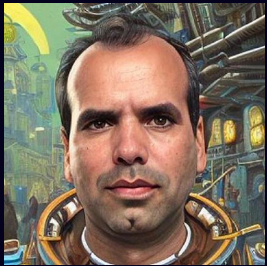# Cloud Control Planes on Kubernetes and Policy Validation

Carlos Santana

Sr. Specialist Solutions Architect, Kubernetes

in csantanapr

X 🐦 @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.
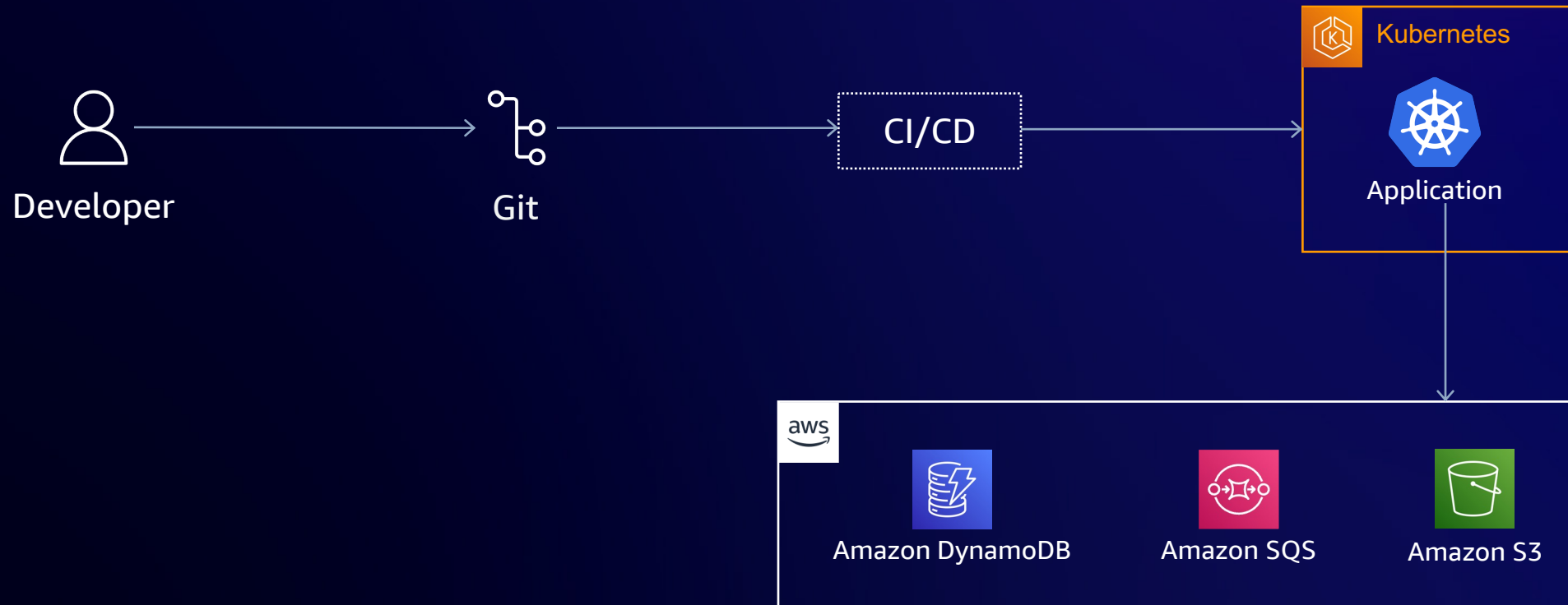
CNCF State of Cloud Native Development
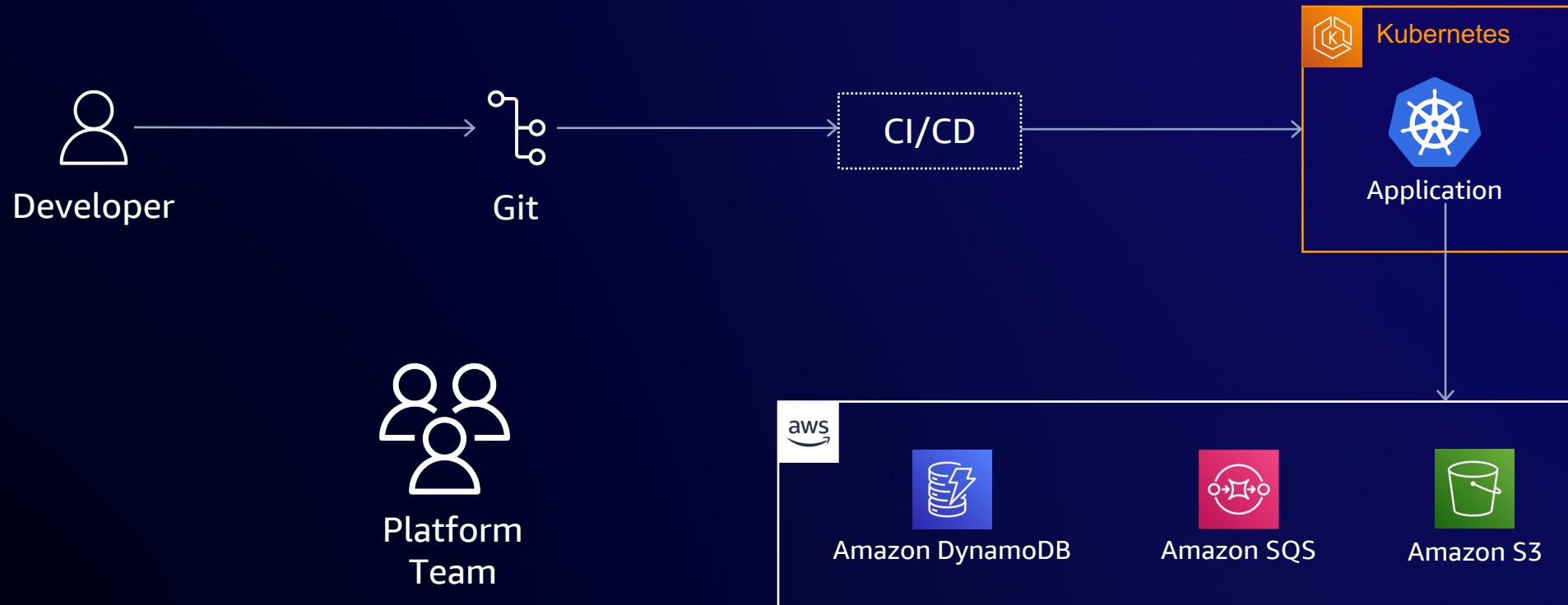
Deploy Kubernetes Applications

Developer → Git → CI/CD → Kubernetes Application

Deploy Kubernetes Applications

Deploy Kubernetes Applications
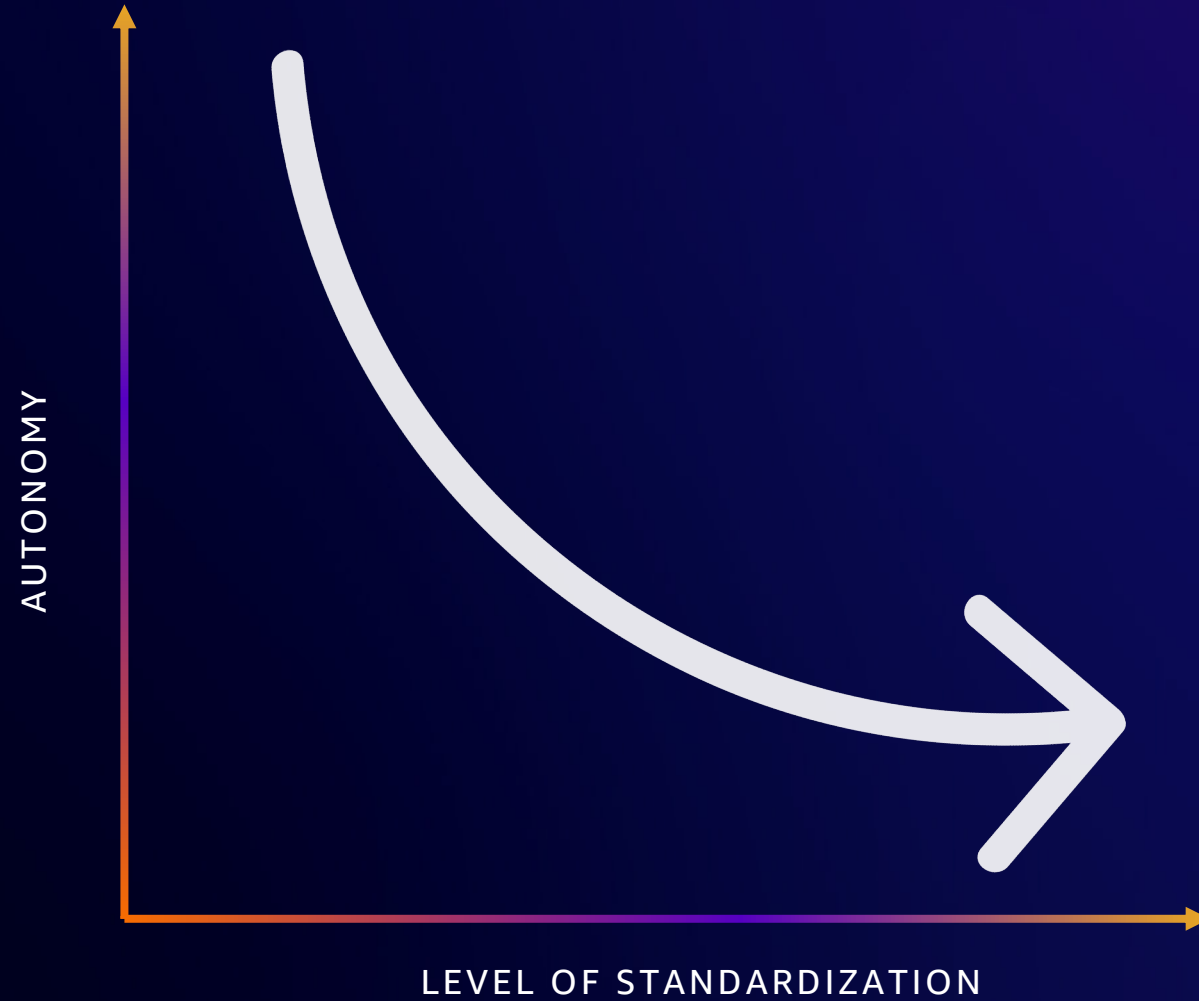
Separation of Concerns

Platform Team

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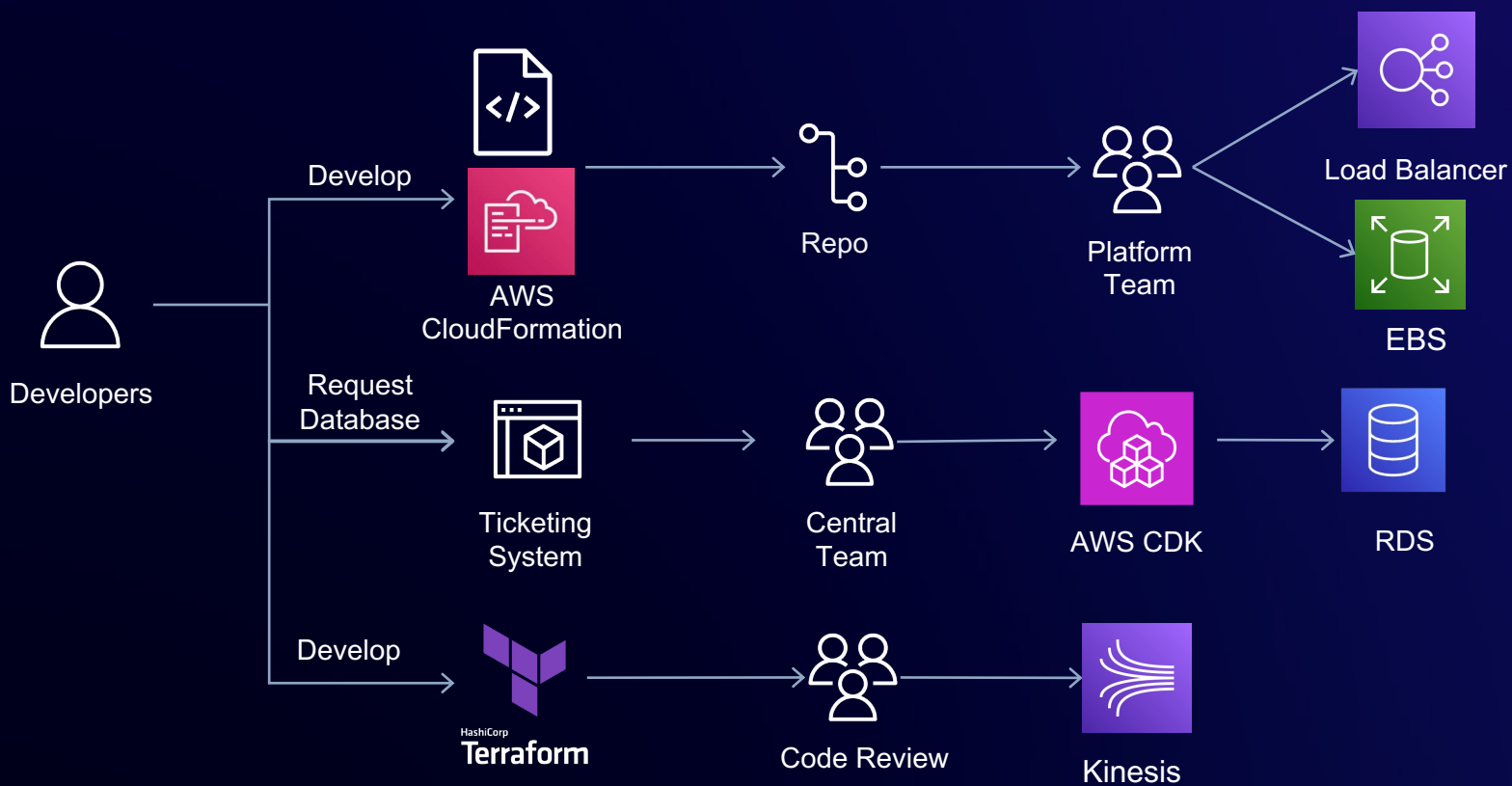
Application Teams

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

Autonomy vs Standardization

AUTONOMY

LEVEL OF STANDARDIZATION

Disjointed Workflows Adds Developer Overhead

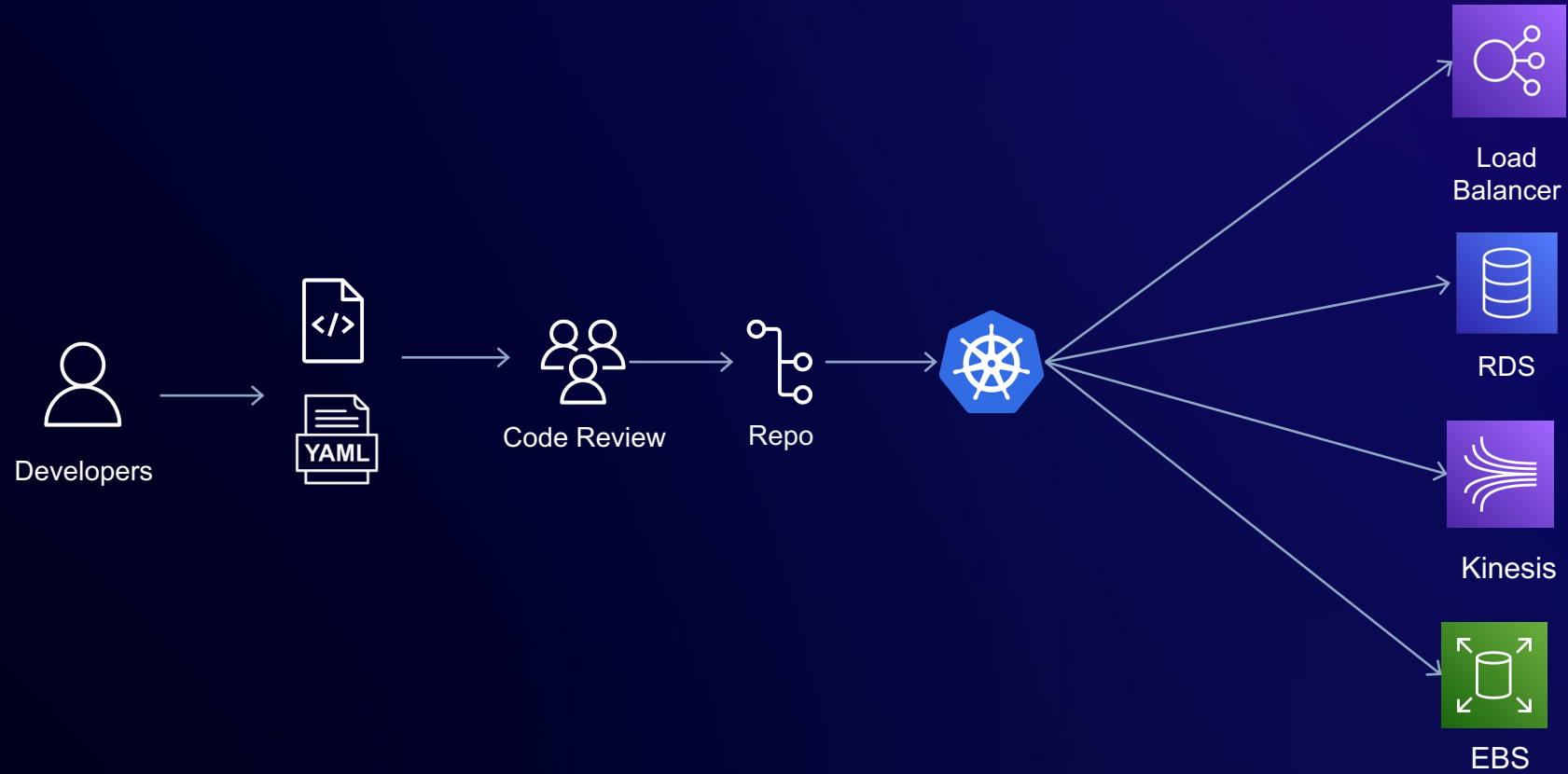How do you **get the state of the entire system?**

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

Developers → YAML → Code Review → Repo → Kubernetes → Load Balancer, RDS, Kinesis, EBS

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
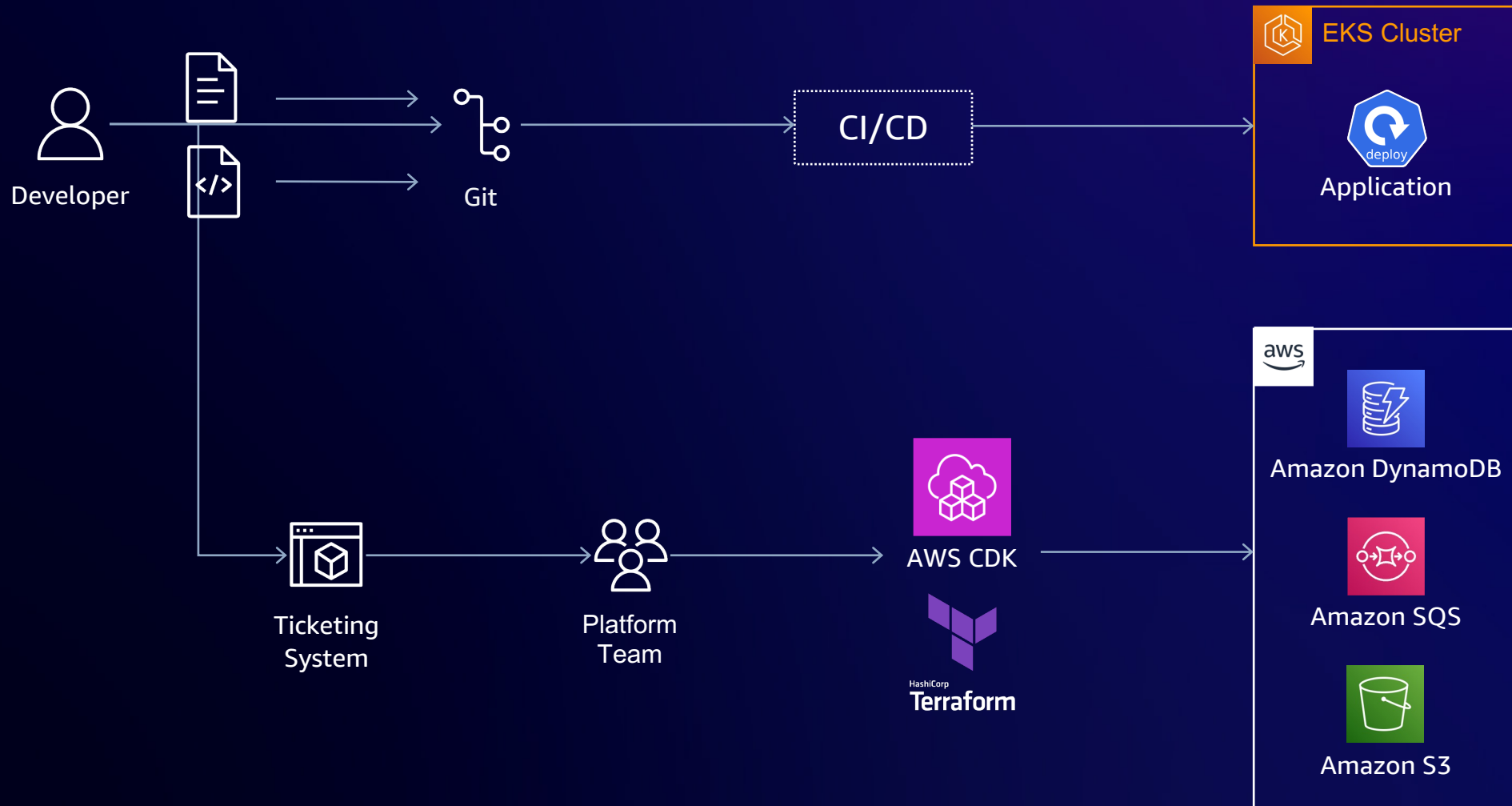
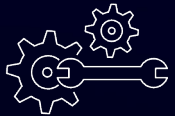# Kubernetes Infrastructure Controller
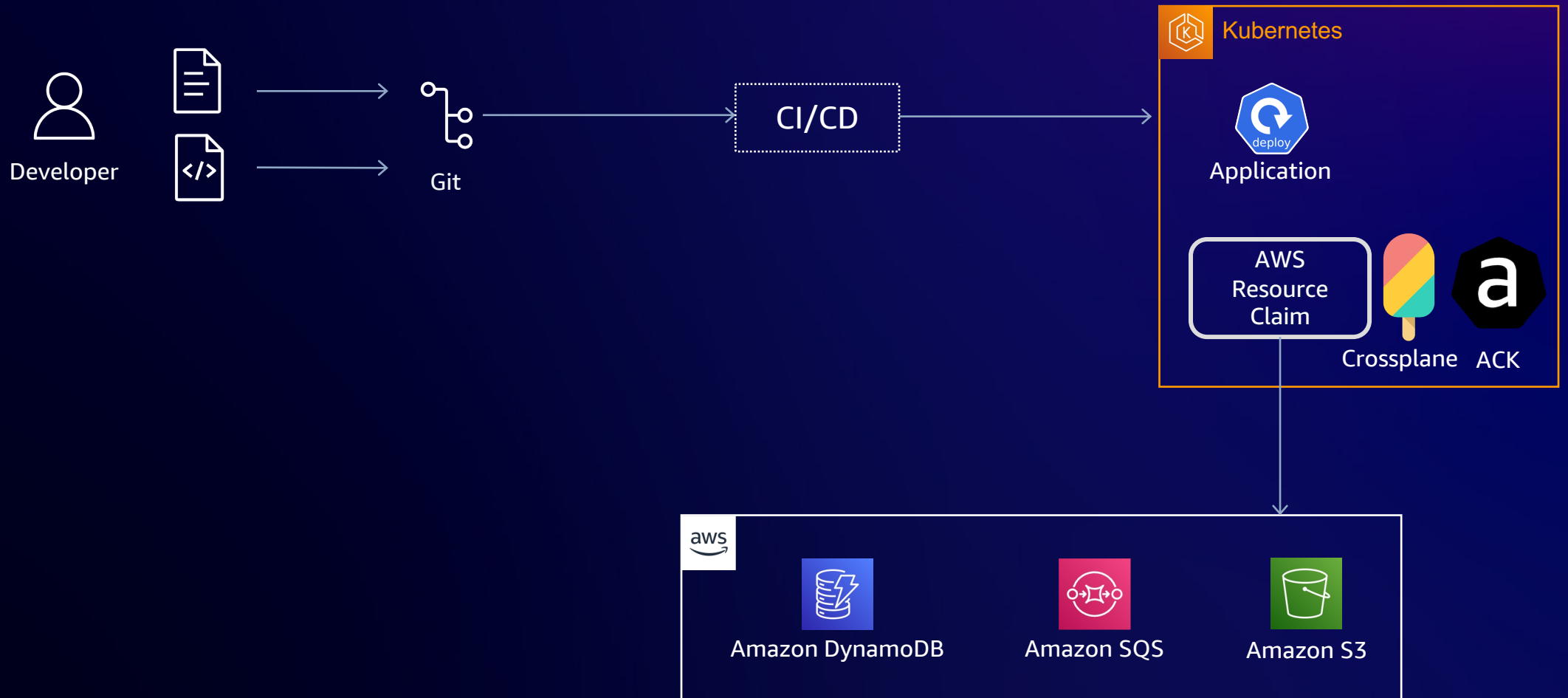
Infrastructure controllers

Manage cloud services using Kubernetes API

A single API for Kubernetes and cloud services

ACK

Create your own platform API
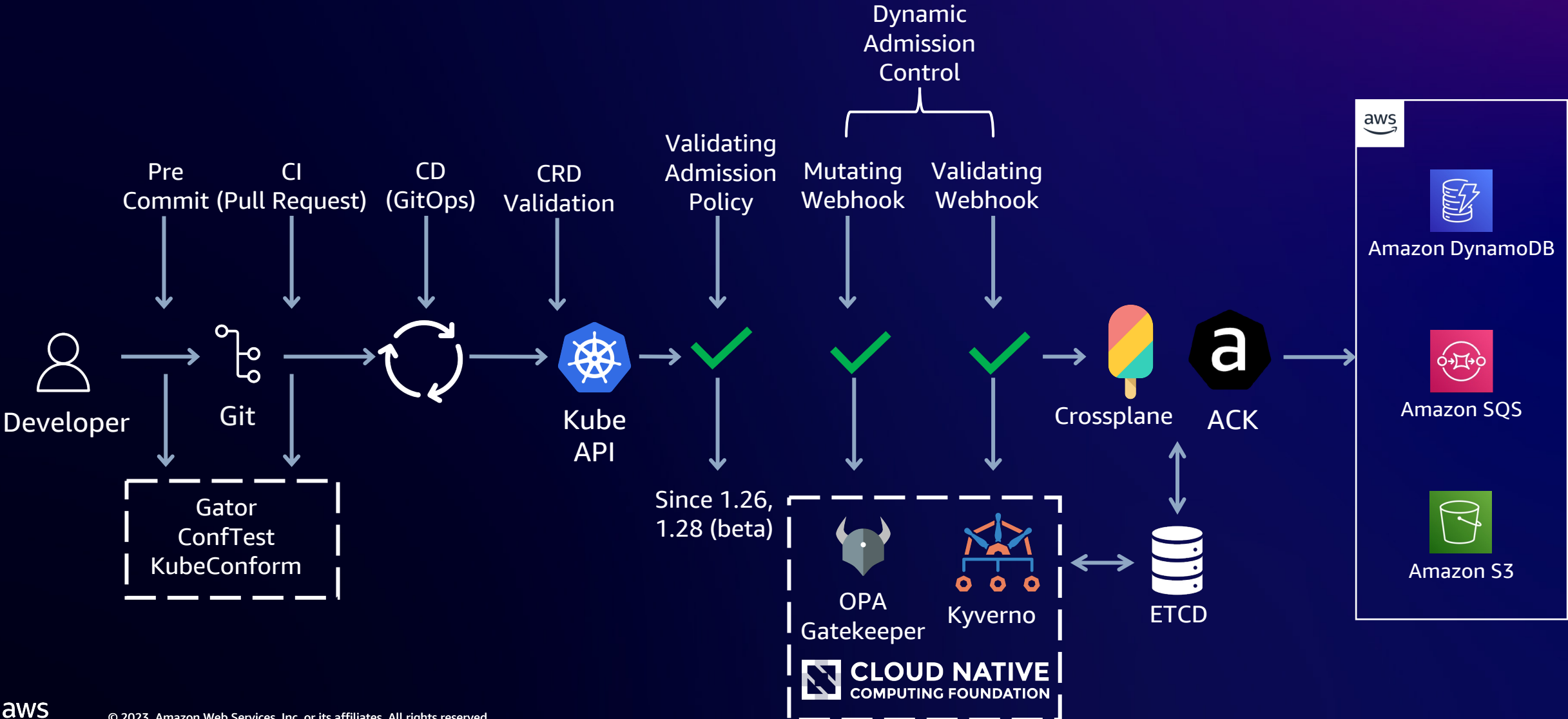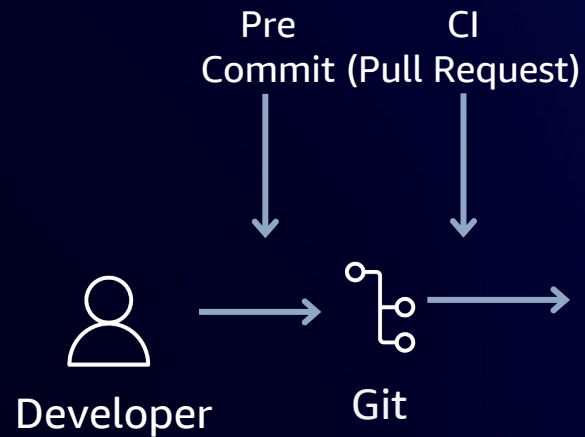
Declarative infrastructure configuration
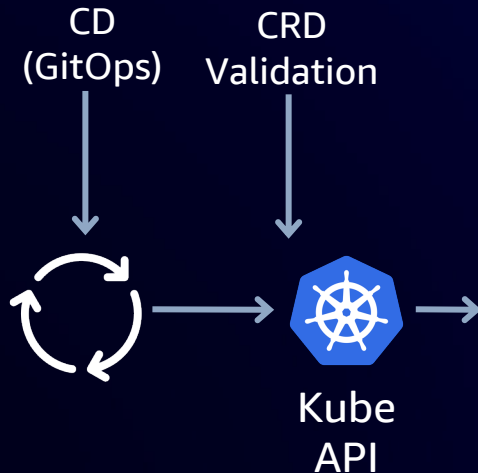
Crossplane

Where to check?

Local Testing

- Rego
  - opa test
  - gator verify
  - conftest
- Kyverno
- Kubeconform

Pre Commit

CI (Pull Request)

Developer

Git

Kubernetes CRD

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

```
1   ...
2   openAPIV3Schema:
3     type: object
4     properties:
5       spec:
6         type: object
7         x-kubernetes-validations:
8           - rule: "self.minReplicas <= self.replicas"
9             message: "replicas should be greater than or equal to minReplicas."
10          - rule: "self.replicas <= self.maxReplicas"
11            message: "replicas should be smaller than or equal to maxReplicas."
12        properties:
13
14          minReplicas:
15            type: integer
16          replicas:
17            type: integer
18          maxReplicas:
19            type: integer
20        required:
21          - minReplicas
22          - replicas
23          - maxReplicas
```

CD
(GitOps)

CRD
Validation

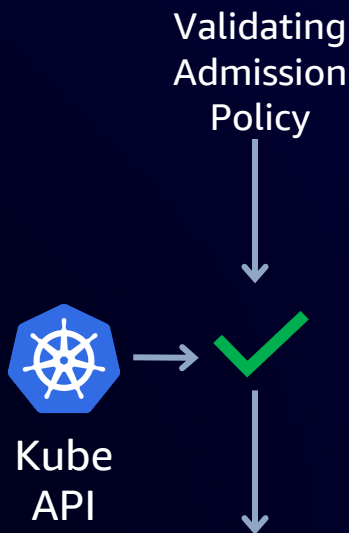Kube
API

Kubernetes CRD

```
 1  ...
 2  openAPIV3Schema:
 3    type: object
 4    properties:
 5      spec:
 6        type: object
 7        x-kubernetes-validations:
 8          - rule: "self.minReplicas <= self.replicas"
 9            message: "replicas should be greater than or equal to minReplicas."
10          - rule: "self.replicas <= self.maxReplicas"
11            message: "replicas should be smaller than or equal to maxReplicas."
12        properties:
13
14          minReplicas:
15            type: integer
16          replicas:
17            type: integer
18          maxReplicas:
19            type: integer
20        required:
21          - minReplicas
22          - replicas
23          - maxReplicas
```

Validating Admission

- Validating Admission Policy

- Common Expression Language (CEL)

- 1.26 Beta

- 1.29 GA (~12/23)   KEP 3488

Validating
Admission
Policy

Kube
API

```
5   apiVersion: admissionregistration.k8s.io/v1beta1
6   kind: ValidatingAdmissionPolicy
7   metadata:
8     name: "image-matches-namespace-environment.policy.example.com"
9   spec:
10    failurePolicy: Fail
11    matchConstraints:
12      resourceRules:
13      - apiGroups:    ["apps"]
14        apiVersions:  ["v1"]
15        operations:   ["CREATE", "UPDATE"]
16        resources:    ["deployments"]
17    variables:
18    - name: environment
19      expression: "'environment' in namespaceObject.metadata.labels ? namespaceObject.metadata.labels['environment'] : 'prod'"
20    - name: exempt
21      expression: "'exempt' in object.metadata.labels && object.metadata.labels['exempt'] == 'true'"
22    - name: containers
23      expression: "object.spec.template.spec.containers"
24    - name: containersToCheck
25      expression: "variables.containers.filter(c, c.image.contains('example.com/'))"
26    validations:
27    - expression: "variables.exempt || variables.containersToCheck.all(c, c.image.startsWith(variables.environment + '.'))"
28      messageExpression: "'only ' + variables.environment + ' images are allowed in namespace ' + namespaceObject.metadata.name"
```

Validating Admission

```yaml
 5    apiVersion: admissionregistration.k8s.io/v1beta1
 6    kind: ValidatingAdmissionPolicy
 7    metadata:
 8      name: "image-matches-namespace-environment.policy.example.com"
 9    spec:
10      failurePolicy: Fail
11      matchConstraints:
12        resourceRules:
13        - apiGroups:   ["apps"]
14          apiVersions: ["v1"]
15          operations:  ["CREATE", "UPDATE"]
16          resources:   ["deployments"]
17      variables:
18      - name: environment
19        expression: "'environment' in namespaceObject.metadata.labels ? namespaceObject.metadata.labels['environment'] : 'prod'"
20      - name: exempt
21        expression: "'exempt' in object.metadata.labels && object.metadata.labels['exempt'] == 'true'"
22      - name: containers
23        expression: "object.spec.template.spec.containers"
24      - name: containersToCheck
25        expression: "variables.containers.filter(c, c.image.contains('example.com/'))"
26      validations:
27      - expression: "variables.exempt || variables.containersToCheck.all(c, c.image.startsWith(variables.environment + '.'))"
28        messageExpression: "'only ' + variables.environment + ' images are allowed in namespace ' + namespaceObject.metadata.name"
```
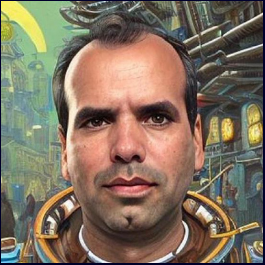
Policy Engines

- OPA/Gatekeeper
- Kyverno
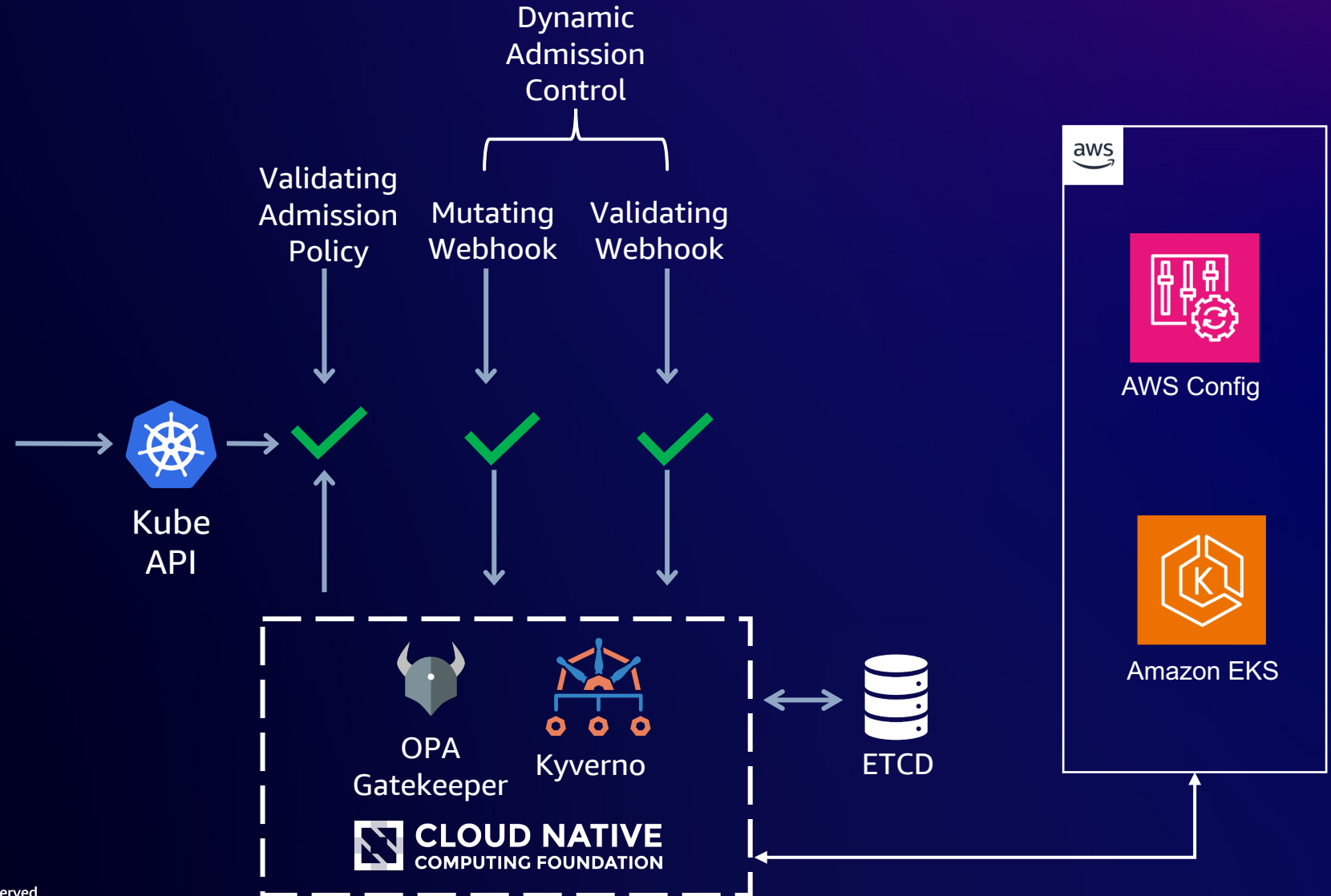
Kubernetes Infrastructure Controller
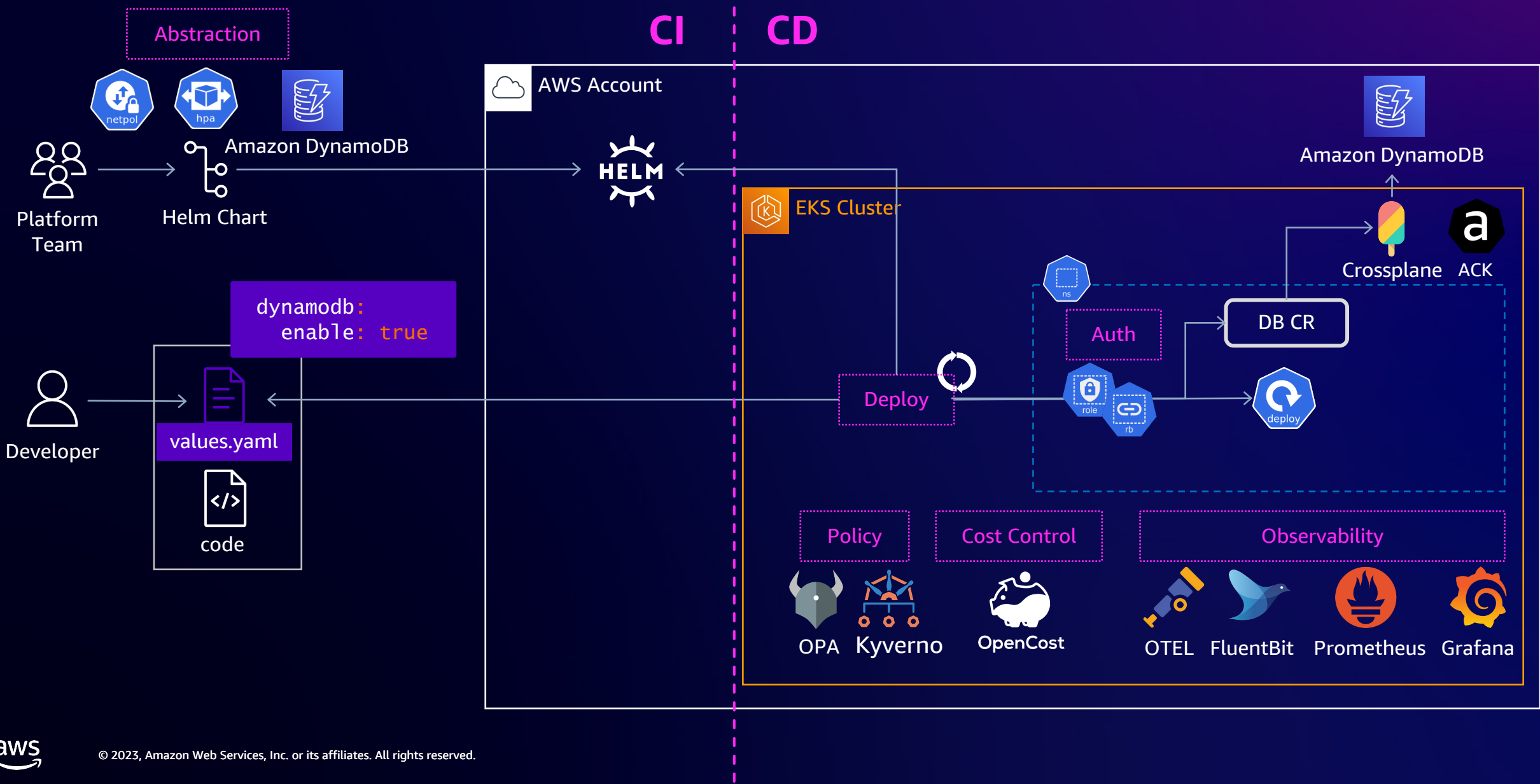
Thank You

Carlos Santana

carrlos@amazon.com

in csantanapr

X 🐦 @csantanapr

**GitHub Repo
Demo and Resources**