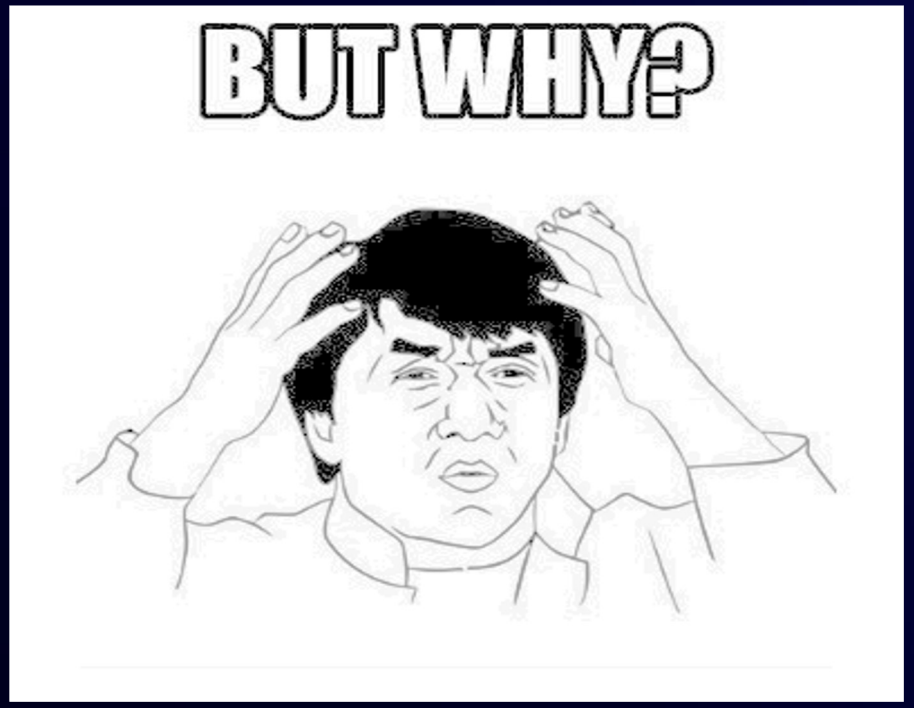
How do you debug and remediate issues?

How do you pass secrets from one pipeline passed to the other?

How do you tear-down *all* unused resources?

Kubernetes to Deploy Cloud Services



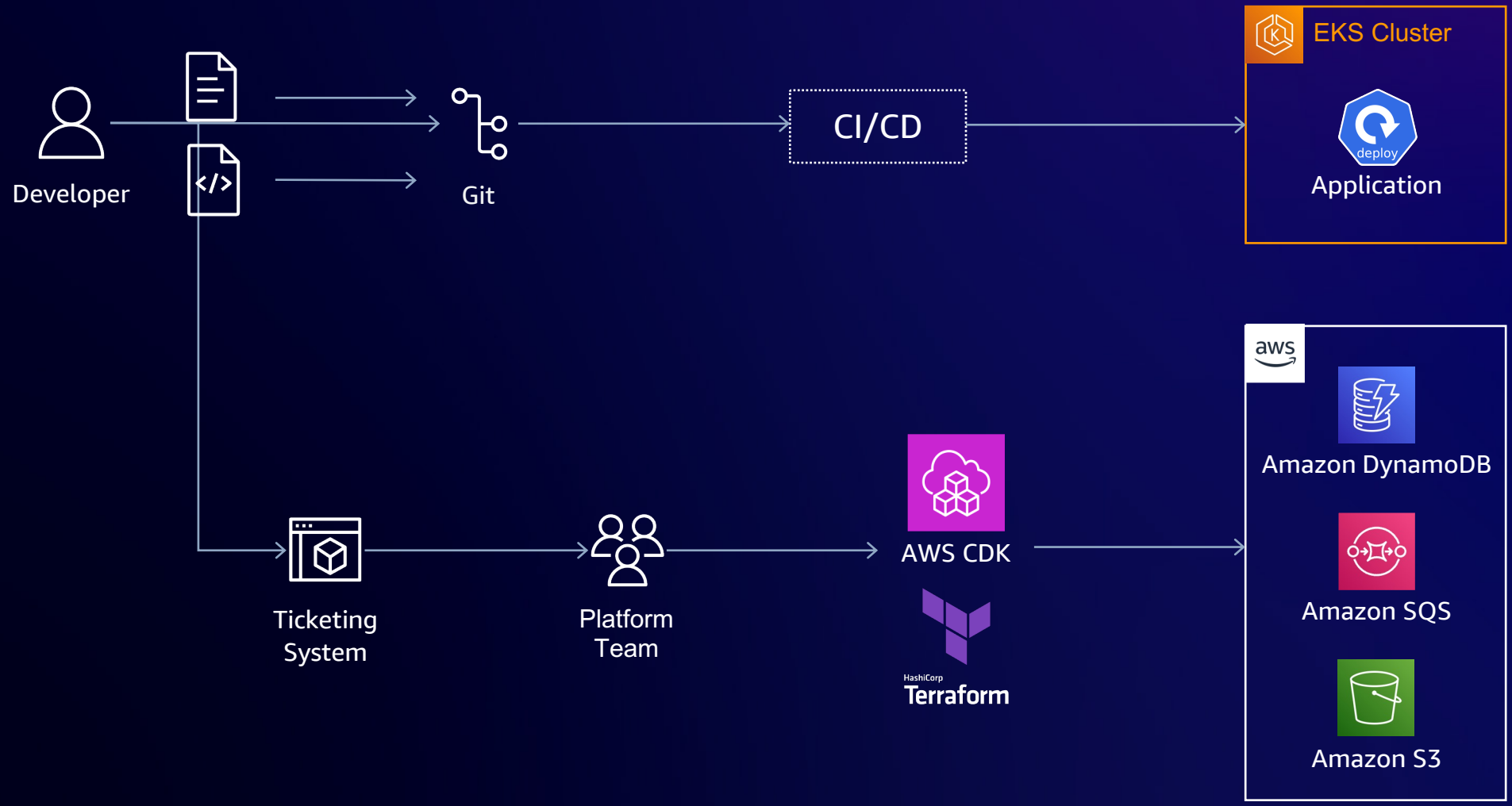


Leverage existing workflows for deploying into Kubernetes without maintaining several pipelines

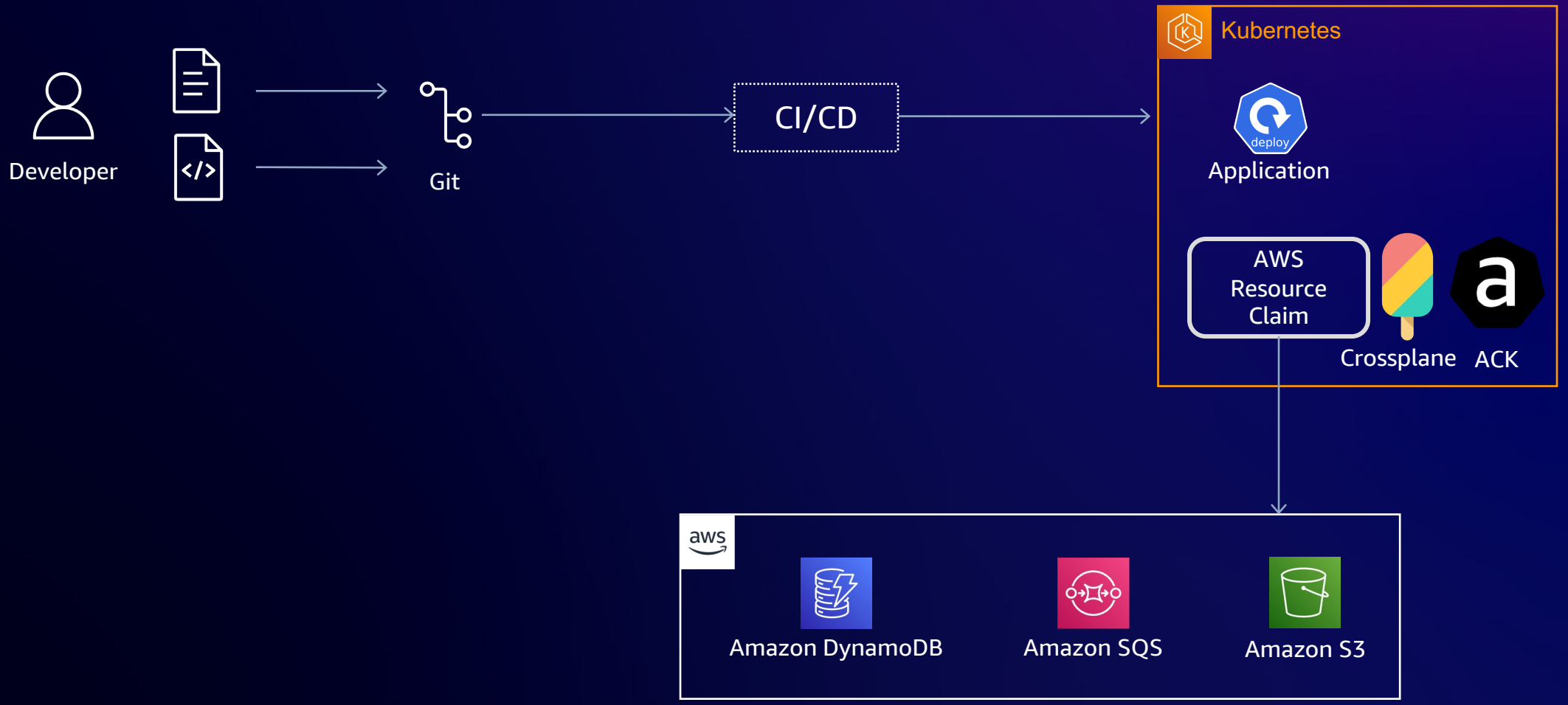
Centralized cost, security, and auditing for any cloud resource with OPA or Kyverno, utilize existing RBAC and IRSA

Self healing using GitOps for all cloud resources (containers, databases, queues, and streams)

Kubernetes Infrastructure Controller



Kubernetes Infrastructure Controller



Infrastructure controllers

Manage cloud services using Kubernetes API

A single API for Kubernetes and cloud services

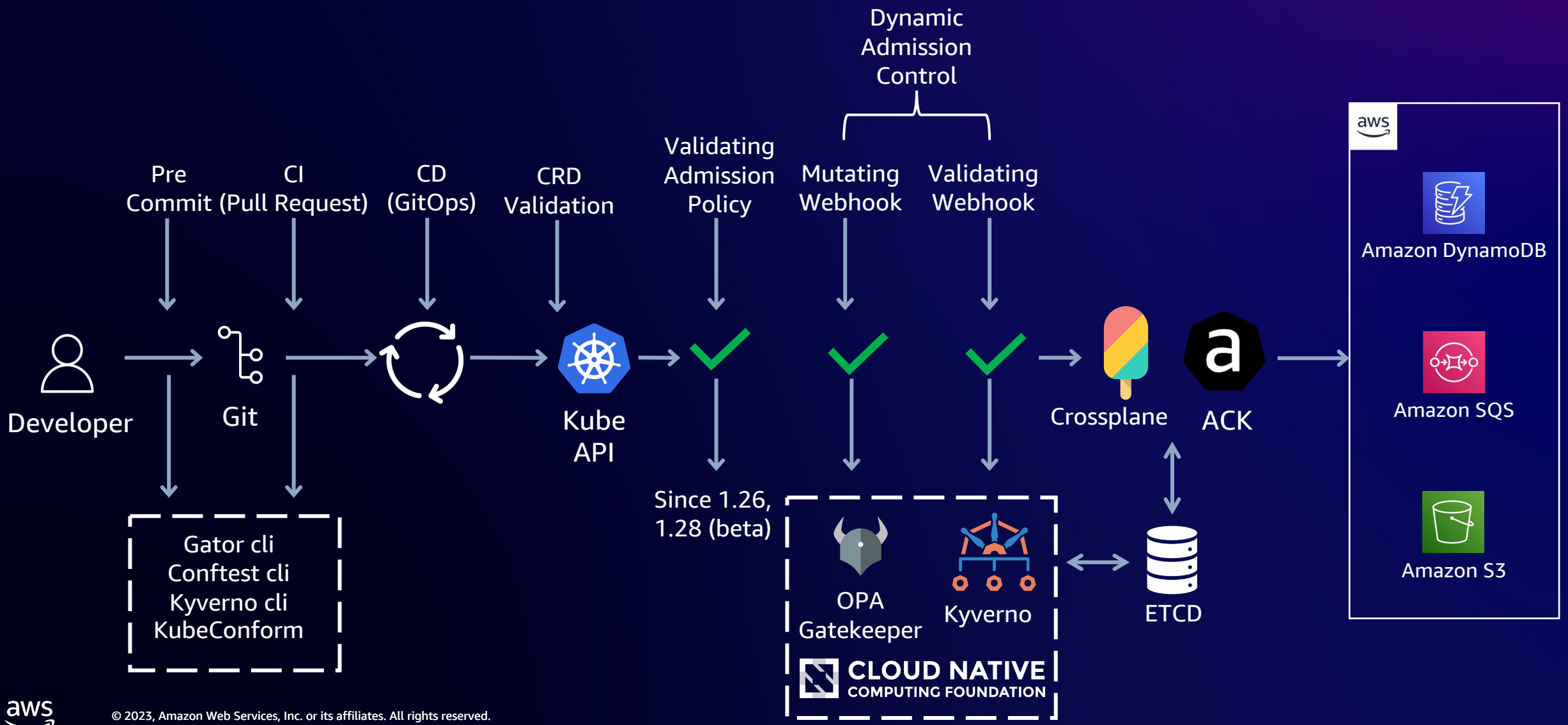
Create your own platform API

Declarative infrastructure configuration

ACK

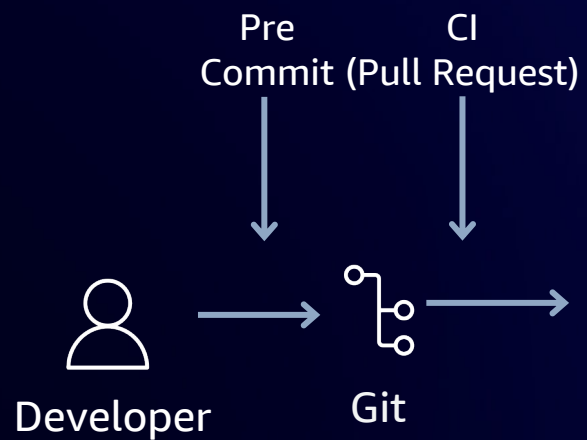
Crossplane

Where to check?



Local Testing

- Rego
  - opa
  - gator
  - conftest
- Kyverno
- Kubeconform

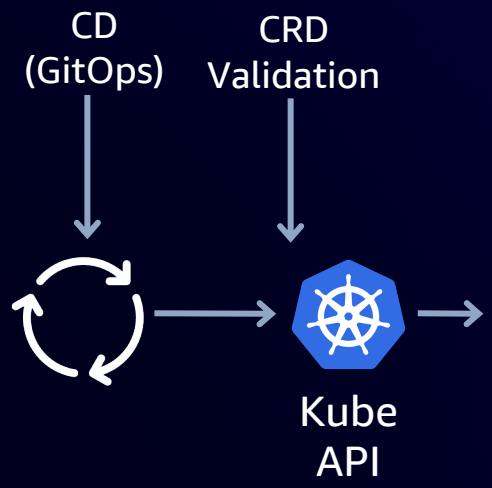


Kubernetes CRD

- CRD Validation Expression
- Common Expression Language (CEL)
- 1.25 Beta
- 1.29 GA (~12/23) KEP 2876

```yaml
...
openAPIV3Schema:
  type: object
  properties:
    spec:
      type: object
      x-kubernetes-validations:
        - rule: "self.minReplicas <= self.replicas"
          message: "replicas should be greater than or equal to minReplicas."
        - rule: "self.replicas <= self.maxReplicas"
          message: "replicas should be smaller than or equal to maxReplicas."
      properties:

        minReplicas:
          type: integer
        replicas:
          type: integer
        maxReplicas:
          type: integer
      required:
        - minReplicas
        - replicas
        - maxReplicas
```



Kubernetes CRD

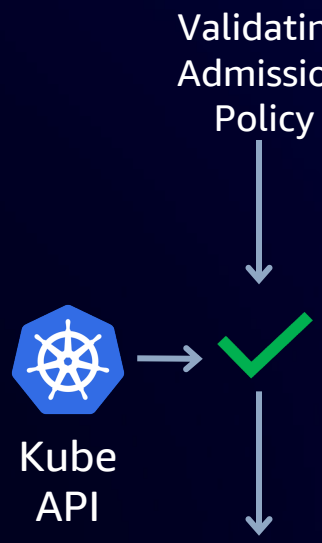
```yaml
...
openAPIV3Schema:
  type: object
  properties:
    spec:
      type: object
      x-kubernetes-validations:
        - rule: "self.minReplicas <= self.replicas"
          message: "replicas should be greater than or equal to minReplicas."
        - rule: "self.replicas <= self.maxReplicas"
          message: "replicas should be smaller than or equal to maxReplicas."
      properties:

        minReplicas:
          type: integer
        replicas:
          type: integer
        maxReplicas:
          type: integer
      required:
        - minReplicas
        - replicas
        - maxReplicas
```

Validating Admission

- Validating Admission Policy
- Common Expression Language (CEL)
- 1.26 Beta
- 1.29 GA (~12/23) KEP 3488

```yaml
apiVersion: admissionregistration.k8s.io/v1beta1
kind: ValidatingAdmissionPolicy
metadata:
  name: "image-matches-namespace-environment.policy.example.com"
spec:
  failurePolicy: Fail
  matchConstraints:
    resourceRules:
    - apiGroups:   ["apps"]
      apiVersions: ["v1"]
      operations:  ["CREATE", "UPDATE"]
      resources:   ["deployments"]
  variables:
  - name: environment
    expression: "'environment' in namespaceObject.metadata.labels ? namespaceObject.metadata.labels['environment'] : 'prod'"
  - name: exempt
    expression: "'exempt' in object.metadata.labels && object.metadata.labels['exempt'] == 'true'"
  - name: containers
    expression: "object.spec.template.spec.containers"
  - name: containersToCheck
    expression: "variables.containers.filter(c, c.image.contains('example.com/'))"
  validations:
  - expression: "variables.exempt || variables.containersToCheck.all(c, c.image.startsWith(variables.environment + '.'))"
    messageExpression: "'only ' + variables.environment + ' images are allowed in namespace ' + namespaceObject.metadata.name"
```

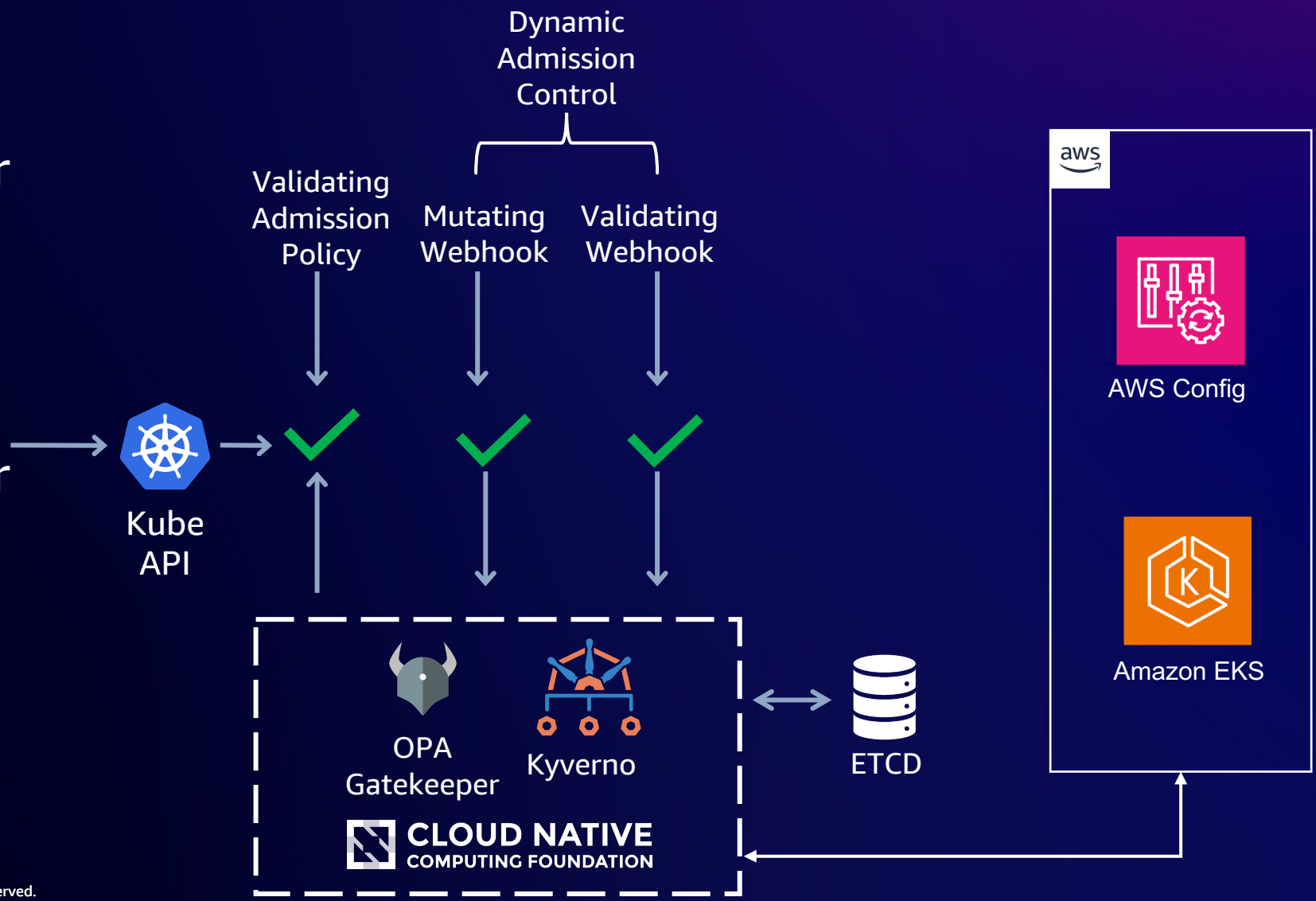


Validating Admission

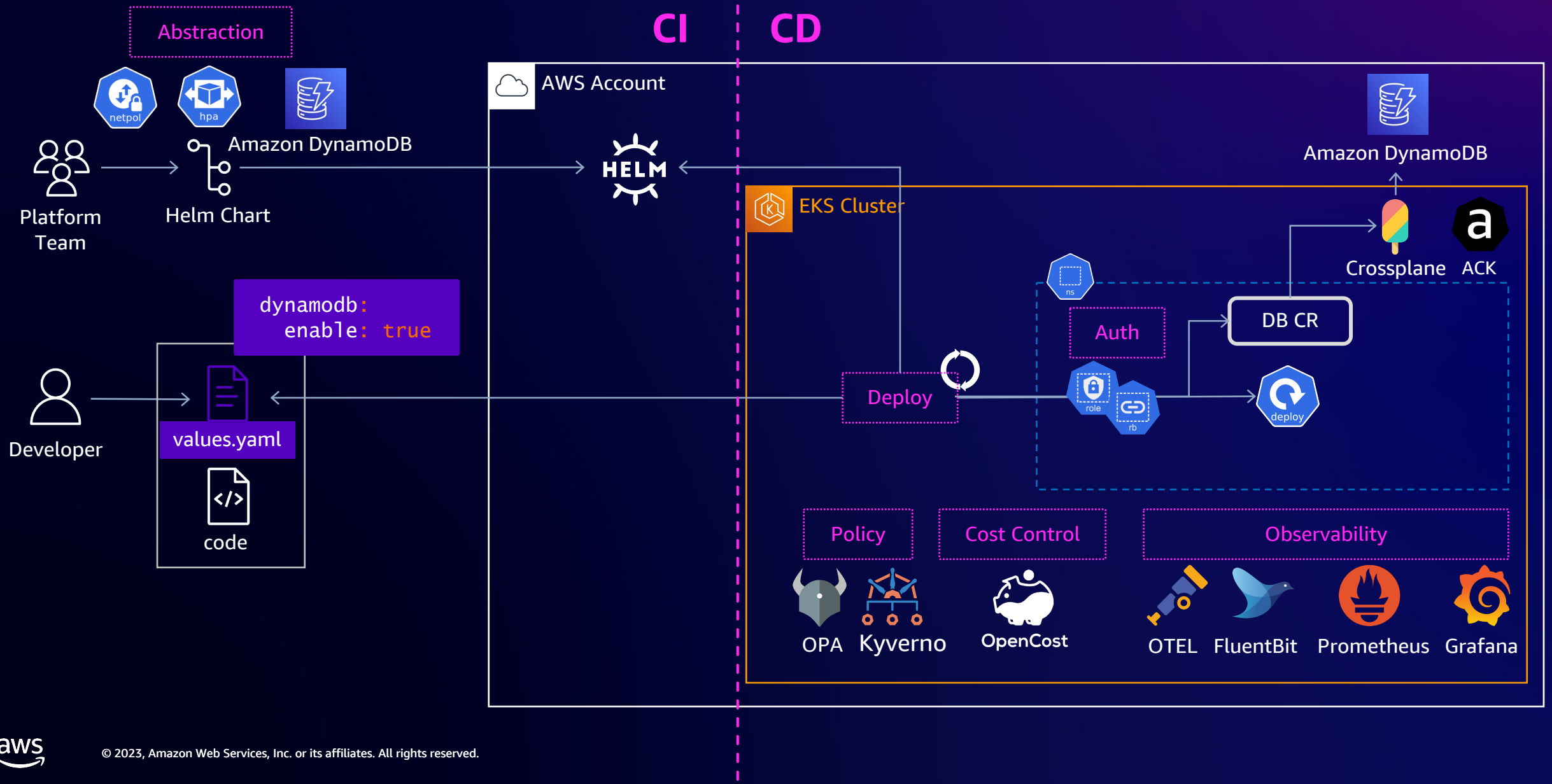
```yaml
apiVersion: admissionregistration.k8s.io/v1beta1
kind: ValidatingAdmissionPolicy
metadata:
  name: "image-matches-namespace-environment.policy.example.com"
spec:
  failurePolicy: Fail
  matchConstraints:
    resourceRules:
    - apiGroups:   ["apps"]
      apiVersions: ["v1"]
      operations:  ["CREATE", "UPDATE"]
      resources:   ["deployments"]
  variables:
  - name: environment
    expression: "'environment' in namespaceObject.metadata.labels ? namespaceObject.metadata.labels['environment'] : 'prod'"
  - name: exempt
    expression: "'exempt' in object.metadata.labels && object.metadata.labels['exempt'] == 'true'"
  - name: containers
    expression: "object.spec.template.spec.containers"
  - name: containersToCheck
    expression: "variables.containers.filter(c, c.image.contains('example.com/'))"
  validations:
  - expression: "variables.exempt || variables.containersToCheck.all(c, c.image.startsWith(variables.environment + '.'))"
    messageExpression: "'only ' + variables.environment + ' images are allowed in namespace ' + namespaceObject.metadata.name"
```

Policy Engines

- OPA/Gatekeeper
  - Webhooks, Controller
  - AWS Config custom rules
- Kyverno
  - Webhooks, Controller
  - Adapter for EKS



Kubernetes Infrastructure Controller



Thank You

Carlos Santana
carrlos@amazon.com
csantanapr
@csantanapr

GitHub Repo
Demo and Resources