



# Kubernetes

# Agenda

- O que é
- Histórico
- Principal funcionalidades
- Dinâmica de funcionamento
- Principais Componentes
- Executando Kubernetes localmente
- Cluster Kubernetes na Nuvem
- Ferramentas úteis de gerenciamento
- Helm

## O que é

"Kubernetes, also known as K8s, is an open-source system for automating deployment, scaling, and management of containerized applications."

<https://kubernetes.io/>

# Histórico

- Borg
- Omega
- Kubernetes

<https://queue.acm.org/detail.cfm?id=2898444>

## Visão geral

- Cluster: Conjunto de nós (nodes) que executam container workloads.
- Master: Controla e gerencia o estado desejado do cluster.
- Nodes: Máquinas (VMs ou servidores físicos) que executam os containers.

## Pods

- Menor unidade de deploy no Kubernetes.
- Um Pod pode conter um ou mais containers.
- Encapsulamento: Um Pod encapsula o espaço de endereço de rede e os volumes de armazenamento para seus containers.

## Pods

- Diferença fundamental: Containers são a camada de execução, enquanto Pods são a camada de abstração no Kubernetes.
- Multi-container: Um Pod pode conter múltiplos containers operando juntos.  
Compartilhamento: Containers em um mesmo Pod compartilham recursos de rede e armazenamento. Um Pod pode conter um ou mais containers.
- Encapsulamento: Um Pod encapsula o espaço de endereço de rede e os volumes de armazenamento para seus containers.



# Pods

192.168.2.34 - 128MB - 2vCPU

Container X



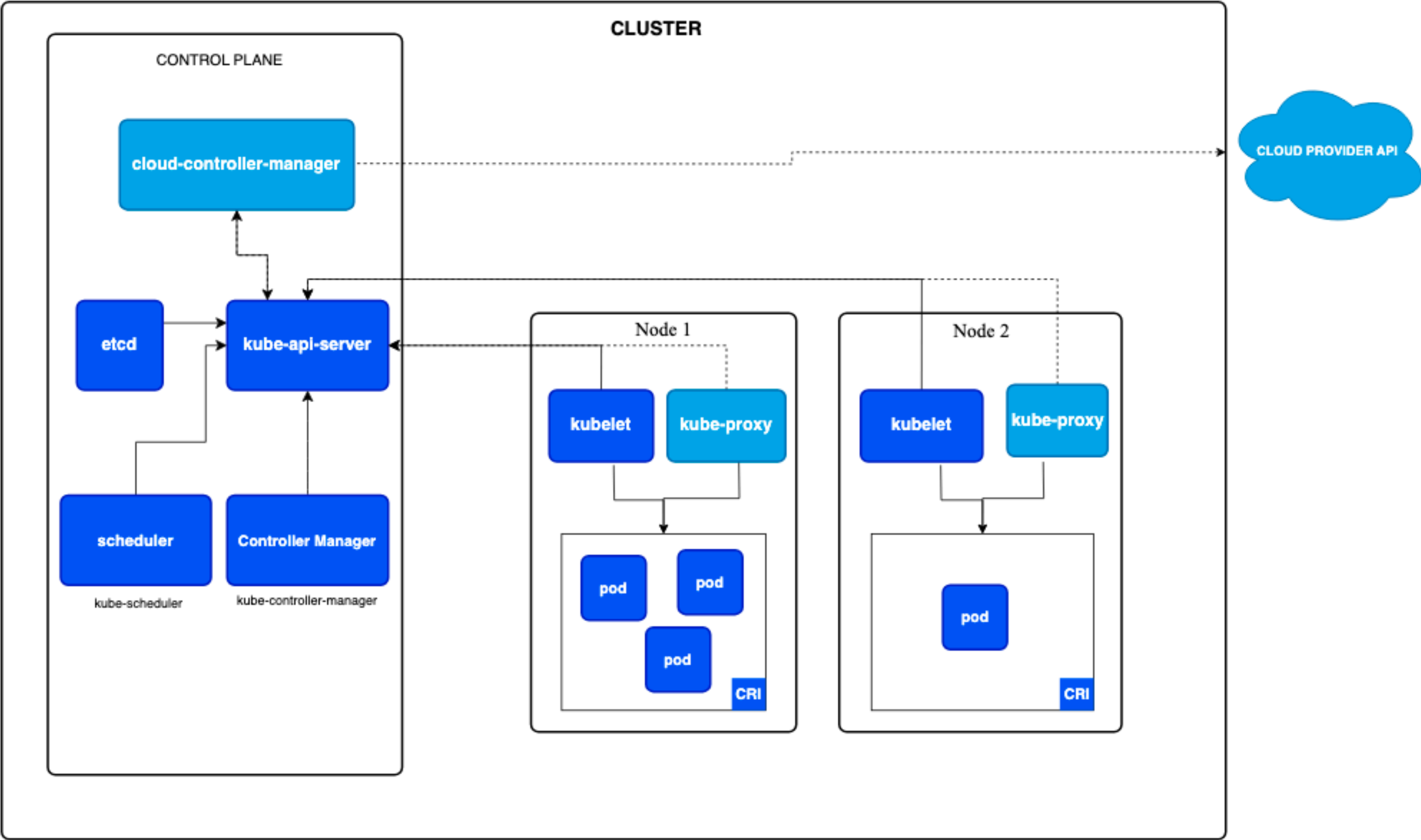
# Pods

192.168.2.34 - 128MB - 2vCPU

Container X

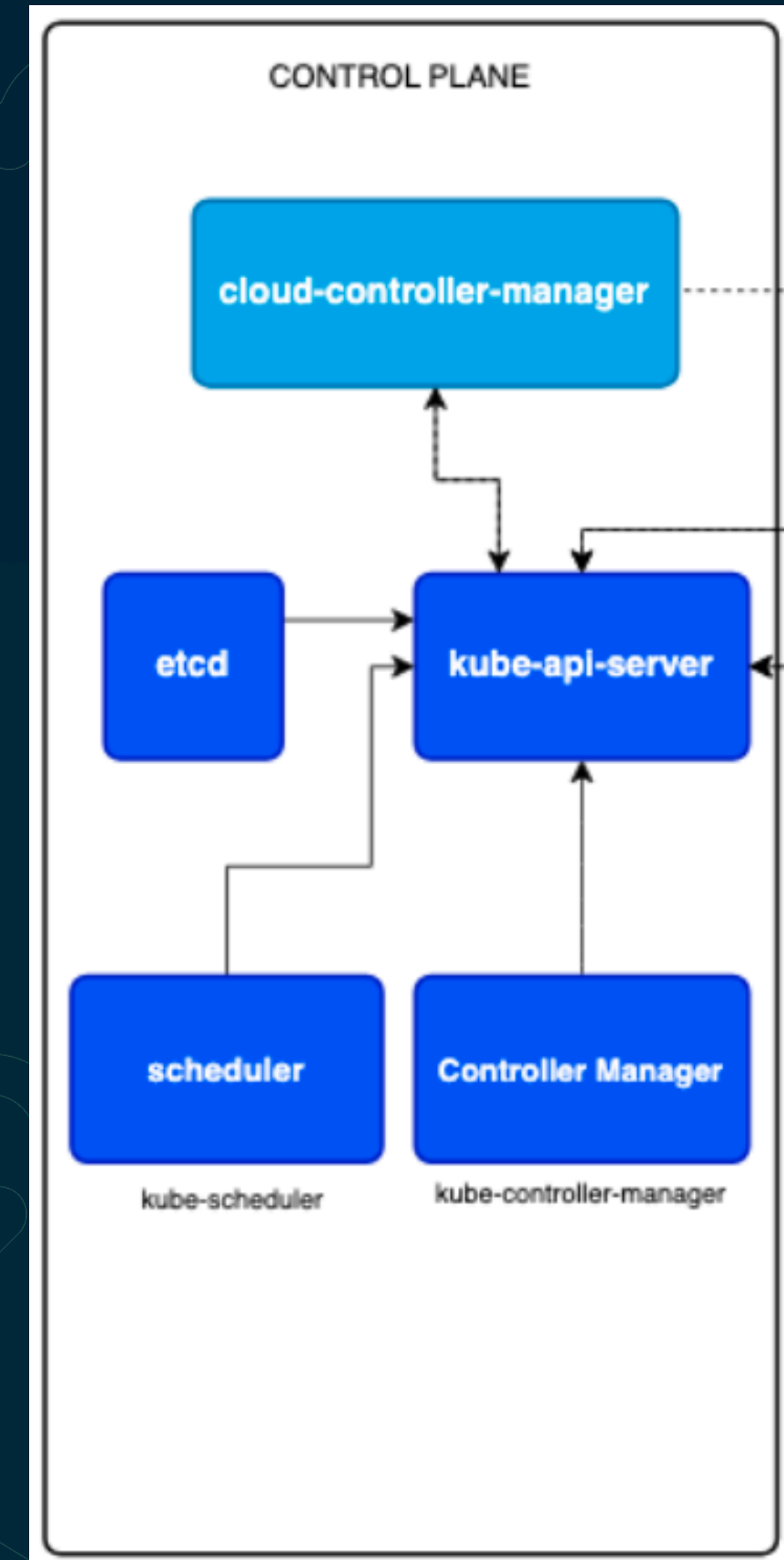
Container Y

# Arquitetura



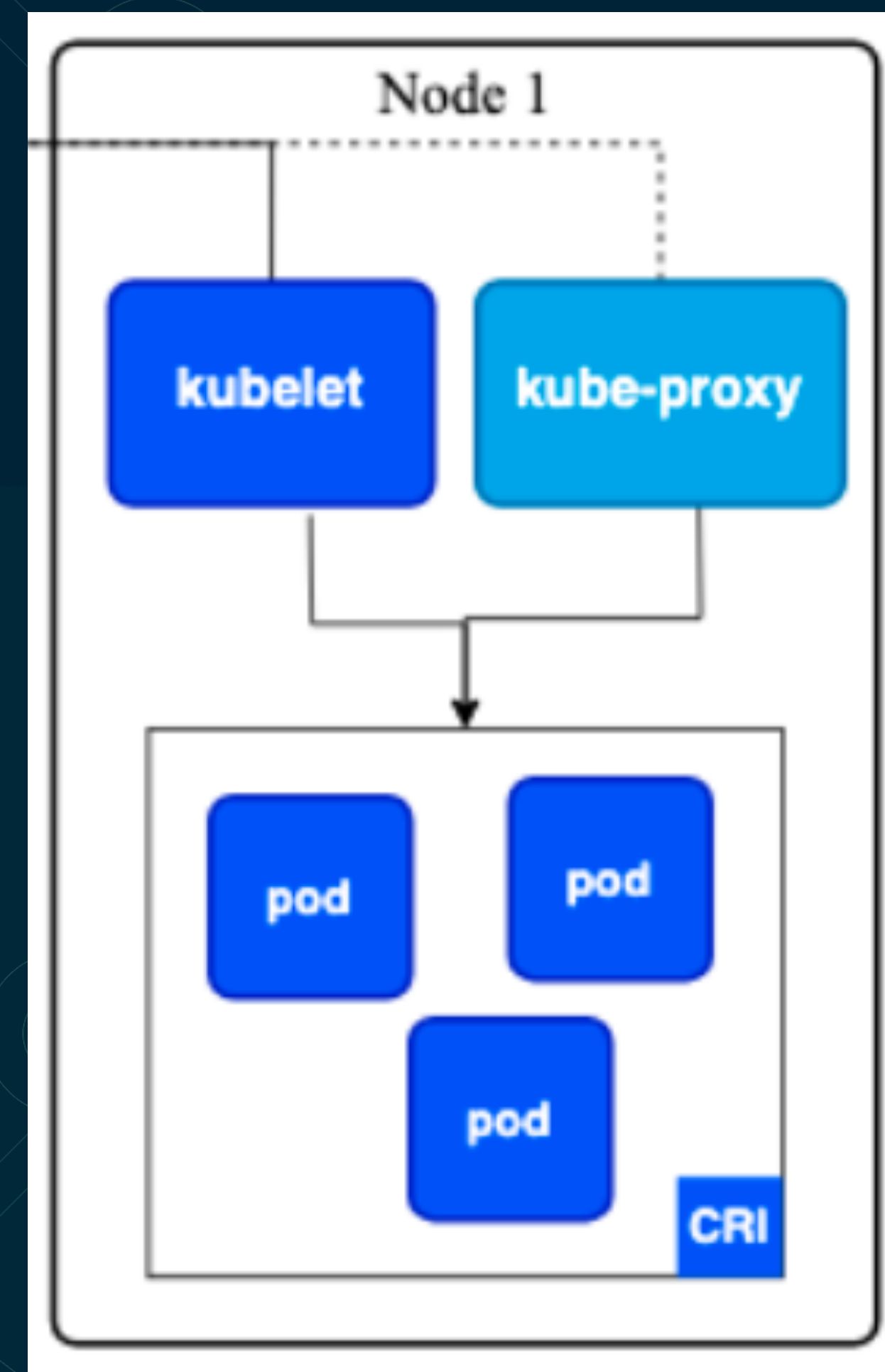
## Componentes do Master

- API Server: Ponto central de gerenciamento e comunicação.
- Scheduler: Distribui os Pods entre os Nodes.
- Controller Manager: Orquestra os controladores de nó, endpoint, serviço e outros.
- Etcd: Armazenamento consistente e de alta disponibilidade do estado do cluster.



## Componentes dos Nodes

- Kubelet: Agente que executa e gerencia os estados dos containers no nó.
- Kube-proxy: Gerencia o networking dos containers, redirecionando as requisições.
- Container Runtime: Software responsável por rodar os containers



## Docker no Kubernetes

- No passado o container runtime padrão do Kubernetes era o Docker
- O Kubernetes começou a padronizar a interface dos containers runtime
- Docker não fez essa padronização. Ele usava um componente que era difícil de manter chamado dockershin
- Por conta disso, Kubernetes parou de utilizar o Docker como container runtime
- Isso não impede que as imagens geradas com docker, rodem no Kubernetes

<https://kubernetes.io/blog/2022/02/17/dockershim-faq/>

# ReplicaSets

- Gerencia o estado dos Pods
- Determina a quantidade de Pods em execução
- Garante a quantidade mínima de Pods em execução



# ReplicaSets

Replicas: 10

192.168.2.34 - 128MB - 2vCPU

Container X

Container Y



## Deployments

- Exerce controle sob os ReplicaSets
- A cada mudança de versão do container, o Deployment cria um novo ReplicaSet
- Caso a versão anterior seja aplicada novamente, ele utiliza a versão antiga (já criada) do ReplicaSet

# Deployments

Replicas: 10

192.168.2.34 - 128MB - 2vCPU

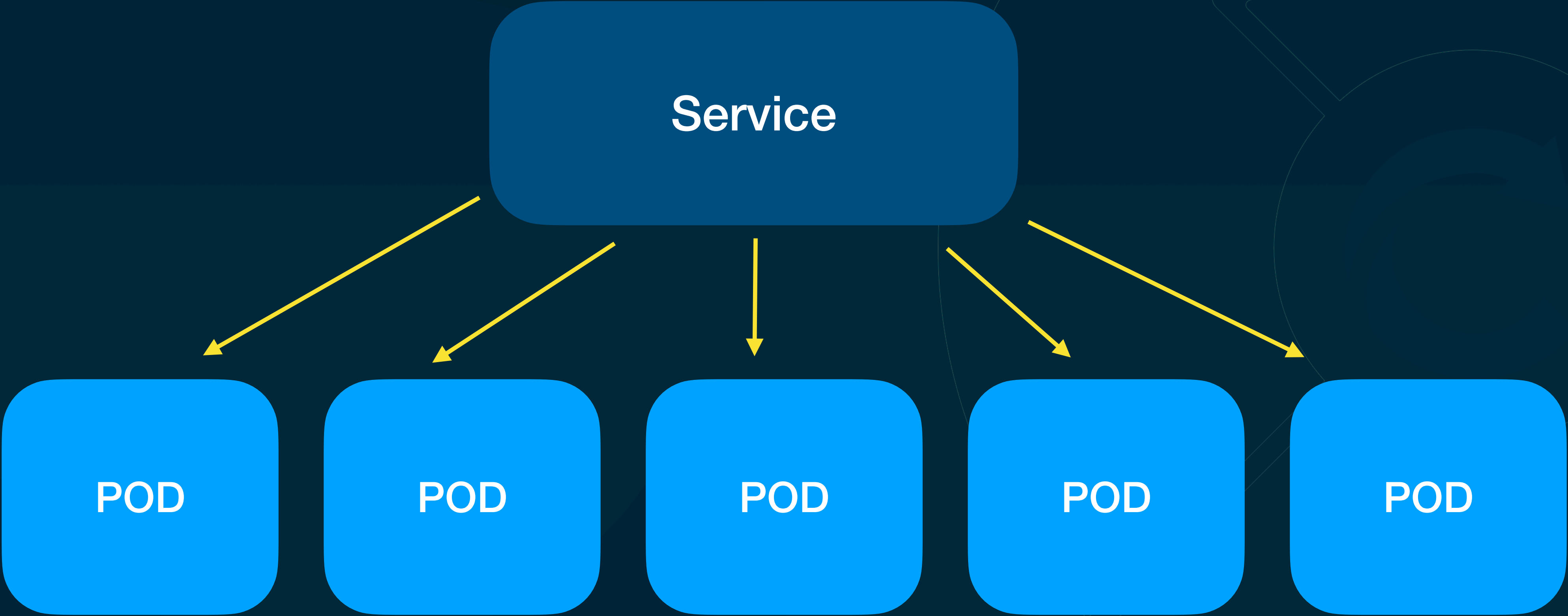
Container X

Container Y

## Services

- Pontos de entrada para os PODs
- PODs não são acessados diretamente. Você acessa o service que balanceia a carga e envia a requisição para um dos PODs
- Resolução automática de DNS.
- 4 tipos de services

# Services



## Services

- ClusterIP: Chamadas internas no Kubernetes
- LoadBalancer: IP Externo para acesso ao cluster
- NodePort: Acesso externo ao cluster através de uma porta pré-definida que pode ser acessada através do IP externo de qualquer um dos nomes
- ExternalName: Mapeia o serviço para um endereço externo

## Funcionalidades Chave do Kubernetes

- Implantação e atualização automáticas de aplicações (Rollouts/Rollbacks).
- Auto reparação: Reinicia containers que falham, substitui e reescala containers.
- Balanceamento de carga e descoberta de serviço.
- Gerenciamento de configurações e segredos.

## Services

- Services: Abstrai o acesso aos Pods, permitindo comunicação interna e externa.
- Networking: Soluções de rede que permitem a comunicação entre os componentes do cluster. Ingress:
- Gerencia o acesso externo aos serviços do cluster, roteamento de tráfego e SSL/TLS.