

O que é Kubernetes?

O Kubernetes é um orquestrador de código aberto para implantação de aplicações containerizadas.

Ele foi originalmente desenvolvido pelo Google, inspirado em uma década de experiência com a implantação de sistemas escaláveis e confiáveis em containers por meio de APIs orientada a aplicações (tudo começou com o Borg 🐼).

É um dos maiores e mais populares projetos de código aberto do mundo.

Criar os containers

Manter o estado ideal do cluster

Escalável

Gerenciar seu funcionamento

Portável

Mestre / Worker

Benefícios

- Velocidade:** não só a relação ao número de funcionalidades que você é capaz de lançar por hora ou por dia, mas também à quantidade de itens que pode lançar, ao mesmo tempo que mantém a disponibilidade.

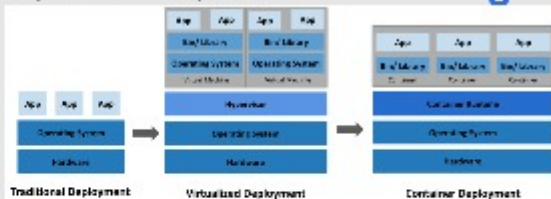
Imutabilidade

Configuração declarativa

Autocorreção

- Escalabilidade:** escalar seu serviço é uma questão de alterar um número no arquivo de configuração (e deixar o Kubernetes cuidar do resto). Ou configurar o autoscaling.
- Abstração da infraestrutura:** o Kubernetes possui vários pluggins que possibilitam a abstração de uma nuvem em particular... por exemplo: sabe como criar balanceadores de carga para as principais nuvens públicas.
- Eficiência:** o Kubernetes oferece ferramentas para automatizar a distribuição das aplicações em um cluster de máquinas, garantindo o nível mais alto de utilização, em comparação com o que é possível fazer com ferramentas tradicionais.

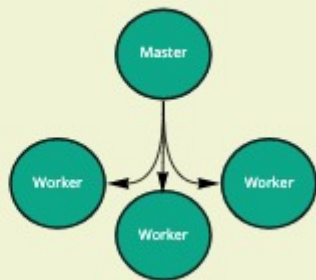
Começando a Entender a Arquitetura



- Traditional:** aplicações em execução em servidores físicos. Não era possível limitar os recursos para cada aplicação.
- Virtualized:** possibilidade de execução de várias VMs em um único servidor. Aplicações isoladas em cada VM. Melhor utilização de recursos e escalabilidade. Cada VM possui seu próprio SO.
- Container:** semelhante às VMs, mas com propriedades de isolamento flexibilizados para compartilhar o SO entre aplicações. Leves e portáveis (nuvem e SO).

Clusters

Ao implantar o Kubernetes você obtém um cluster.

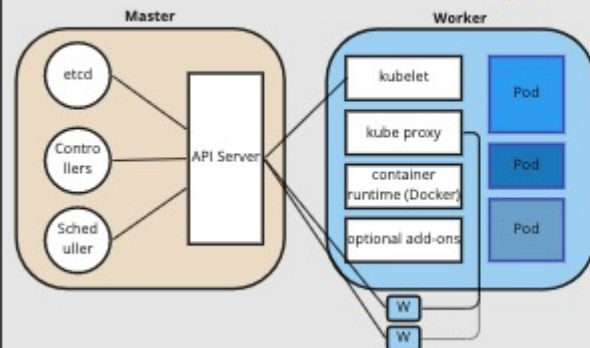


O nó "master" é o responsável pelo gerenciamento do cluster, já os "workers" vão de fato conter os containers rodando as aplicações.

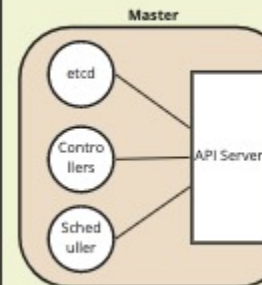
Exemplo: 3 instâncias EC2 na Amazon com Kubernetes instalado

```
kubectl get nodes
NAME             STATUS   AGE
ip-10-0-1-10     Ready    1m
ip-10-0-1-11     Ready    1m
ip-10-0-1-12     Ready    1m
```

Componentes do Kubernetes



Master



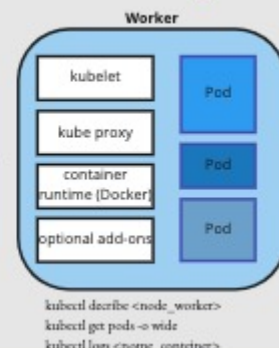
- API Server:** é o componente que expõe a API do Kubernetes. É o frontend para a camada de gerenciamento (ex: kubectl faz chamadas para esse componente).
- etcd:** é um armazenamento do tipo chave/valor no qual o kubernetes armazena todas as informações (quais nós existem, recursos dos nós, estados, etc). [Meus artigos sobre etcd.](#)
- Controllers:** é o responsável por executar os controladores de recursos, por exemplo, os deployments.
- Scheduler:** decide em quais nós executar os pods.

```
kubectl get componentstatuses
```

```
kubectl describe <node_master>
```

Worker

- kubelet:** garante que os containers estejam sendo executados em um Pod (inicia as cargas de trabalho e monitora seu status).
- kube proxy:** balanceia e roteia requisições entre os Pods.
- container runtime:** agente de execução do container. Em geral é o Docker, mas o Kubernetes aceita outros como rkt.
- optional add-ons:** recursos complementares (como DNS, dashboard, etc).
- Pod:** é o menor objeto do kubernetes, é onde são organizados os containers. Um Pod pode possuir vários containers.



```
kubectl describe <node_worker>
kubectl get pods -o wide
kubectl logs <nome_container>
```

Recomendações de Leitura

Esse material foi baseado na bibliografia abaixo:

Kubernetes Básico
Bredan Burns, Joe Beda e Kelsey Hightower

Devops Nativo de Nuvem com Kubernetes
John Arundel e Justin Domingus

Kubernetes Tudo sobre Orquestração de Containers
Lucas Santos

Documentação do Kubernetes disponível em:
<https://kubernetes.io/>



Marcelo Ortiz
Engenheiro da Computação
[Linkedin](#)

Visitem meus canais:

[Youtube](#)

[medium](#)

miro

O que é Kubernetes?



O Kubernetes é um orquestrador de código aberto para implantação de aplicações containerizadas.

Ele foi originalmente desenvolvido pelo Google, inspirado em uma década de experiência com a implantação de sistemas escaláveis e confiáveis em containêires por meio de APIs orientada a aplicações (tudo começou com o Borg 😊).

É um dos maiores e mais populares projetos de código aberto do mundo.

Criar os contêiners

Manter o estado ideal
do cluster

Escalável

Gerenciar seu
funcionamento

Portável

Mestre / Worker

- **Velocidade:** não só a relação ao número de funcionalidades que você é capaz de lançar por hora ou por dia, mas também à quantidade de itens que pode lançar, ao mesmo tempo que mantém a disponibilidade.

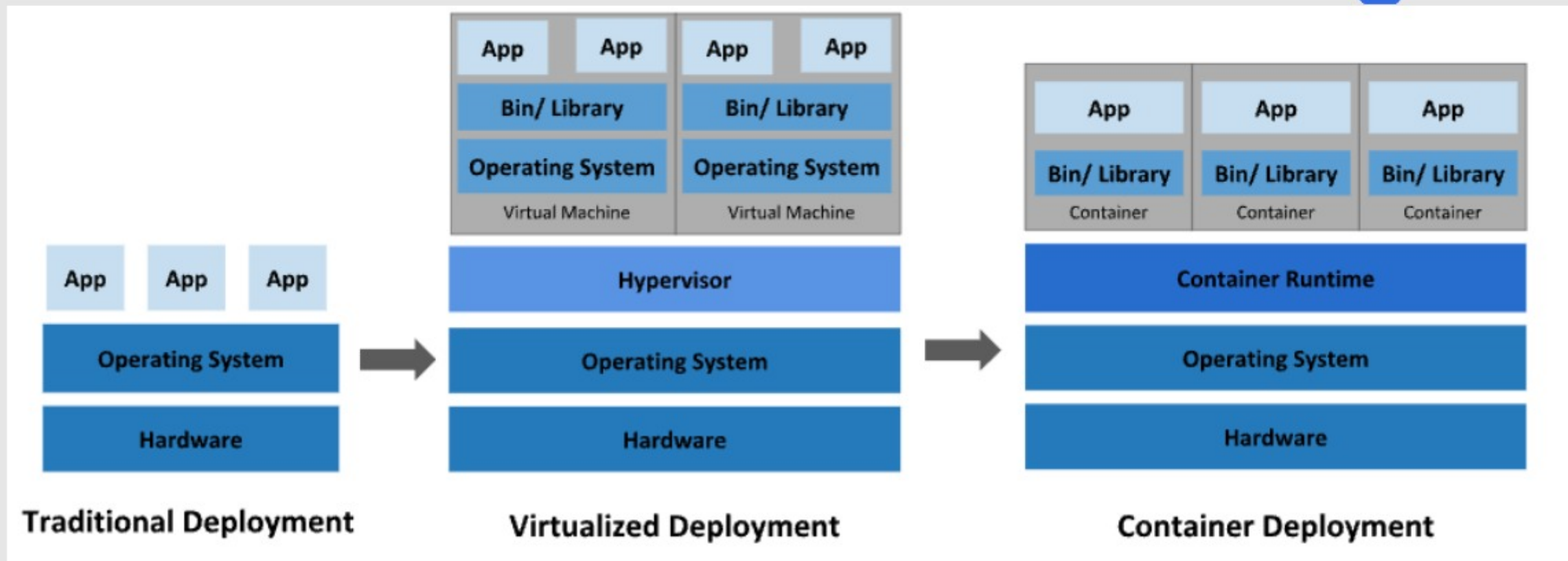
Imutabilidade

Configuração
declarativa

Autocorreção

- **Escalabilidade:** escalar seu serviço é uma questão de alterar um número no arquivo de configuração (e deixar o Kubernetes cuidar do resto). Ou configurar o autoscaling.
- **Abstração da Infraestrutura:** o Kubernetes possui vários pluggins que possibilitam a abstração de uma nuvem em particular... por exemplo: sabe como criar balanceadores de carga para as principais nuvens públicas.
- **Eficiência:** o Kubernetes oferece ferramentas para automatizar a distribuição das aplicações em um cluster de máquinas, garantindo o nível mais alto de utilização, em comparação com o que é possível fazer com ferramentas tradicionais.

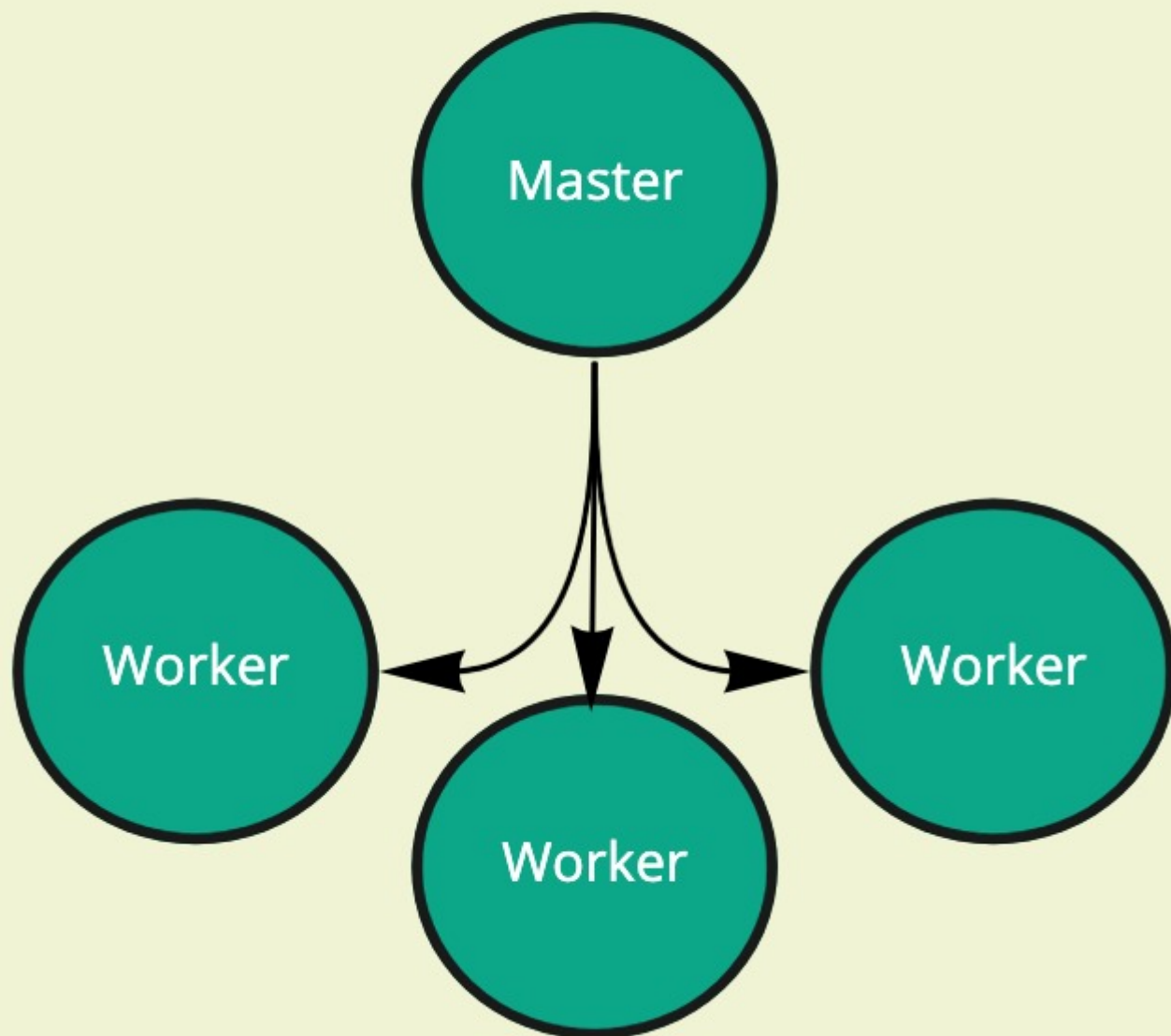
Começando a Entender a Arquitetura



- **Traditional:** aplicações em execução em servidores físicos. Não era possível limitar os recursos para cada aplicação.
- **Virtualized:** possibilidade de execução de várias VMs em um único servidor. Aplicações isoladas em cada VM. Melhor utilização de recursos e escalabilidade. Cada VM possui seu próprio SO.
- **Container:** semelhante às VMs, mas com propriedades de isolamentos flexibilizados para compartilhar o SO entre aplicações. Leves e portáteis (nuvem e SO).

Clusters

Ao implantar o Kubernetes você obtém um cluster.



O nó "**master**" é o responsável pelo gerenciamento do cluster, já os "**workers**" vão de fato conter os contêineres rodando as aplicações.

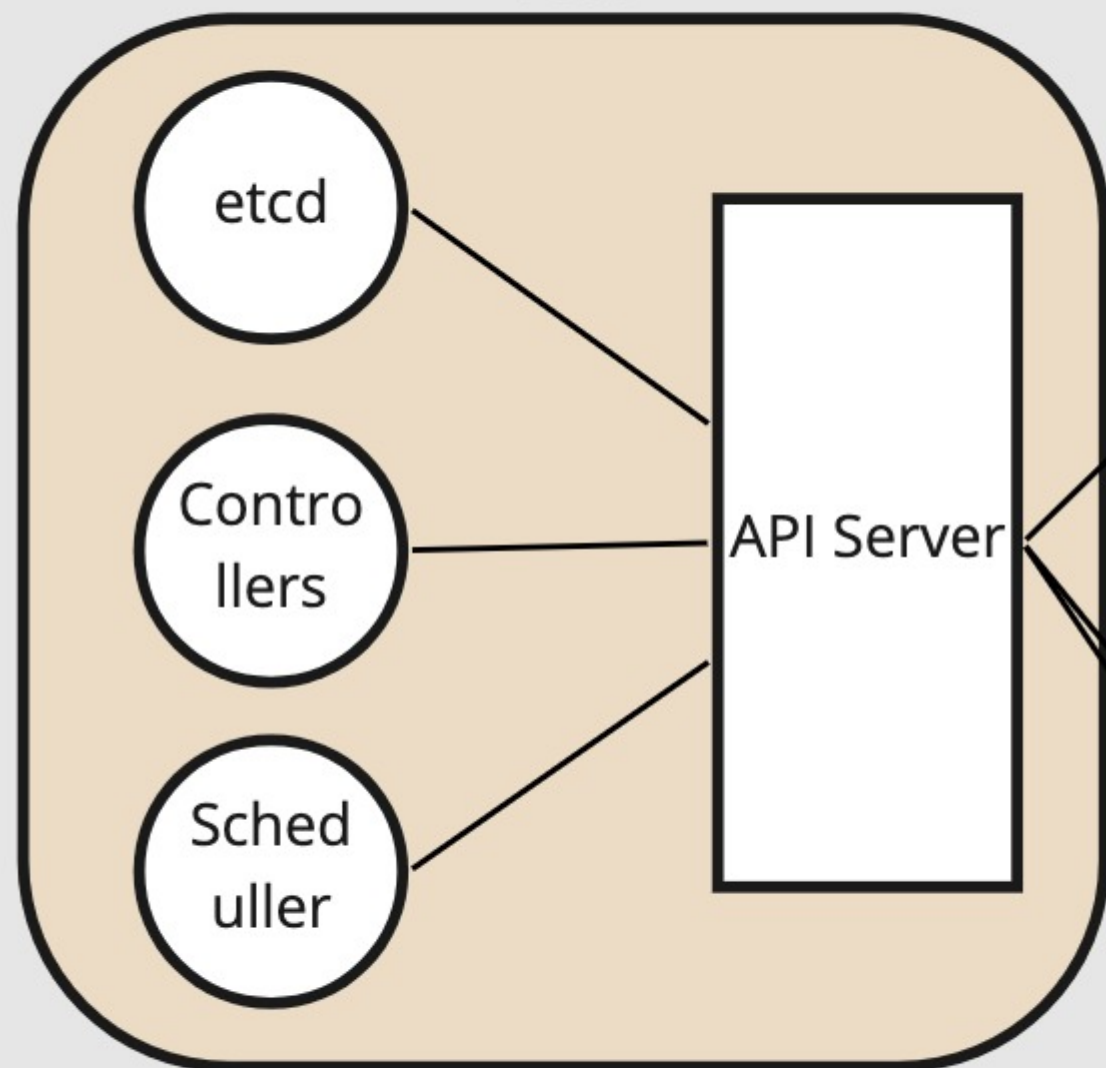
Exemplo: 3 instâncias EC2 na Amazon com Kubernetes instalado

```
root@ip-172-31-83-166:~# kubectl get nodes
```

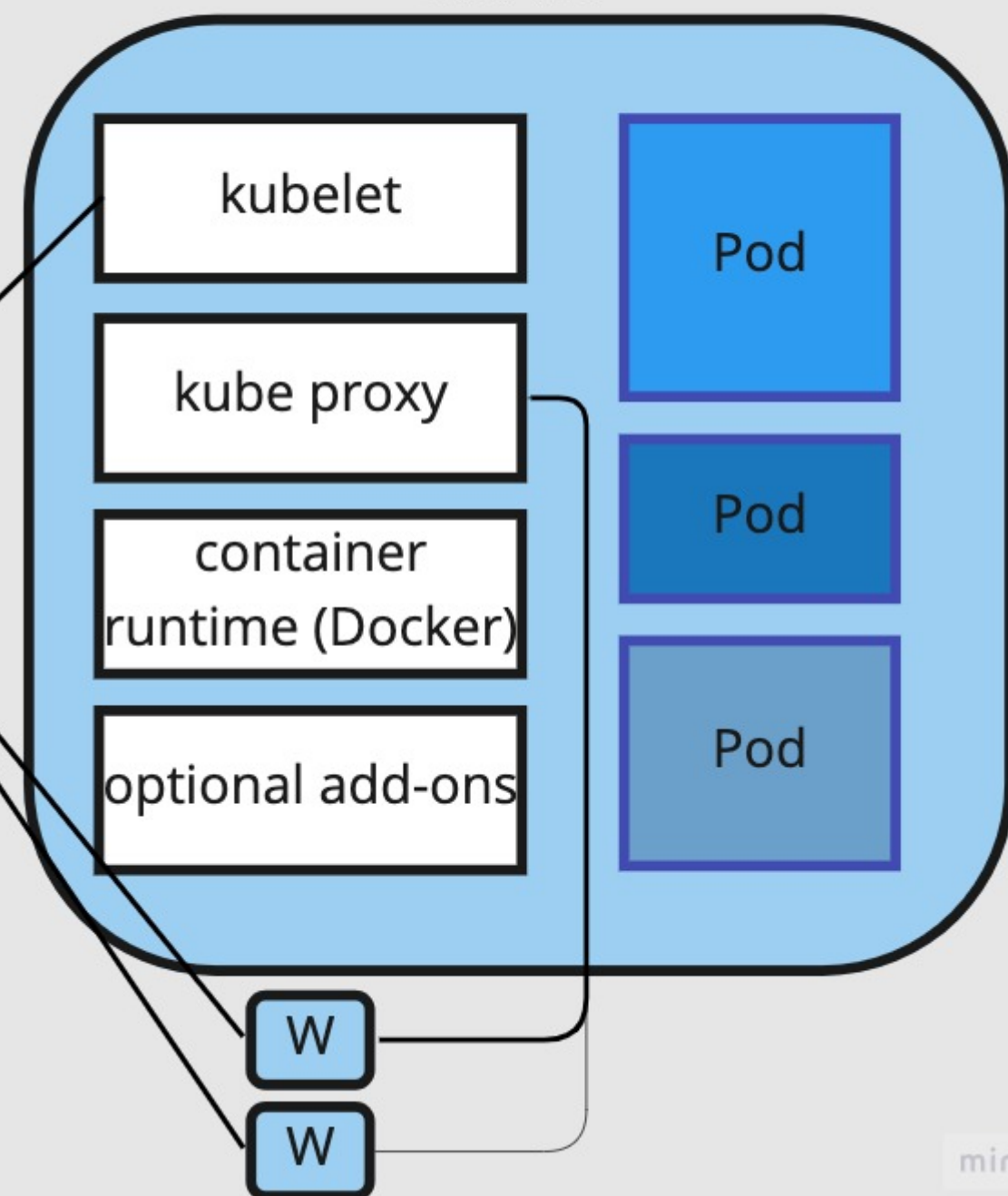
NAME	STATUS	ROLES	AGE	VERSION
ip-172-31-83-166	Ready	control-plane,master	4d16h	v1.23.5
ip-172-31-95-112	Ready	<none>	4d16h	v1.23.5
ip-172-31-95-24	Ready	<none>	4d16h	v1.23.5

Componentes do Kubernetes

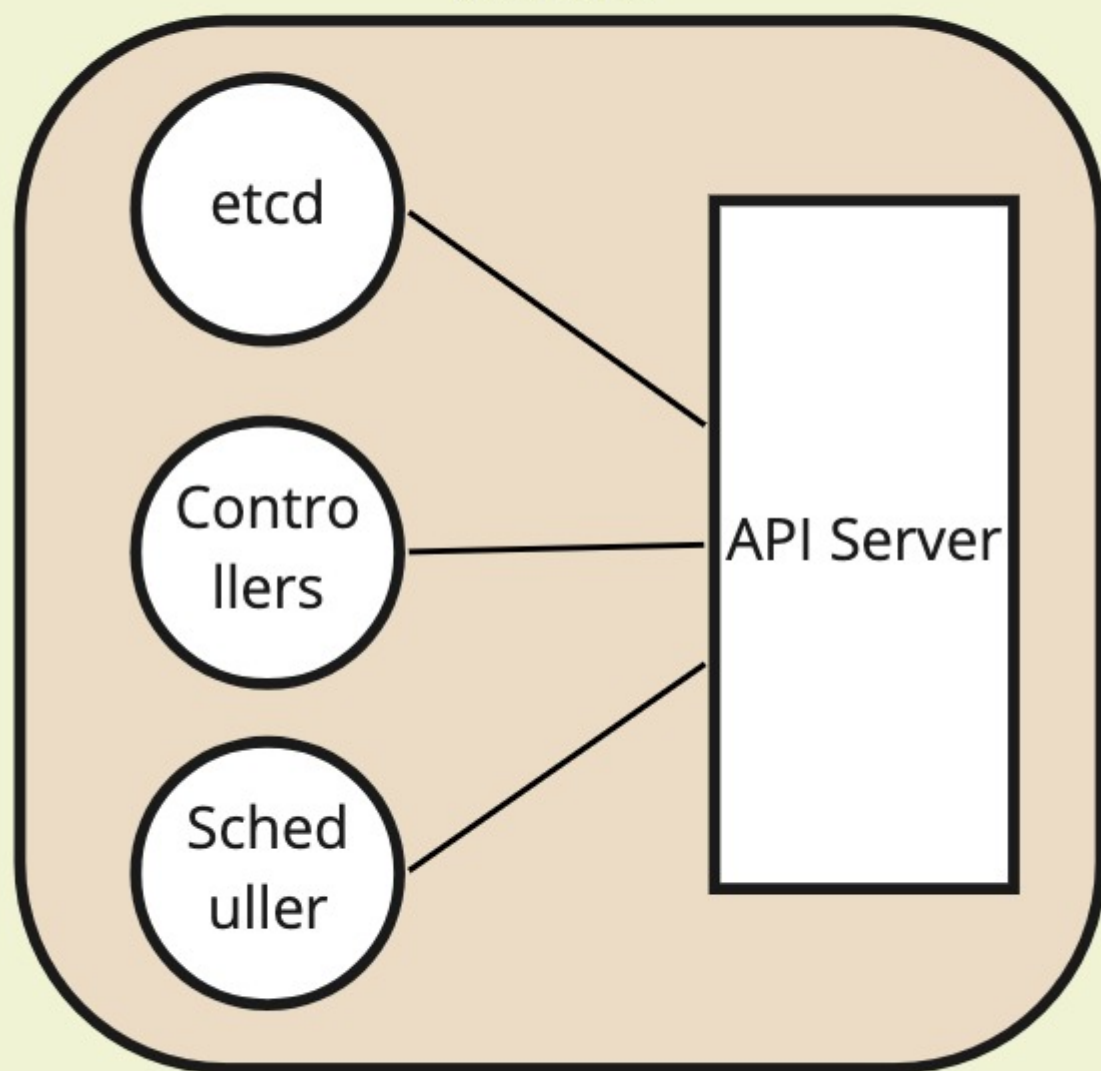
Master



Worker



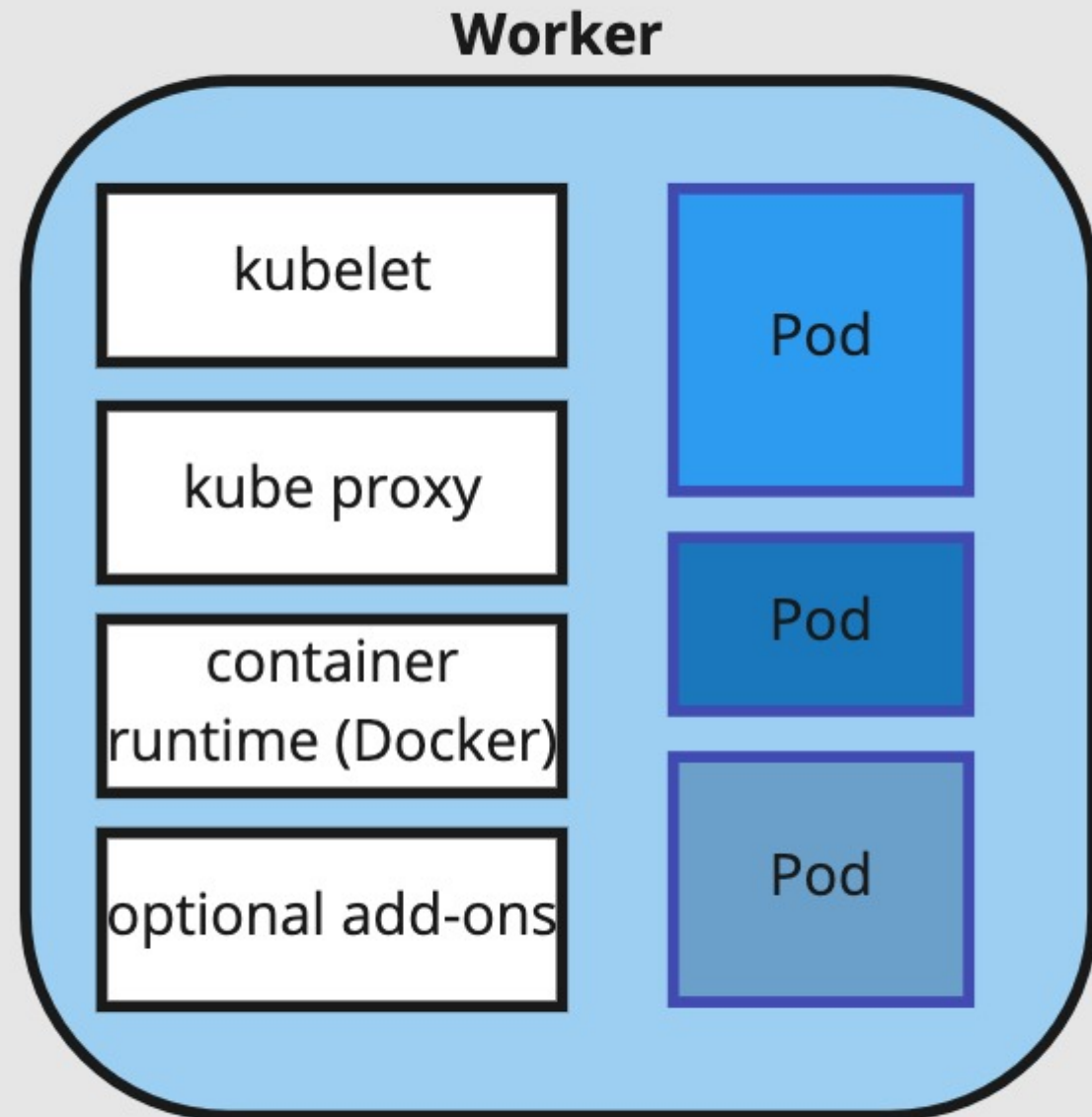
Master



- **API Server:** é o componente que expõe a API do Kubernetes. É o frontend para a camada de gerenciamento (ex: kubectl faz chamadas para esse componente).
- **etcd:** é um armazenamento do tipo chave/valor no qual o kubernetes armazena todas as informações (quais nós existem, recursos dos nós, estados, etc). [Meus artigos sobre etcd.](#)
- **Controllers:** é o responsável por executar os controladores de recursos, por exemplo, os deployments.
- **Scheduler:** decide em quais nós executar os pods.

```
kubectl get componentstatuses  
kubectl describe <node_master>
```


- **kubelet**: garante que os contêineres estejam sendo executados em um Pod (inicia as cargas de trabalho e monitora seu status).
- **kube proxy**: balanceia e roteia requisições entre os Pods.
- **container runtime**: agente de execução do contêiner. Em geral é o Docker, mas o Kubernetes aceita outros como rkt.
- **optional add-ons**: recursos complementares (como DNS, dashboard, etc).
- **Pod**: é o menor objeto do kubernetes, é onde são organizados os contêineres. Um Pod pode possuir vários contêineres.



```
kubectl describe <node_worker>  
kubectl get pods -o wide  
kubectl logs <nome_container>
```


Recomendações de Leitura

Esse material foi baseado na bibliografia abaixo:

Kubernetes Básico

Bredan Burns, Joe Beda e Kelsey Hightower

Devops Nativo de Nuvem com Kubernetes

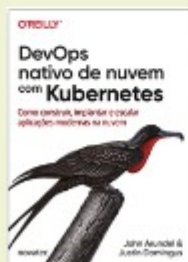
John Arundel e Justin Domingus

Kubernetes Tudo sobre Orquestração de
Contêineres

Lucas Santos

Documentação do Kubernetes disponível em:

<https://kubernetes.io/>



Marcelo Ortiz

Engenheiro da Computação

[Linkedin](#)

Visitem meus canais:

[Youtube](#)

[medium](#)