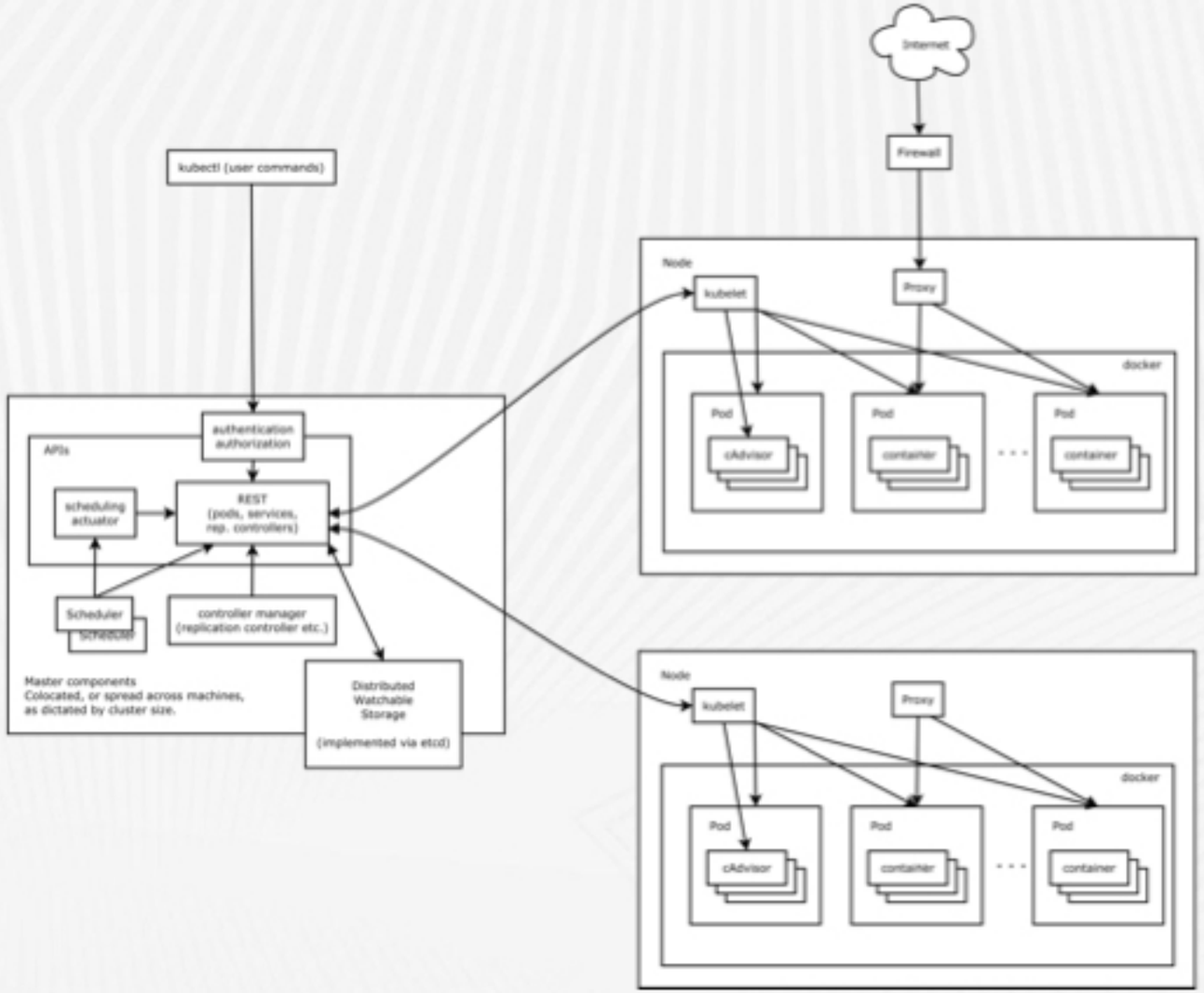
基于 Virtual Kubelet 的多租户Serverless K8s 落地实践

CSDN

目录

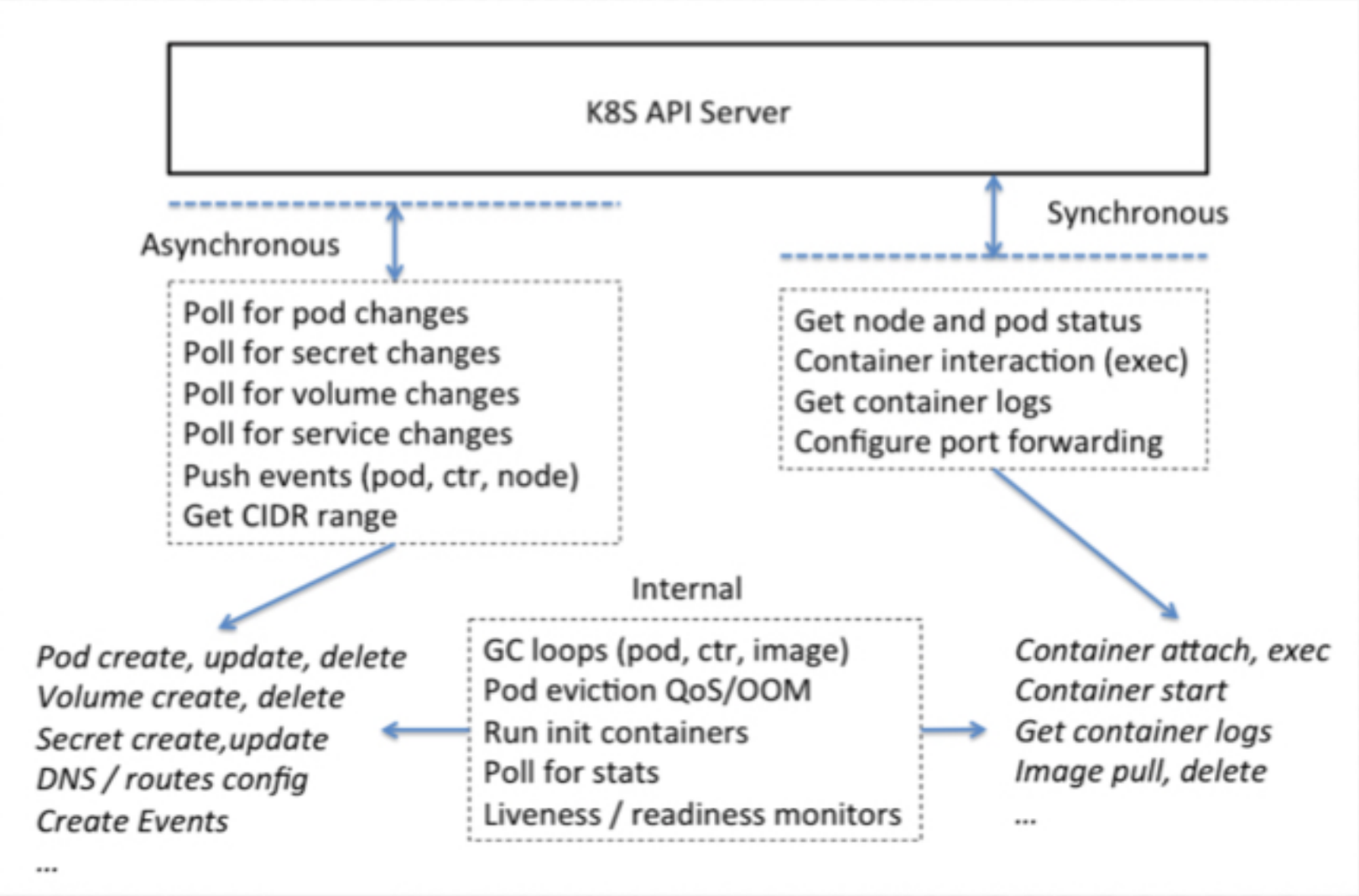
01 背景介绍

原生 Kubernetes 集群运维痛点



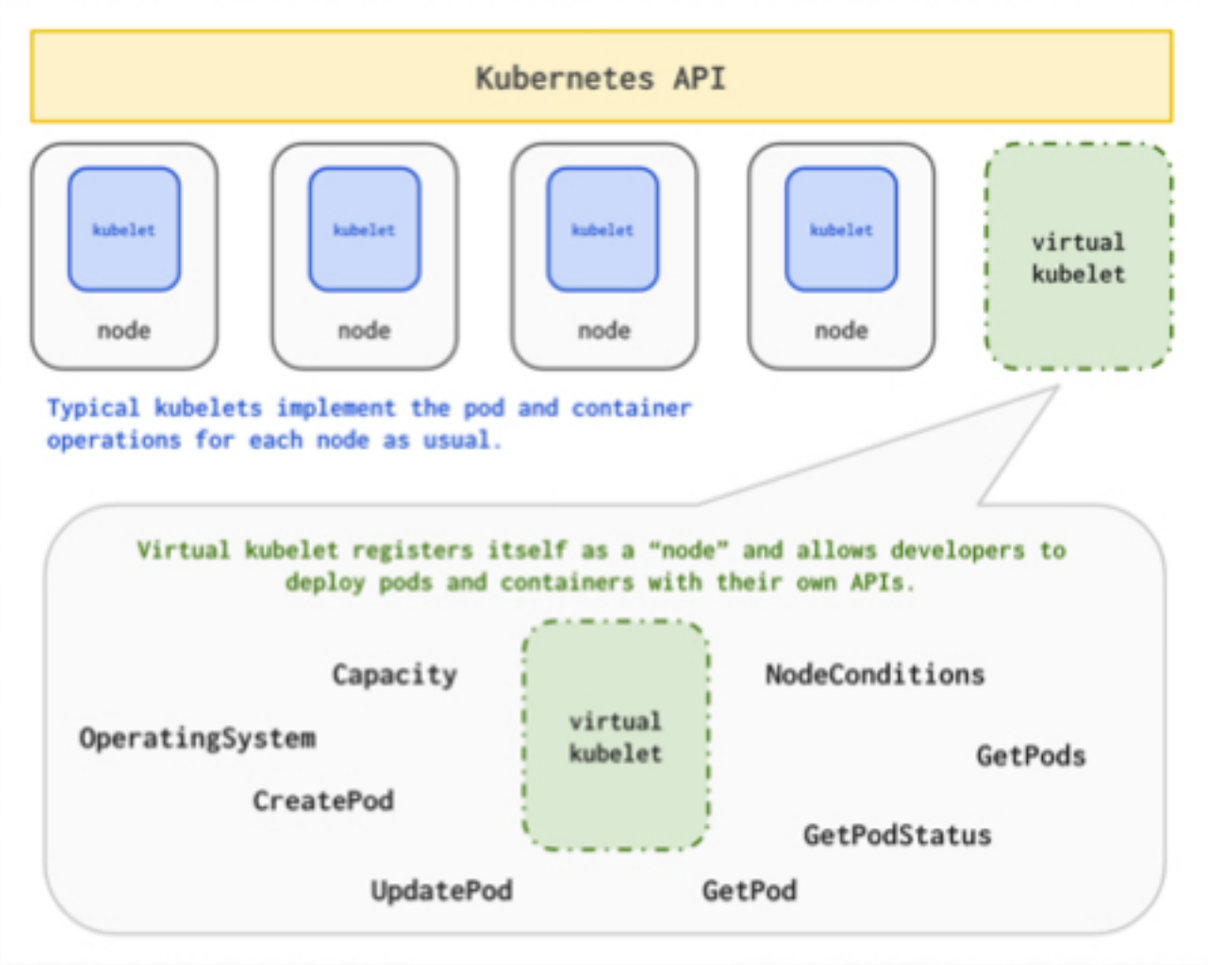
- 弹性能力较弱，扩容一个节点耗费时间较长
- 多集群、多节点都容易产生资源碎片
- 每个节点是单一的故障域，是系统高可用的瓶颈
- 节点运维与 infra 耦合较深，提高节点运维成本
  - 异构节点管理（不同 OS、设备）
  - 节点内核配置

Kubernetes Node 的本质



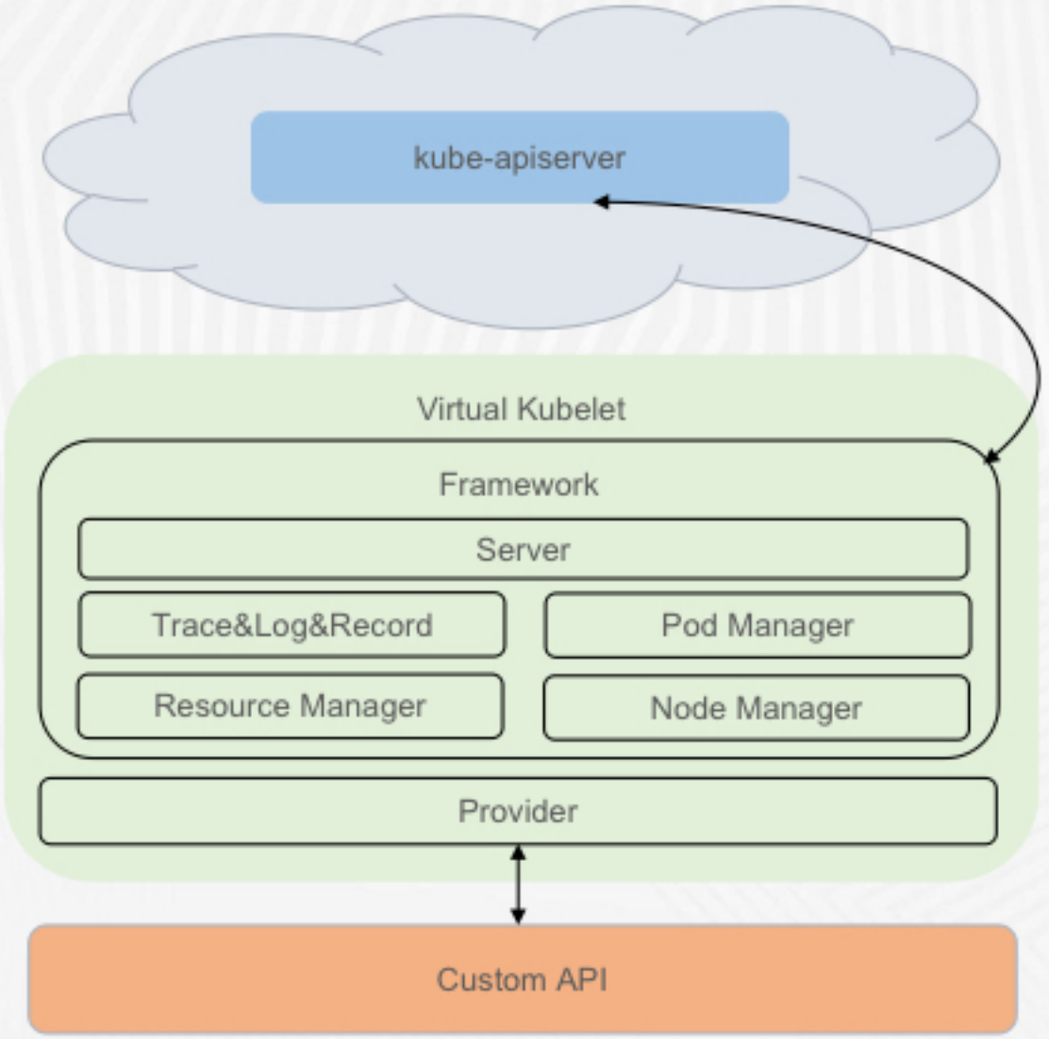
02 Virtual Kubelet 介绍

Virtual Kubelet 功能



- 开源 kubelet 实现，用于伪装成 kubelet 连接 K8s 与其他 API
  - 将节点与 Pod 概念抽象出来
  - 抽象细节由 Provider 提供
  - 框架代码较少，将自身作为节点注册到 kube-apiserver；轮询"node"与"pod"状态并上报；处理与容器的同步交互
- Virtual Kubelet 可作为一个 Pod 部署在集群中
- 适用于单集群高可用、异构节点混合使用、边缘等场景
- Kubernetes API on top, programmable back.
- 2018 年成为 CNCF sandbox 项目

Virtual Kubelet 架构




```go
type Provider interface {
        node.PodLifecycleHandler

        // GetContainerLogs retrieves the logs of a container by name from
        GetContainerLogs(ctx context.Context, namespace, podName, containerN

        // RunInContainer executes a command in a container in the pod, cop
        // between in/out/err and the container's stdin/stdout/stderr.
        RunInContainer(ctx context.Context, namespace, podName, containerNa

        // GetStatsSummary gets the stats for the node, including running p
        GetStatsSummary(context.Context) (*statsv1alpha1.Summary, error)
}
```

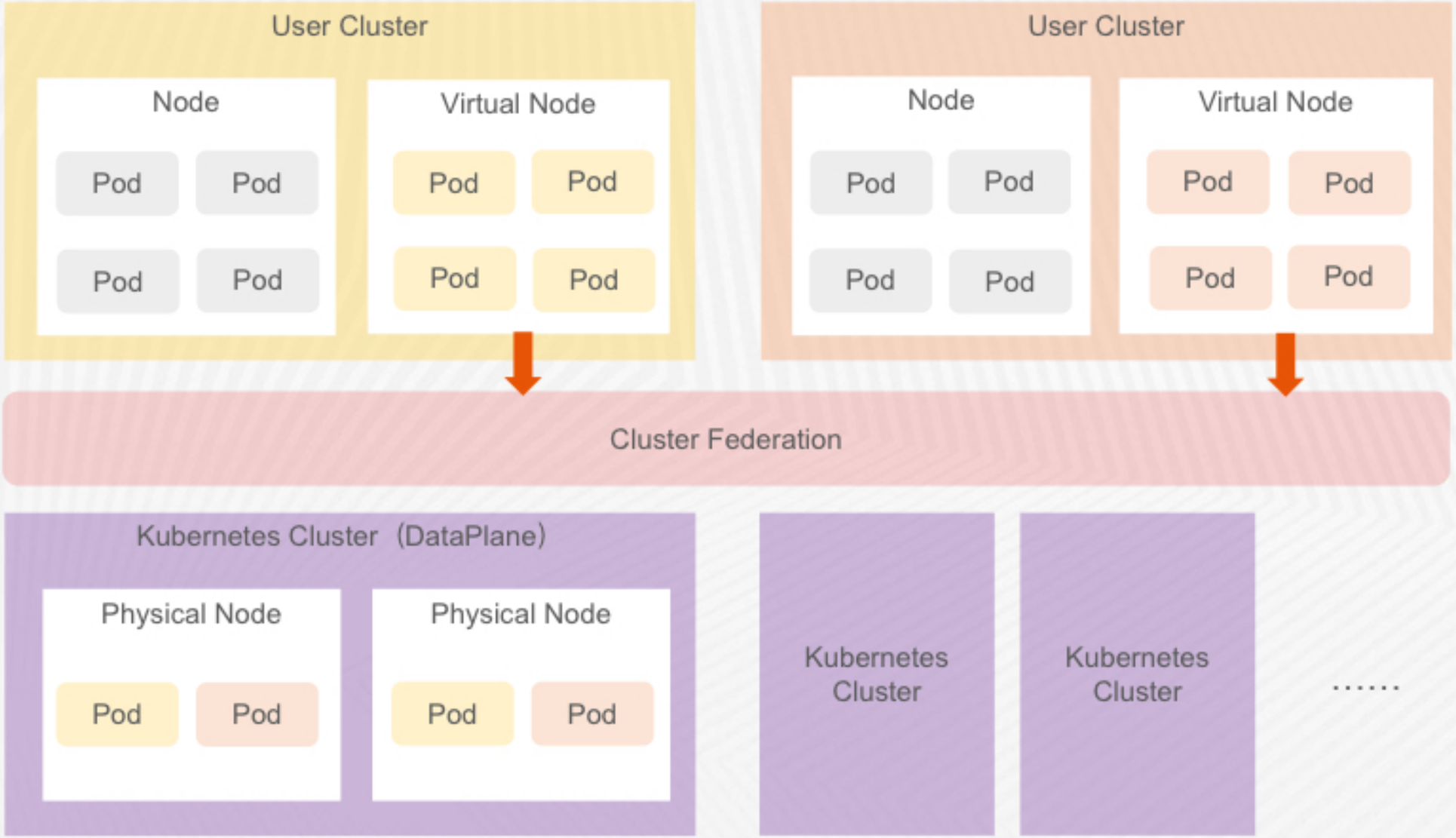
```go
type PodLifecycleHandler interface {
        // CreatePod takes a Kubernetes Pod and deploys it within the provider.
        CreatePod(ctx context.Context, pod *corev1.Pod) error

        // UpdatePod takes a Kubernetes Pod and updates it within the provider.
        UpdatePod(ctx context.Context, pod *corev1.Pod) error

        // DeletePod takes a Kubernetes Pod and deletes it from the provider. Once a pod is d
        // expected to call the NotifyPods callback with a terminal pod status where all the
        // state, as well as the pod. DeletePod may be called multiple times for the same pod
        DeletePod(ctx context.Context, pod *corev1.Pod) error

        // GetPod retrieves a pod by name from the provider (can be cached).
        // The Pod returned is expected to be immutable, and may be accessed
        // concurrently outside of the calling goroutine. Therefore it is recommended
        // to return a version after DeepCopy.
        GetPod(ctx context.Context, namespace, name string) (*corev1.Pod, error)

        // GetPodStatus retrieves the status of a pod by name from the provider.
        // The PodStatus returned is expected to be immutable, and may be accessed
        // concurrently outside of the calling goroutine. Therefore it is recommended
        // to return a version after DeepCopy.
        GetPodStatus(ctx context.Context, namespace, name string) (*corev1.PodStatus, error)

        // GetPods retrieves a list of all pods running on the provider (can be cached).
        // The Pods returned are expected to be immutable, and may be accessed
        // concurrently outside of the calling goroutine. Therefore it is recommended
        // to return a version after DeepCopy.
        GetPods(context.Context) ([]*corev1.Pod, error)
}
```

当前社区版本的局限

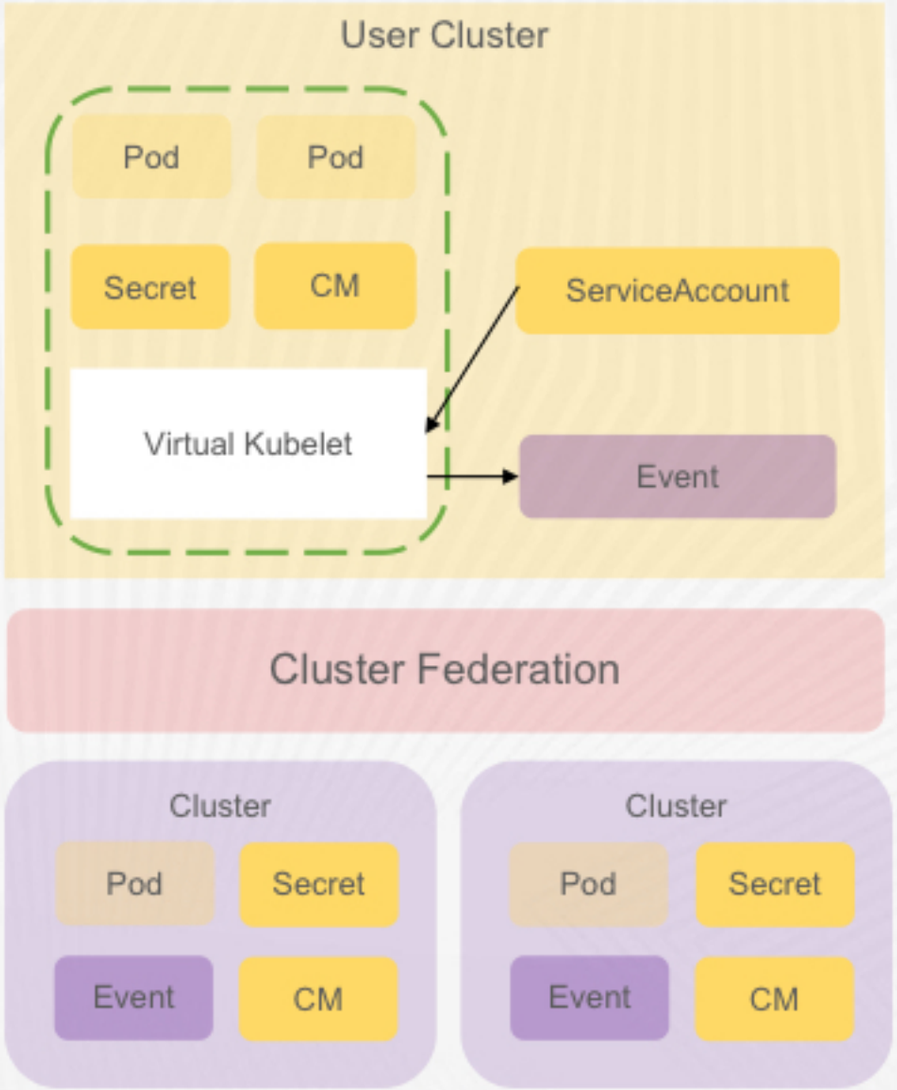
- **支持 K8s 版本较低**
  - 最新 Release 版本支持 K8s v 1.19
- **对原生 K8s 还原度不足**
  - 不支持/metrics、/metrics/cadvisor、/metrics/resource 接口
  - 不支持 logs -f
  - 不支持 Pod 关联资源的更新（cm、secret、event）
- **不支持多副本**

03 Virtual Kubelet + K8s

Virtual Kubelet + K8s 架构

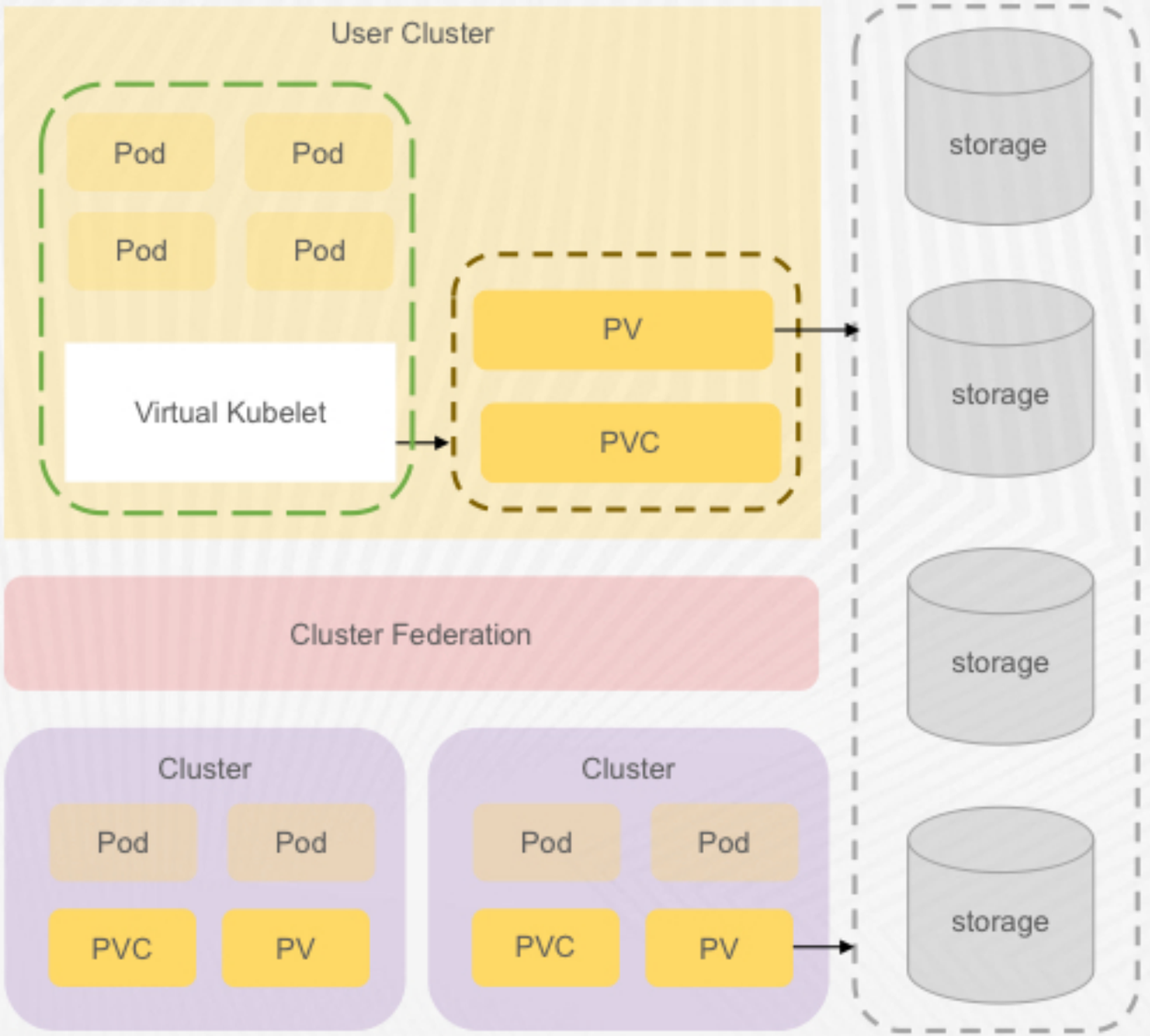


K8s 资源同步



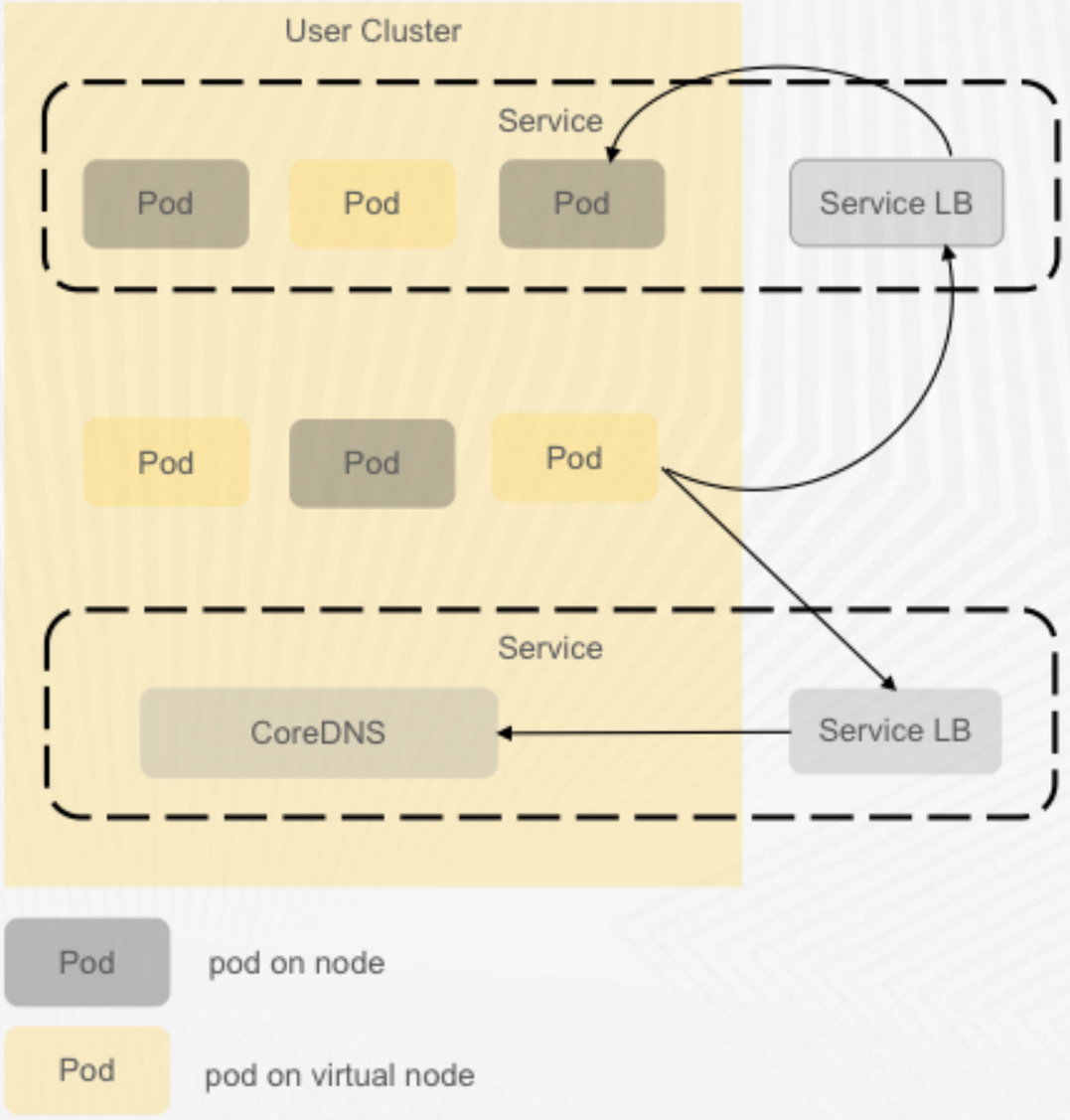
- Virtual Kubelet 额外 watch Pod 关联的 Secret、Configmap，若发生变化实时同步
  - 支持 Secret、Configmap 热更新
- 将 ServiceAccount 转换成 Secret 同步
- 查询承载 Pod 相关 Event 上报给用户集群

存储



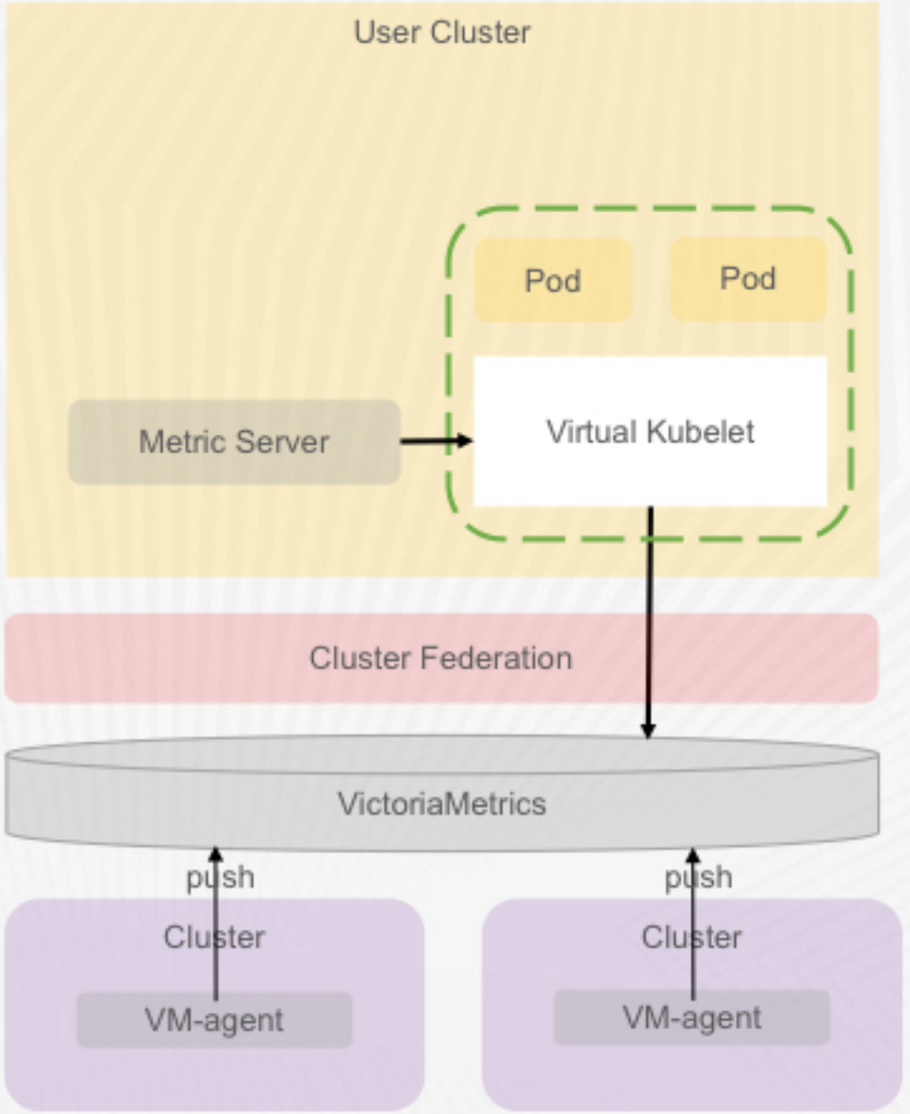
- VirtualKubelet 获取承载 Pod 相关的存储信息（CSI），同步到数据面集群
- 数据面集群根据 PVC/PV 直接挂载相应存储设备

网络



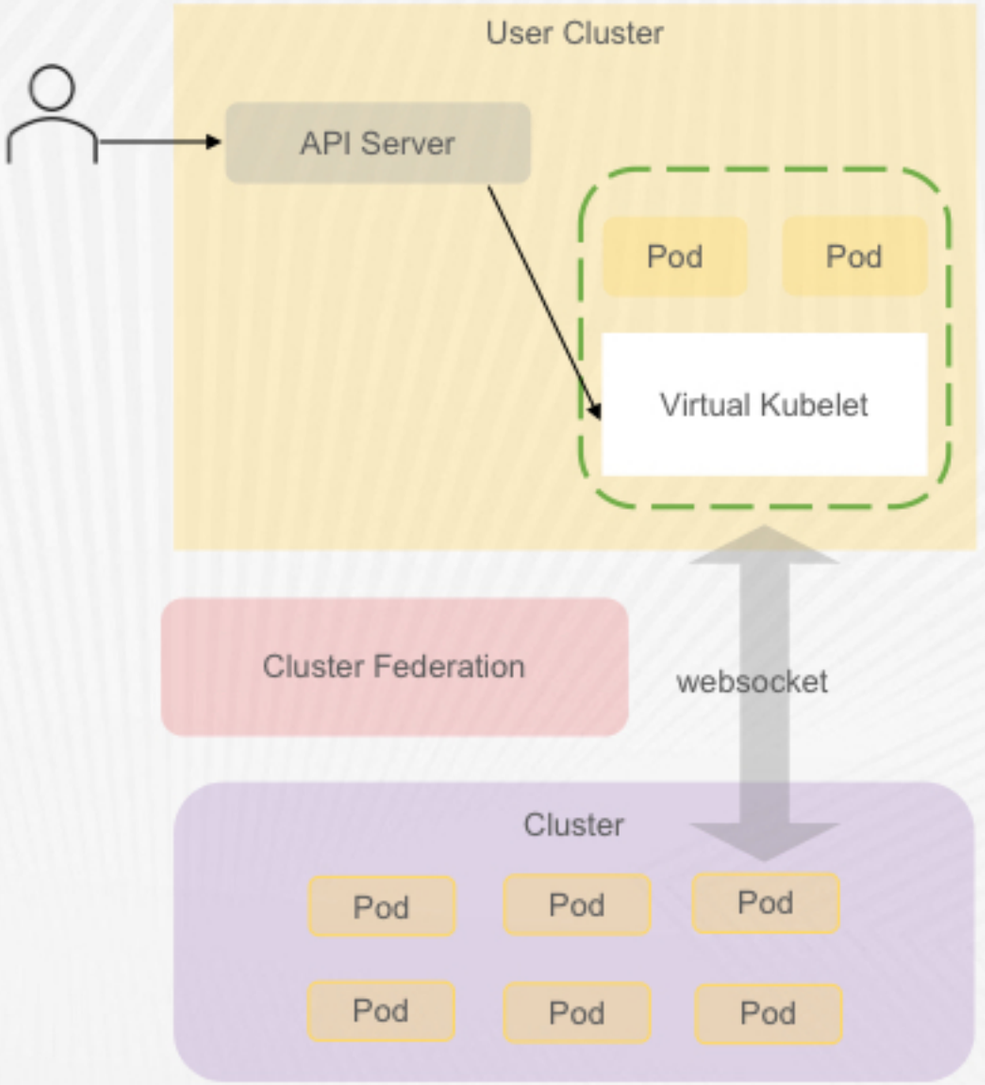
- 所有 Pod 在同一网络平面内，直接互通
- Service ClusterIP 直接下发到外部 LB 上
- 数据面 Pod 通过 LB 直接访问用户集群 Pod
- Pod 下发至数据面时，通过 DNSConfig 将 DNS 配置同步下去

监控



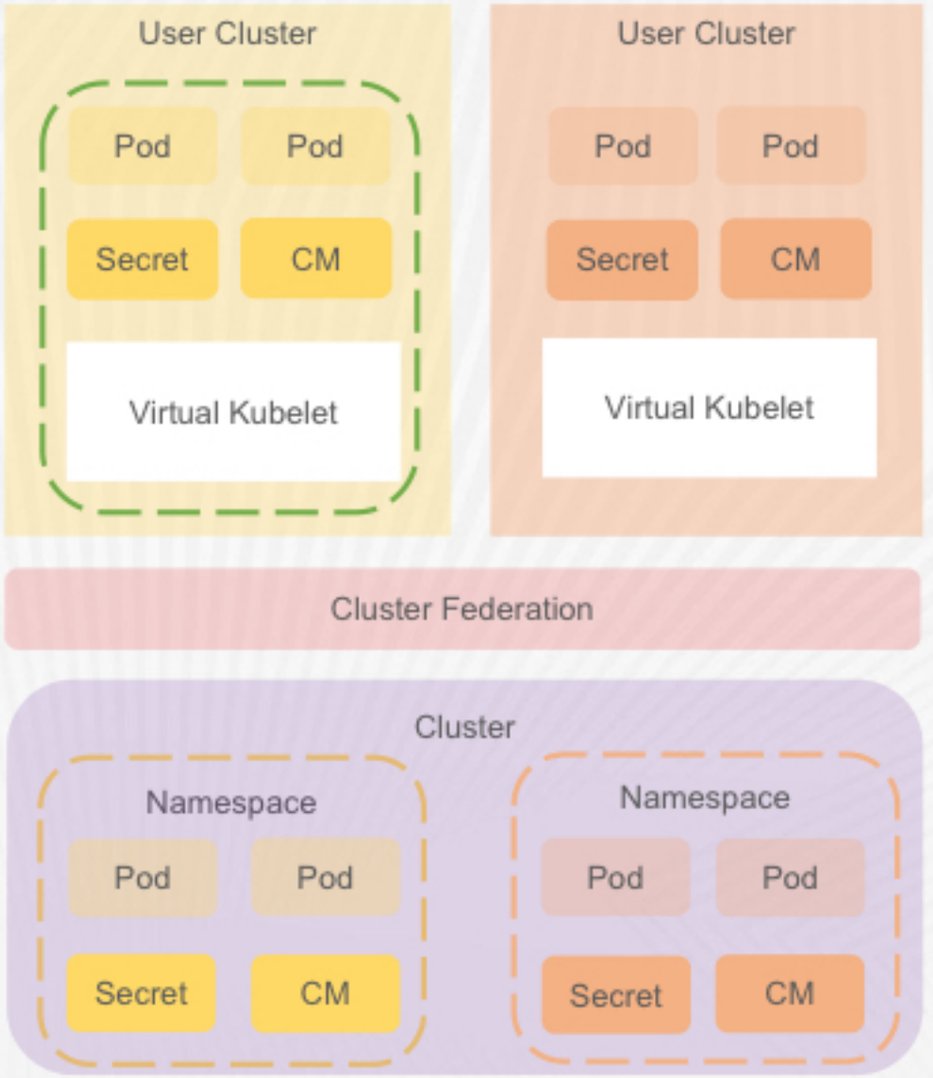
- 引入 VictoriaMetrics 收集数据面集群指标
- Virtual Kubelet 通过 VictoriaMetrics 获取所有承载 Pod 指标
- Virtual Kubelet 暴露 /metrics 等指标接口，将信息暴露给外部组件

日志 & exec



- 用户通过 Virtual Kubelet 请求获取 Pod 日志或执行 exec 时，Federation 会自动生成一个 websocket 连接
- Federation 分配 websocket 连接所需要的 token
- VirtualKubelet 通过该 websocket 与 Pod 实时通信

多租户隔离



- Virtual Kubelet 同步资源携带租户信息
- Federation 校验租户权限，使 Virtual Kubelet 只能访问自身承载资源
- Federation 限制各个租户 quota
- 数据面集群通过 namespace 隔离不同租户的资源

实践成果

- 支撑 100+ 用户集群
- 支撑 10000+ Pod
- 计算资源总体利用率提升 20%

04 规划和展望

规划和展望

- Virtual Kubelet 模块拆分和高可用架构
- 数据面集群回调 Virtual Kubelet
- Cluster Federation 朝社区标准化演进

和优秀的人 做有挑战的事

JOIN BYTEDANCE

成就一亿技术人

成为技术人交流和成长的家园

用户为本 | 求真求是 | 协作共赢 | 极客精神 | 结果导向

CSDN