

Open Source AceCon

2021 智能云边开源峰会

AI x Cloud Native x Edge Computing

人工智能 × 云原生 × 边缘计算

KubeVela: 云原生时代的应用交付 与管理控制平面

孙健波

阿里云 技术专家

KubeVela 是什么？

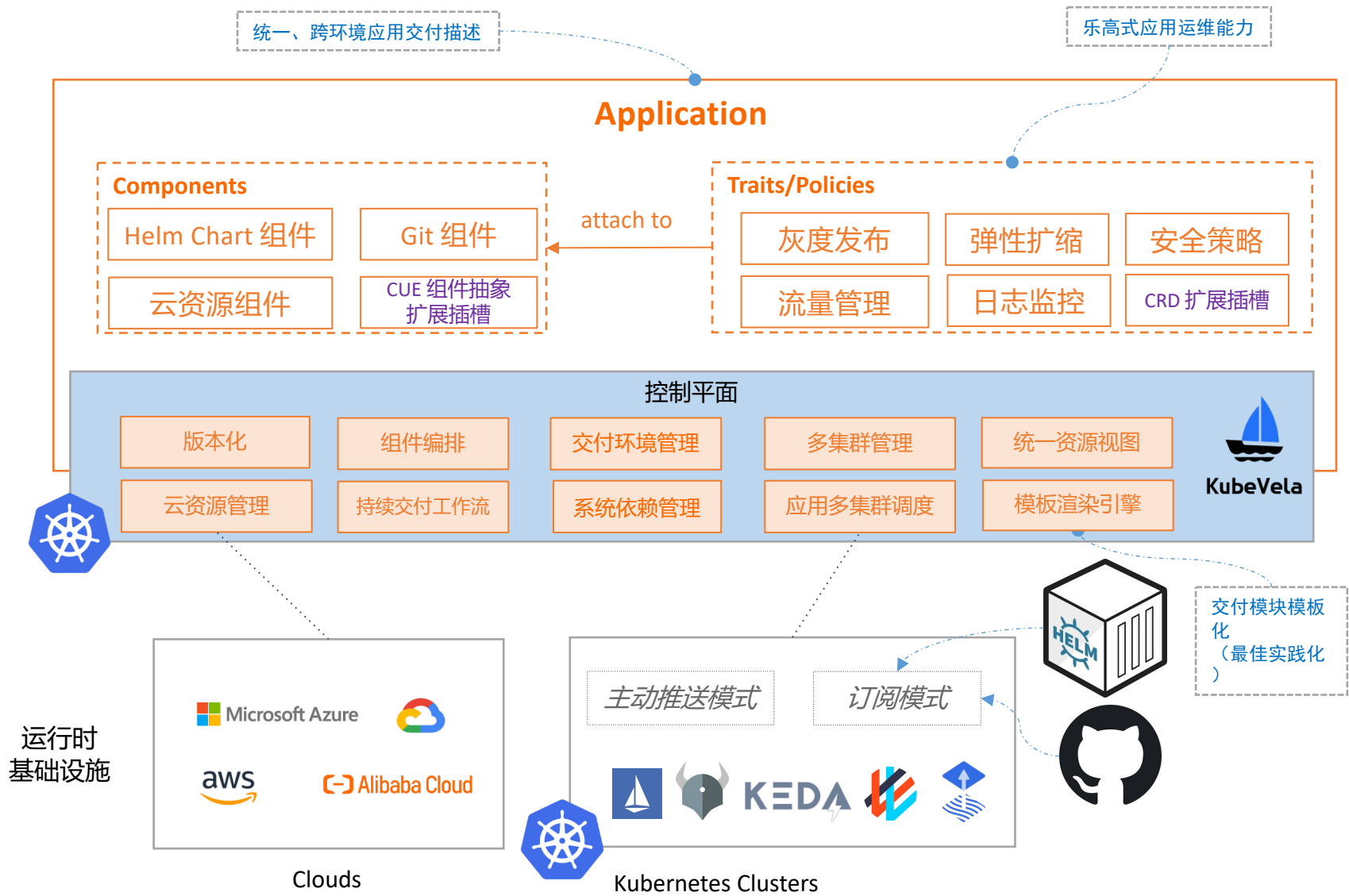
- 一个面向多云混合环境的易用可扩展的部署引擎
 - 基于 Kubernetes 的控制平面
 - 运行时平台无关（可以交付到云平台、边缘、K8s等多样化环境）
 - 大规模混合基础设施的持续交付平台
- 阿里巴巴应用交付和应用管理平台背后的核心组件
 - 构建 GitOps 和 IaC (Infrastructure-as-code) 的解决方案
- OAM（Open Application Model）的官方实现
- CNCF 官方项目
 - <https://github.com/oam-dev/kubevela>

OAM/KubeVela 里程碑

- 2020 年 KubeVela 作为 OAM 的官方实现发布
- 2020 年 OAM 被信通院立项作为行业标准
- 2021.3 KubeVela 发布 v1.0.0 GA 版本
- 2021.6 OAM/KubeVela 正式成为 CNCF 项目
- 2021.7 KubeVela 与 RedHat 联合发布多集群应用管理能力
- 2021.8 KubeVela 多集群混合云控制平面版本 v1.1.0 正式发布
- 截止至 2021.8 累计：
 - 35+ 社区企业用户，3+ 生态合作伙伴，96+ 位贡献者，Maintainer 10+ 人，涵盖云服务PaaS、AI、游戏、金融支付、电商、物联网、音视频等众多场景和领域
 - Star 2.5k+, commit 2k+, Issue/PR 2K+



KubeVela – 一个面向多云混合环境的易用可扩展的部署引擎



KubeVela Application（应用部署计划）

易用可扩展

基于应用模型 IaC 抽象，降低使用门槛
部署资源的完整集合与锚点（single source of truth）

组件能力“胶水”

高度可扩展：模块化接入、拼图式衔接、管道式编排
能力“可编程”：高效响应需求变化，没有系统变更负担

无差别应用交付的控制平面

多集群，多环境应用依赖管理，统一的运维能力建设
应用多环境部署和运维，差异化环境一致体验

应用的执行策略

比如安全策略、健康检查策略、防火墙规则等，任何一个部署前需要遵守的规则都可以在这个阶段声明和执行！

```
1 kind: Application
2 spec:
3   components:
4     - name: express-server
5       type: webservice
6       properties:
7         image: demo/hello-world
8         port: 8000
9   traits:
10    - type: ingress
11      properties:
12        domain: testsvc.example.com
13        http:
14          "/": 8000
15  policies:
16    - type: security
17      properties:
18        audit: enabled
19        secretBackend: vault
20    - type: deployment-insights
21      properties:
22        provider: arms | promethues
23        leadTime: enabled
24        frequency: enabled
25        mttr: enabled
26  workflow:
27    - type: blue-green-rollout
28      stage: post-render
29      properties:
30        partition: "50%"
31    - type: traffic-shift
32      properties:
33        partition: "50%"
34    - type: rollout-promotion
35      propertie:
36        manualApproval: true
37        rollbackIfNotApproved: true
```

待部署组件

比如 Helm chart, Kustomize, 云资源 Cloud Formation 模板, Terraform 模块等... 几乎可以包含一切可交付制品！

运维能力

比如 路由规则、自动扩缩容规则等... 像乐高一样可以附着作用于组件上！

部署的执行过程

比如蓝绿部署、带流量的渐进式部署、手动审批等任意的管道式持续交付策略！

X-Definition (IaC 模块)

- 将基础设施能力变成“乐高”一样的构建模块呈现给最终用户
 - i.e. ComponentDefinition, TraitDefinition, PolicyDefinition, WorkflowDefinition
- 基于 [CUElang](#) 的超级“胶水层”
 - 配置可编程
 - 高效反馈用户需求的变化
 - 没有限制、高度可扩展的PaaS
 - 可以在部署计划中粘合任意组件
 - e.g. Helm, Kustomize, K8s API/CRD, Terraform, AWS CF, ROS ...

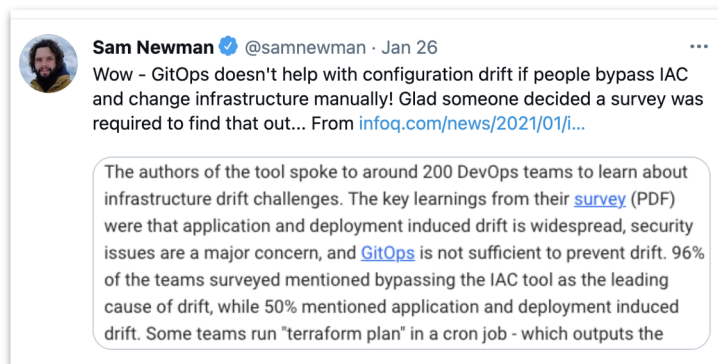
```
apiVersion: core.oam.dev/v1alpha2
kind: TraitDefinition
metadata:
  annotations:
    definition.oam.dev/description: "add sidecar to the app"
  name: sidecar
spec:
  appliesToWorkloads:
    - webservice
    - worker
  extension:
    template: |-
      patch: {
        // +patchKey=name
        spec: template: spec: containers: [parameter]
      }
    parameter: {
      name: string
      image: string
      command?: [...string]
    }
```

CUE

[Full Sample](#)

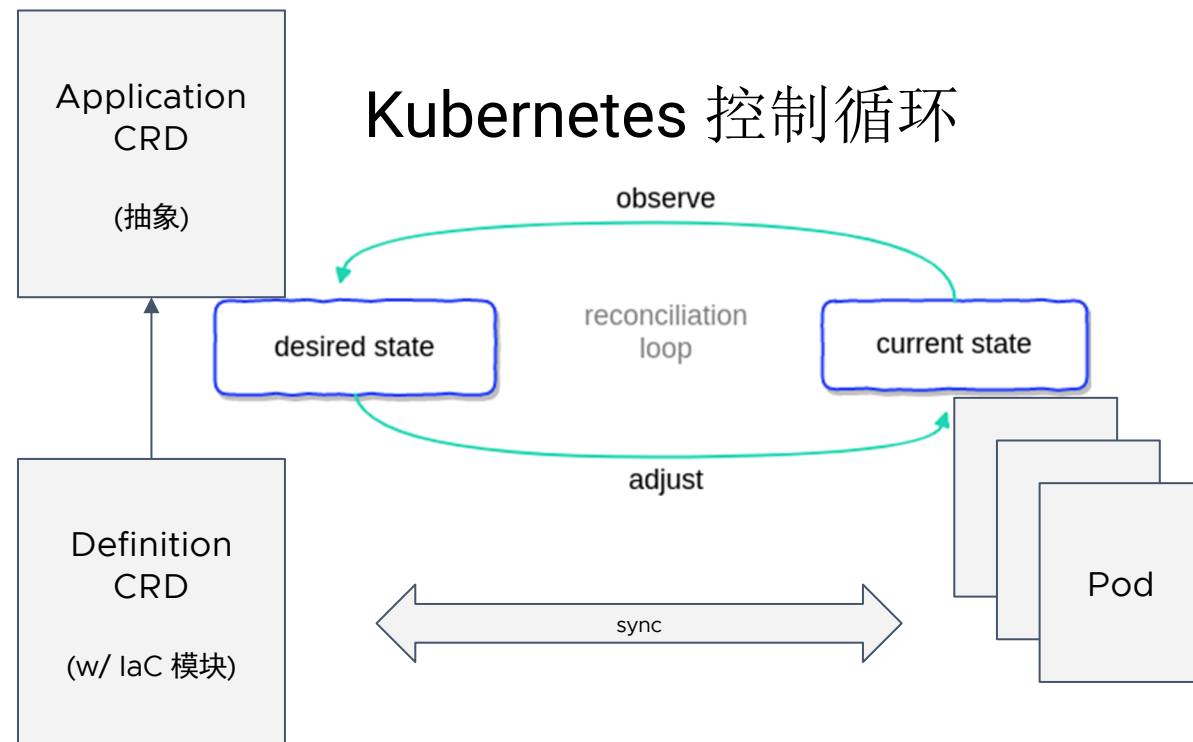
IaC 与 Kubernetes 控制循环的结合

- IaC 是基础设施能力层最好的“胶水”，也是 KubeVela 的魔力所在
- 传统的 IaC 模式会导致“配置漂移”
 - 即：运行的实例与期望配置出现不一致
 - 越是大规模，问题越是严重

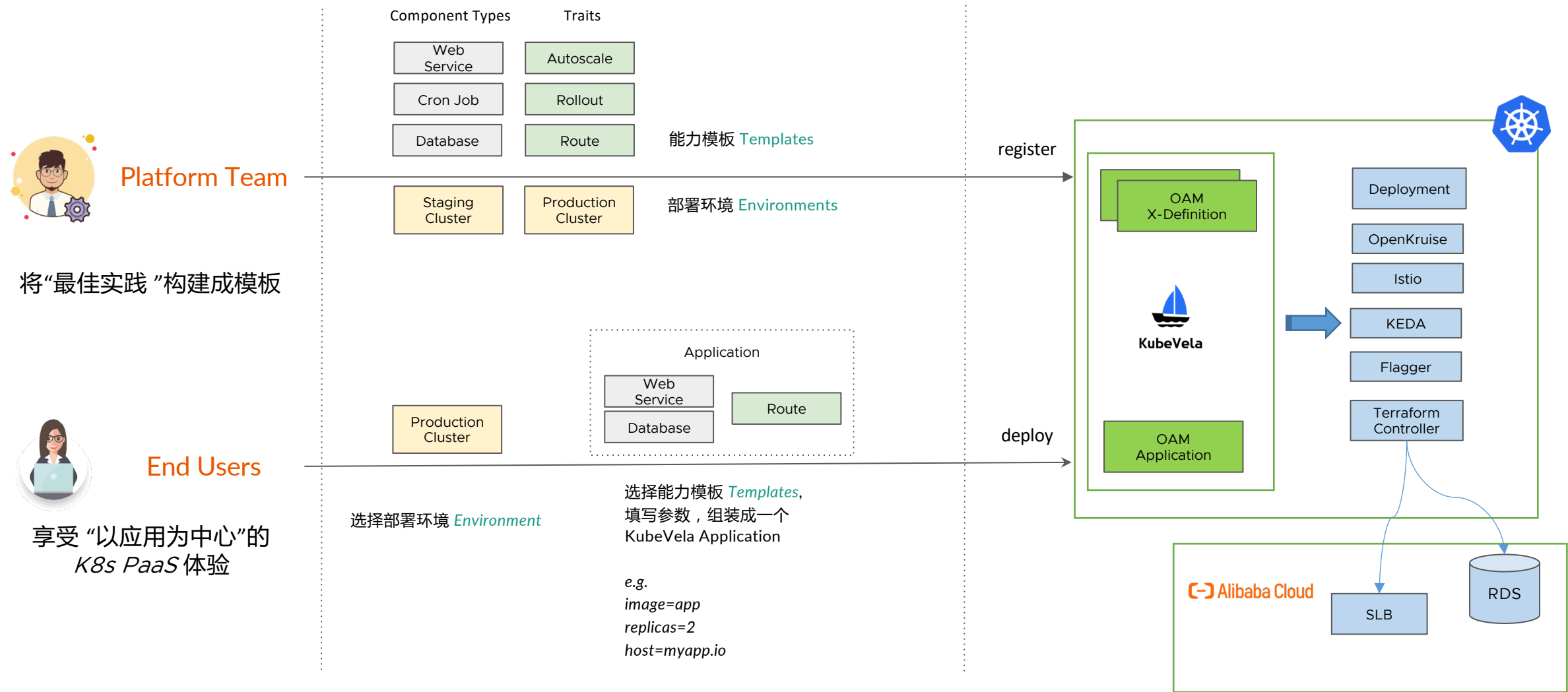


<https://www.infoq.com/news/2021/01/infrastructure-drift-driftctl/>

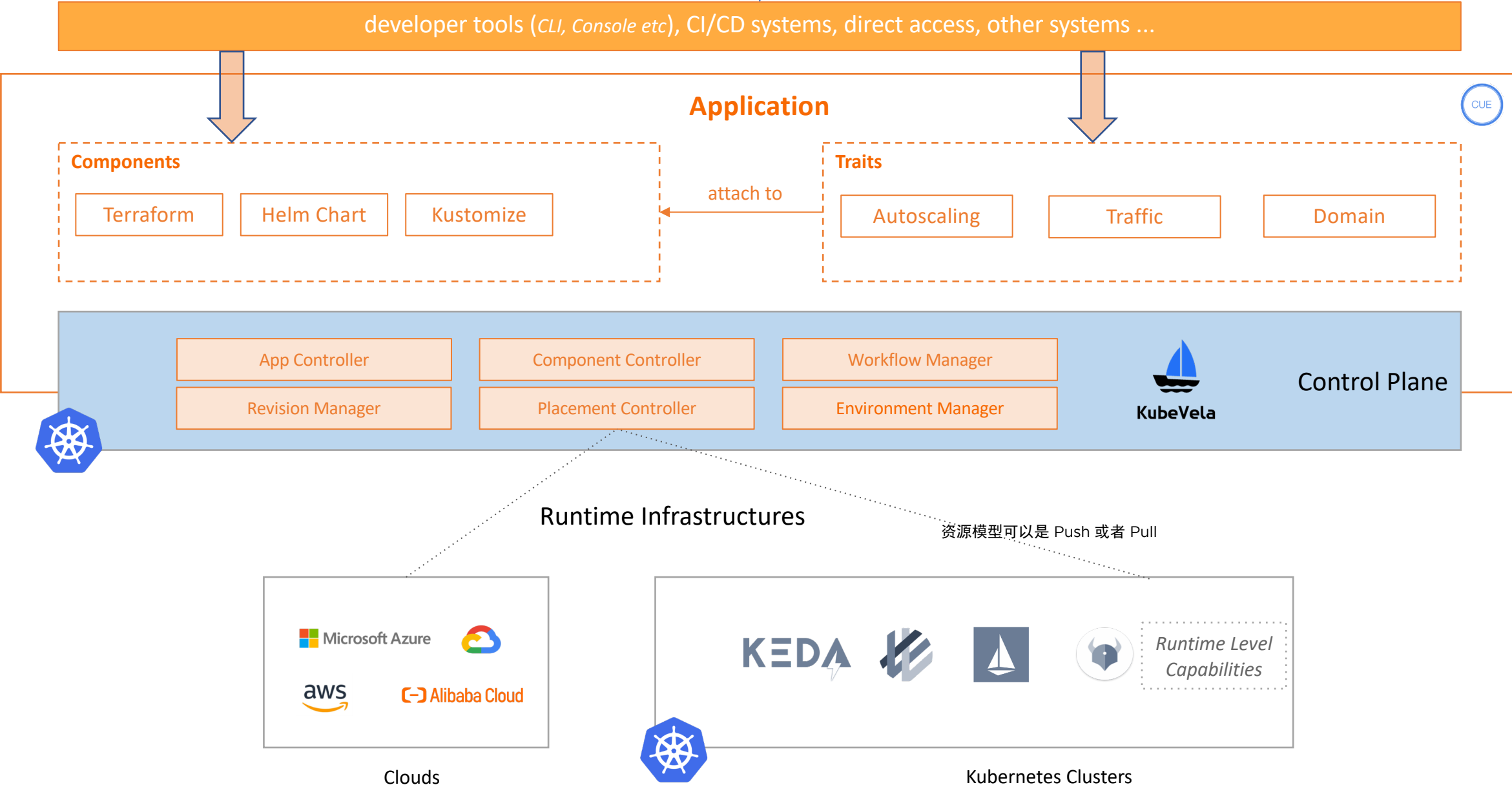
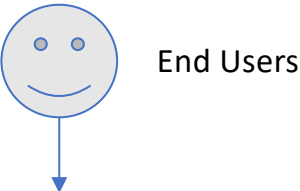
○ ‘Reilly 出版的《微服务设计》《从单体到微服务》作者



用户场景一：易用可扩展的通用 PaaS 引擎



用户场景二：混合环境多集群应用控制平面



用户场景三：DevOps 一致性体验



差异化配置

交付 workflow

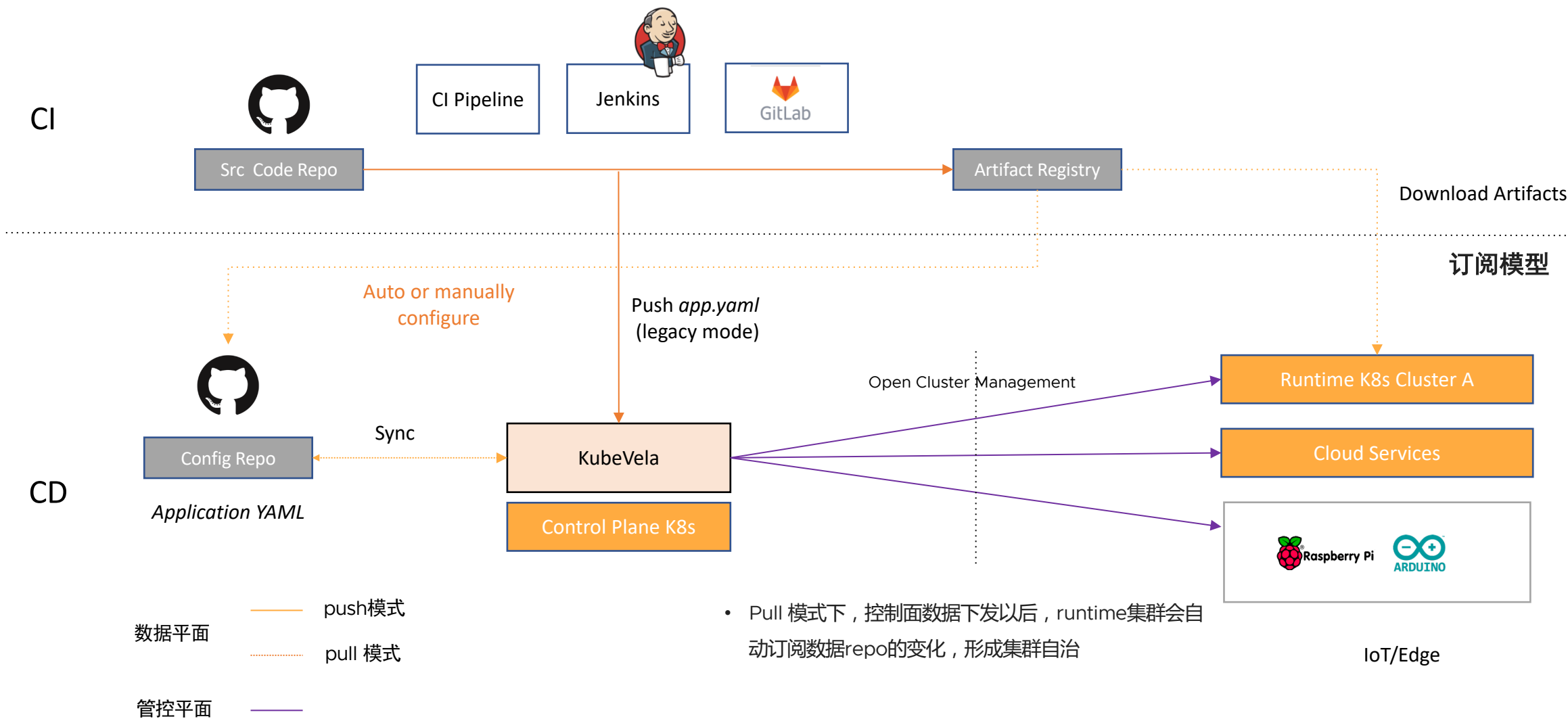
集群初始化

指定环境部署

```
apiVersion: core.oam.dev/v1beta1
kind: Application
metadata:
  name: workflow-demo
  namespace: default
spec:
  components:
    - name: nginx-server
      type: webservice
      properties:
        image: nginx:1.21
        port: 80
  policies:
    - name: patch
      type: env-binding
      properties:
        envs:
          - name: test
            patch:
              traits:
                - type: elk
                  properties:
                    index: my-log-repo
            placement:
              clusterSelector:
                labels:
                  purpose: test
          - name: prod
            patch:
              traits:
                - type: sls
                  properties:
                    repo: my-log-repo
            placement:
              clusterSelector:
                labels:
                  purpose: prod

workflow:
  steps:
    - name: cluster-initilize
      type: initializer
      properties:
        name: ack-cluster-initilizer
        parameters:
          region: cn-hongkong
          version: v1.20
          labels:
            purpose: test
    - name: deploy-server
      type: deploy2cluster
      properties:
        env: prod
        policy: patch
        component: nginx-server
```

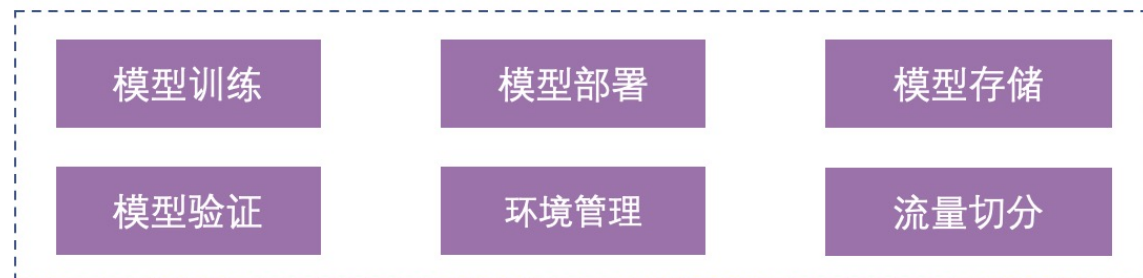
用户场景四：基于 GitOps 模式的多集群多环境应用交付



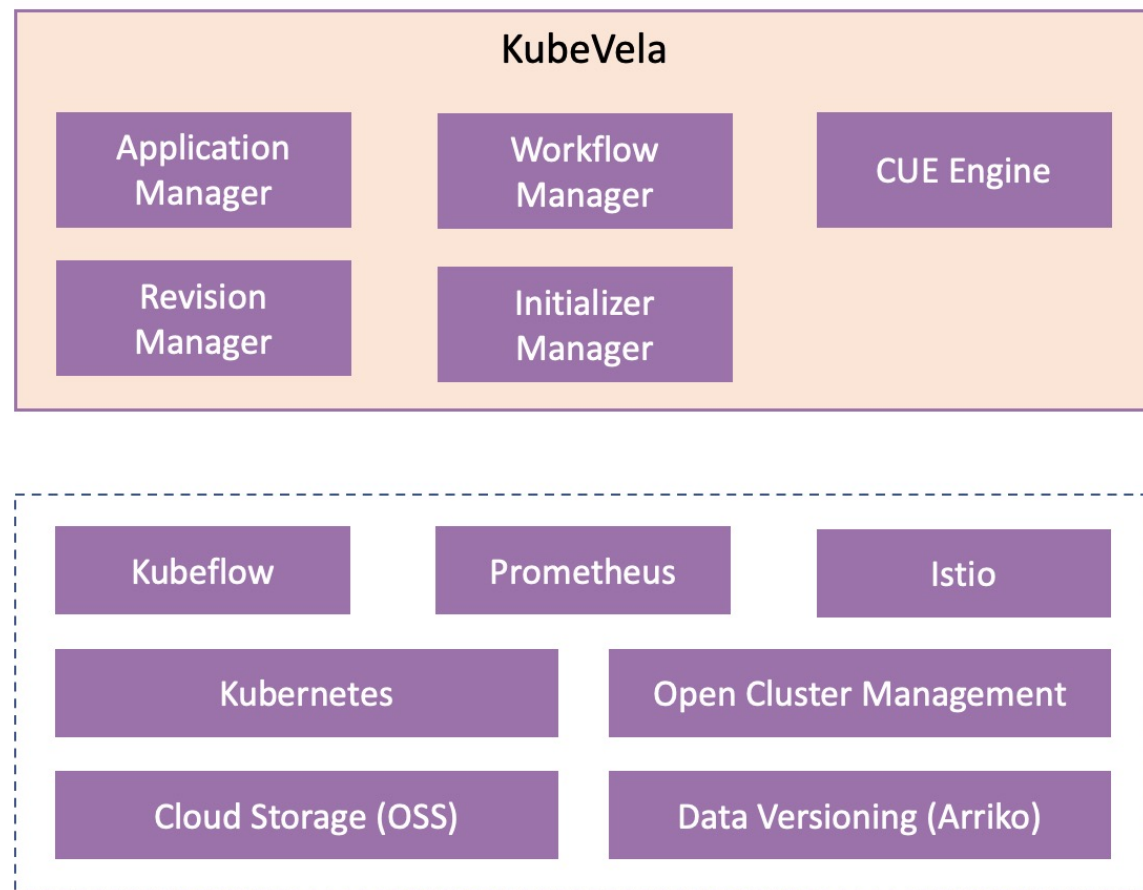
垂直场景举例：

- 机器学习应用部署平台
 - ✓ 基础设施一键创建
 - ✓ 模型训练和服务一站式编排
 - ✓ 云服务和云资源自动化衔接

用户交互层



基础设施层



Open Source AceCon

2021 智能云边开源峰会

AI x Cloud Native x Edge Computing

人工智能 × 云原生 × 边缘计算

Thank You