



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br

Introdução á Kubernetes



kubernetes



Introdução á Kubernetes

O kubernetes, conhecido também como K8s, foi criado pela Google, baseado na experiência deles no gerenciamento de containers em produção, ou seja, além do desenvolvimento de sistemas, quando o sistema está pronto e é publicado.

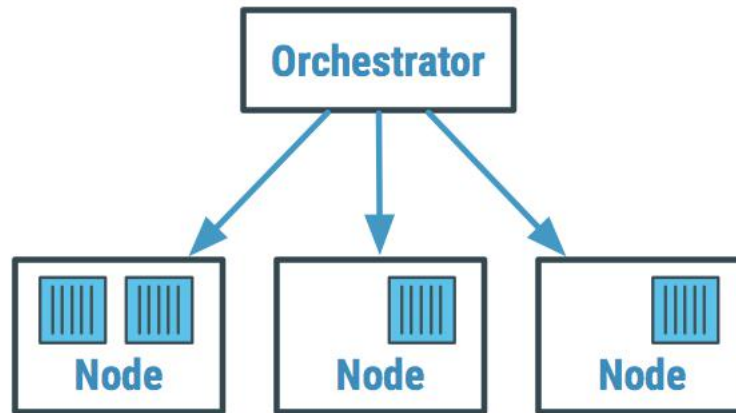
A evolução do kubernetes e ampla adoção se deu após este projeto se tornar open-source, ou seja, de código aberto, na qual qualquer pessoa ou empresa não somente passou a poder utilizá-lo gratuitamente, mas também novas funcionalidades e melhorias foram implementadas de forma mais rápida.

Atualmente o kubernetes é a principal, mas não a única, solução para **orquestração** de containers.



Introdução á Kubernetes

Orquestração de Containers





Introdução á Kubernetes

Orquestração de Containers

Imagine que você está desenvolvendo uma aplicação que possui alguns containers:

- Nginx com Python;
- PostgreSQL;
- Redis;

Você terminou o desenvolvimento e publicou sua aplicação, com toda a ajuda e facilidade que o docker, e os containers, nos proporcionam.

Sua aplicação é um sucesso e você precisa “escalar” para que os milhares de usuários tenham uma navegação tranquila.

Como você faria isso?



Introdução á Kubernetes

Orquestração de Containers

Você pode gerenciar isso manualmente, fazendo sua aplicação escalar up/down de acordo com as necessidades.

Mas existe uma forma mais inteligente, fazendo uso de um **orquestrador** que faça este processo, e outros, de forma automatizada.

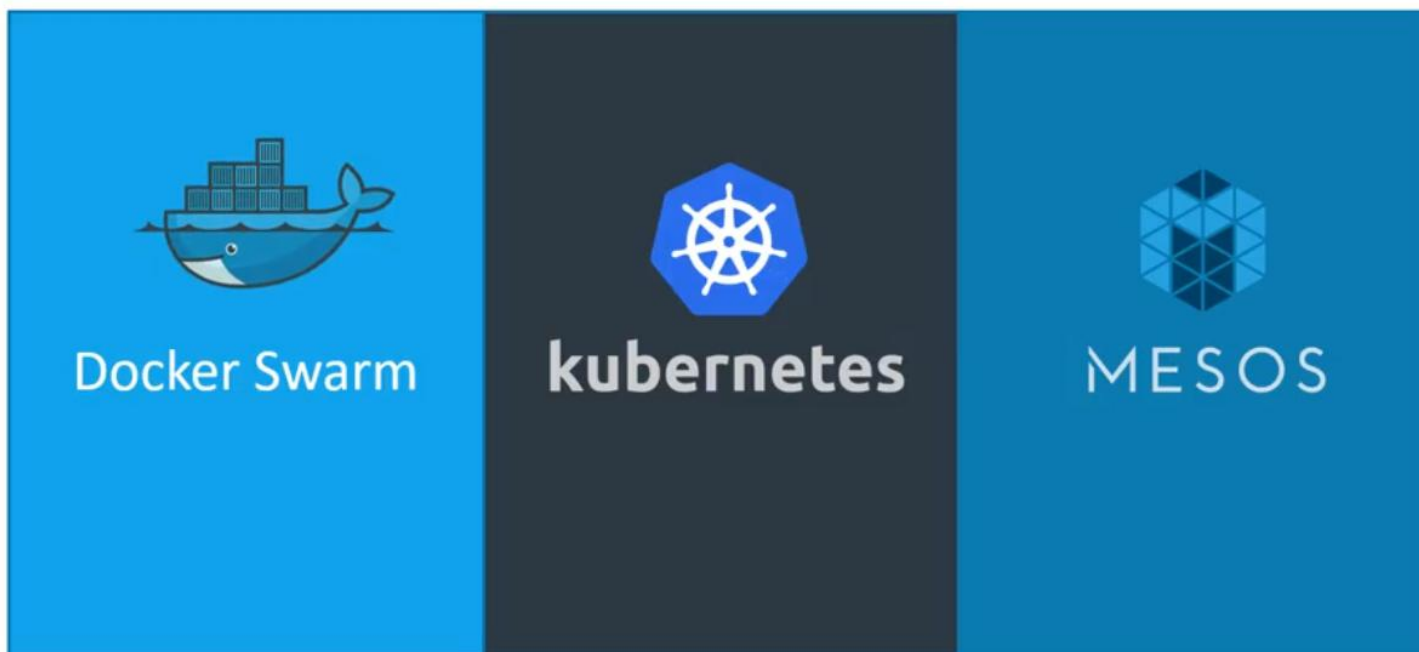


Introdução á Kubernetes

Orquestração de Containers

Você pode gerenciar isso manualmente, fazendo sua aplicação escalar up/down de acordo com as necessidades.

Mas existe uma forma mais inteligente, fazendo uso de um **orquestrador** que faça este processo, e outros, de forma automatizada.





Introdução á Kubernetes

Docker Swarm

Projeto do mesmo criador do Docker, é a forma mais simples e fácil de realizar a orquestração, mas carece de recursos avançados que facilitam o processo de escalamento de uma aplicação.





Introdução á Kubernetes

Mesos

É a ferramenta mais complexa, em questão de configurações, para orquestração de containers.





Introdução á Kubernetes

Kubernetes

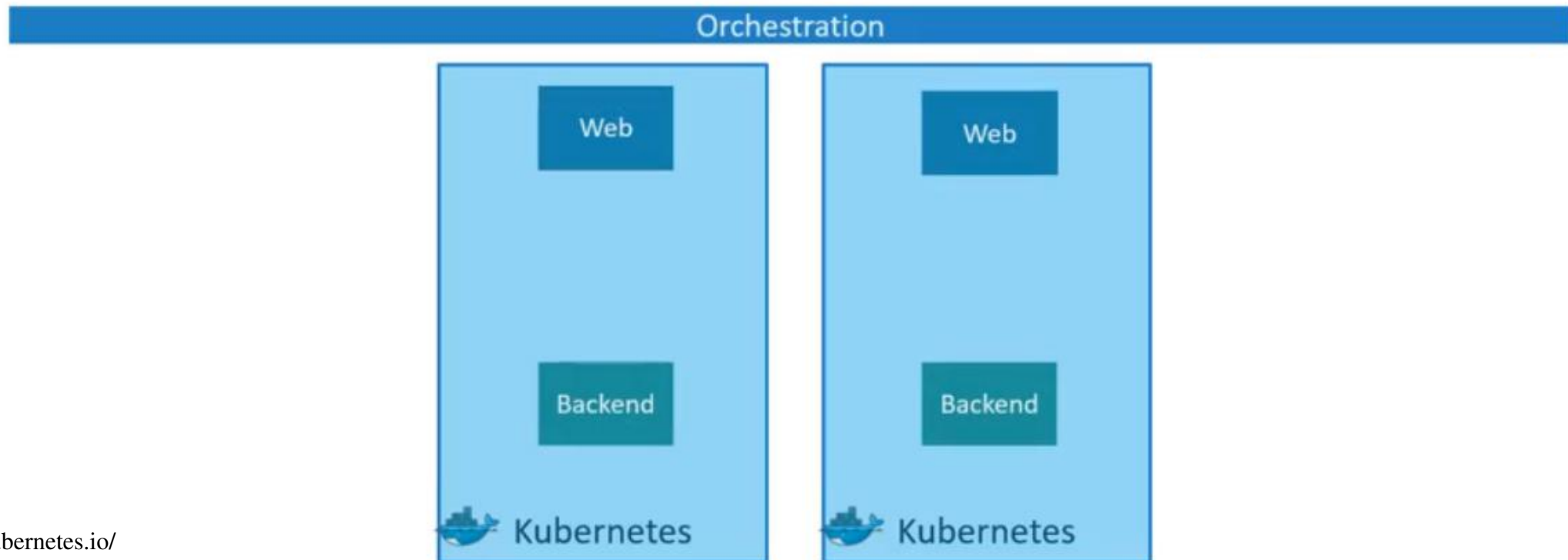
É a ferramenta mais utilizada no mercado, sendo um pouco mais complicado para configurações iniciais, se comparado com o Docker Swarm, mas bem mais simples que o Mesos. E é a ferramenta mais completa e poderosa dentre os 3.





Introdução á Kubernetes

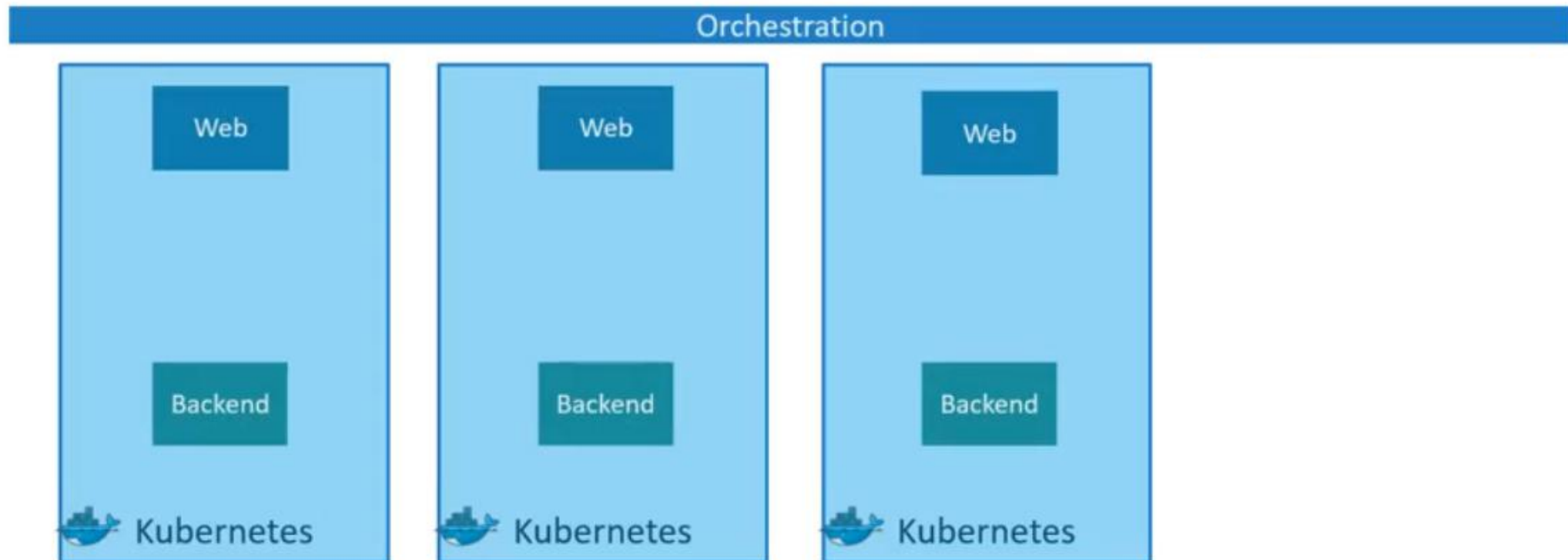
Fazendo uso do Kubernetes, a medida que sua aplicação vai recebendo mais acessos, novos “nodes” são criados, e se a aplicação tiver menos acessos estes “nodes” extras são destruídos, fazendo com que tenhamos mais flexibilidade.





Introdução á Kubernetes

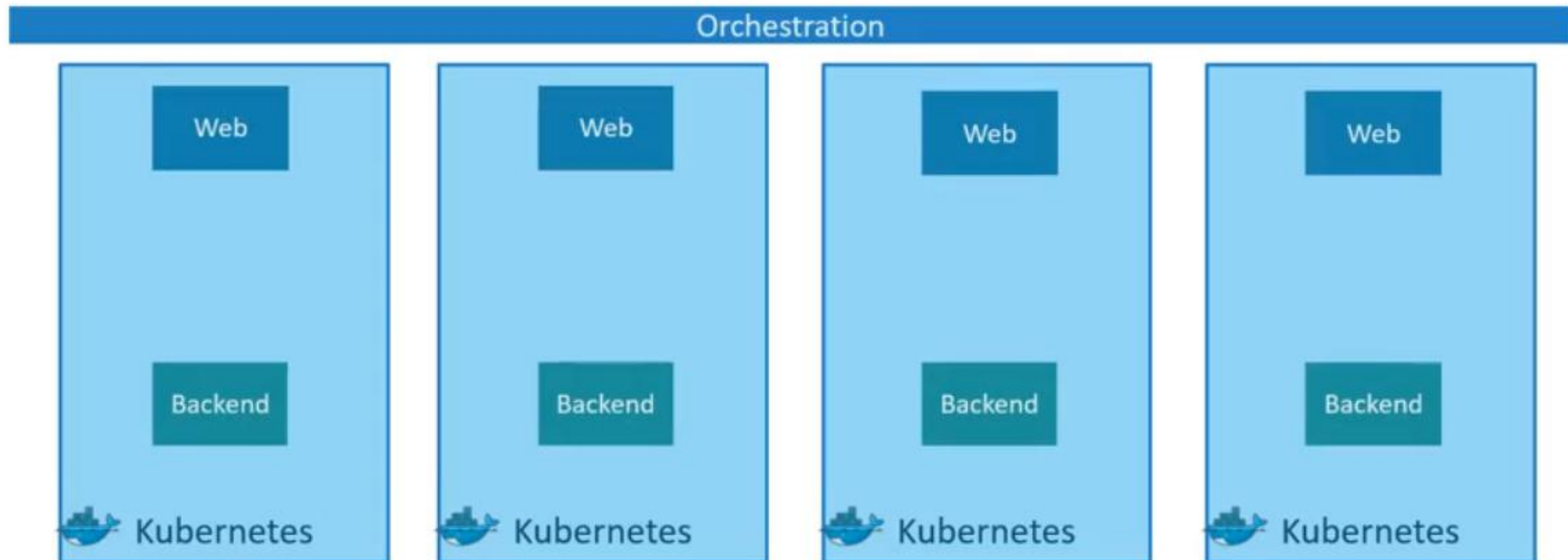
Fazendo uso do Kubernetes, a medida que sua aplicação vai recebendo mais acessos, novos “nodes” são criados, e se a aplicação tiver menos acessos estes “nodes” extras são destruídos, fazendo com que tenhamos mais flexibilidade.





Introdução á Kubernetes

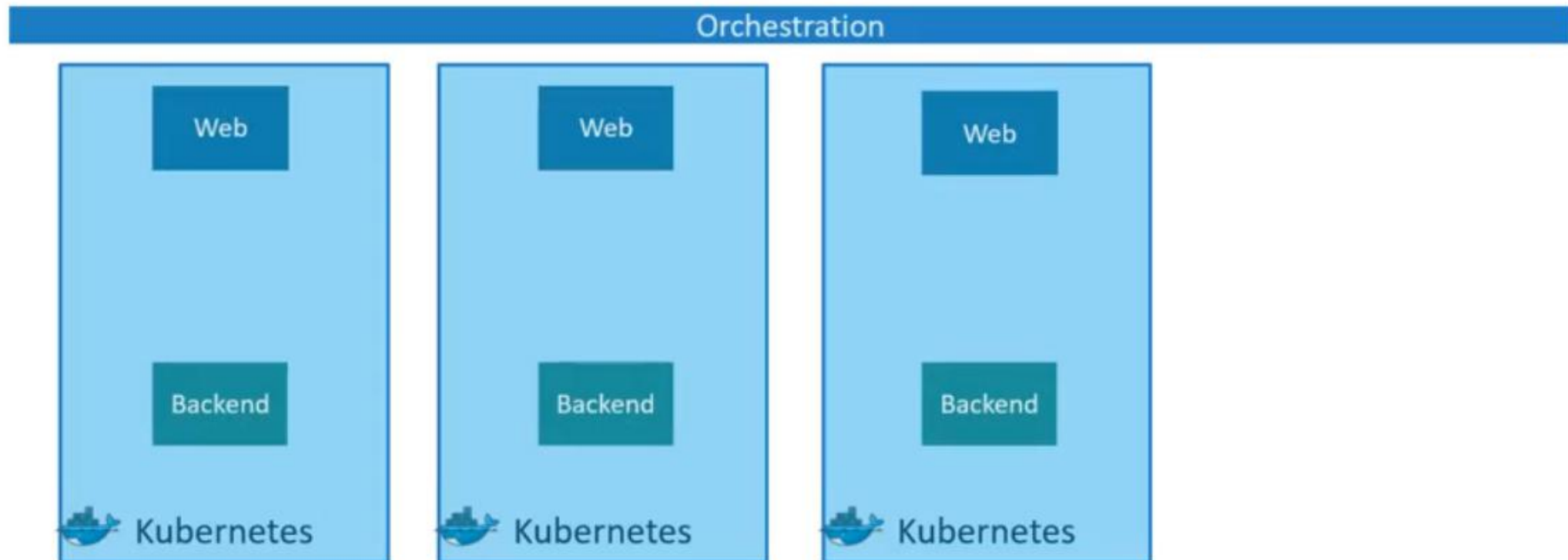
Fazendo uso do Kubernetes, a medida que sua aplicação vai recebendo mais acessos, novos “nodes” são criados, e se a aplicação tiver menos acessos estes “nodes” extras são destruídos, fazendo com que tenhamos mais flexibilidade.





Introdução á Kubernetes

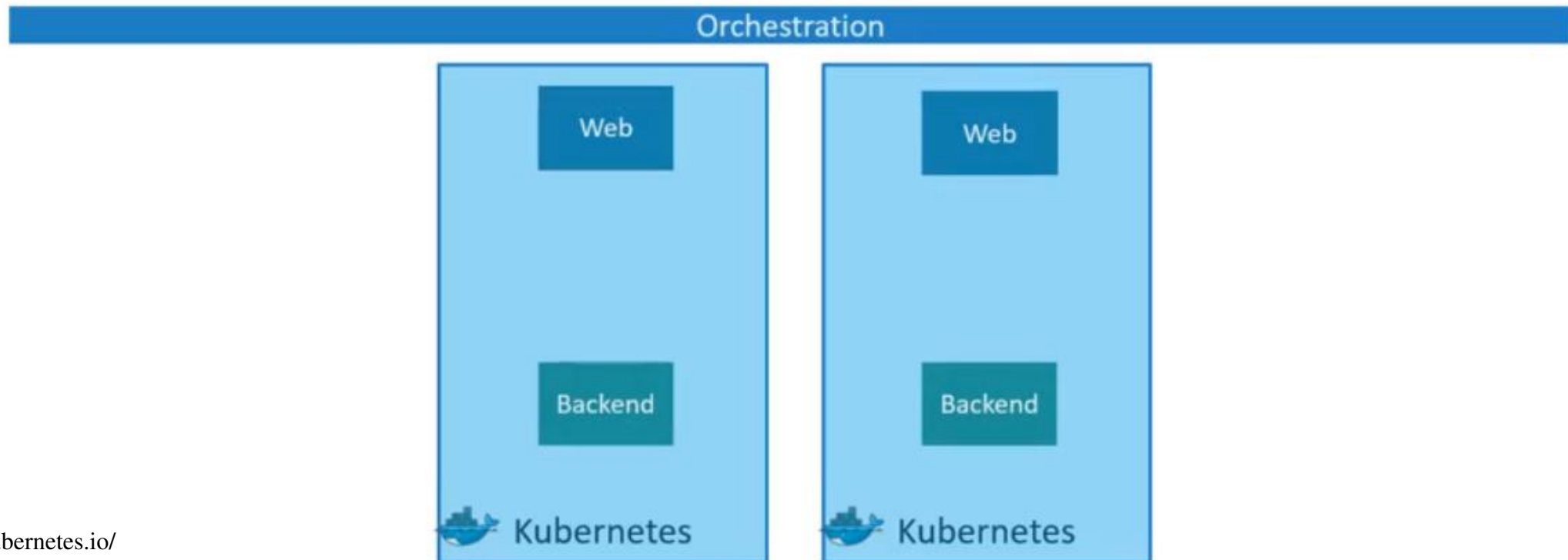
Fazendo uso do Kubernetes, a medida que sua aplicação vai recebendo mais acessos, novos “nodes” são criados, e se a aplicação tiver menos acessos estes “nodes” extras são destruídos, fazendo com que tenhamos mais flexibilidade.





Introdução á Kubernetes

Fazendo uso do Kubernetes, a medida que sua aplicação vai recebendo mais acessos, novos “nodes” são criados, e se a aplicação tiver menos acessos estes “nodes” extras são destruídos, fazendo com que tenhamos mais flexibilidade.



Introdução á Kubernetes

Arquitetura do Kubernetes





Introdução á Kubernetes

Arquitetura do Kubernetes

Para fazer uso do Kubernetes precisamos conhecer sua arquitetura básica, onde temos como componentes principais:

- Node
- Cluster
- Master

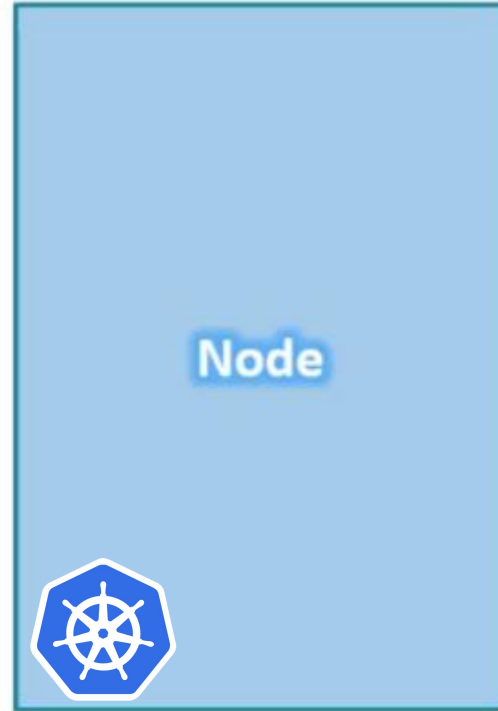
Iremos conhecer agora cada um destes elementos.



Introdução á Kubernetes

Node

Um node é uma máquina, física ou virtual, onde o Kubernetes está instalado.



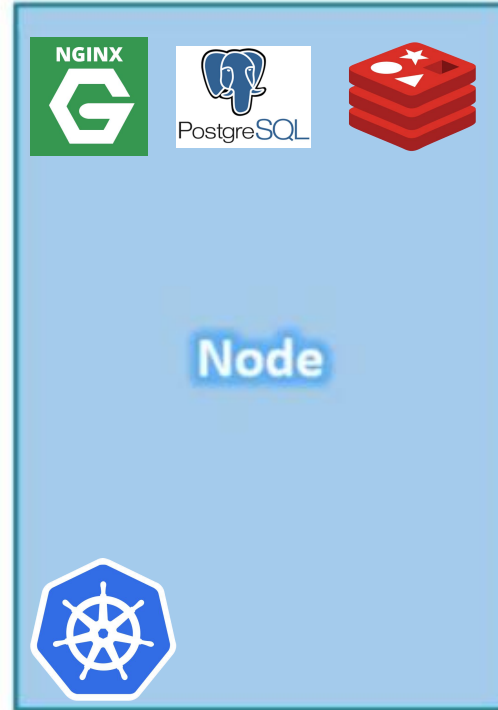


Introdução á Kubernetes

Node

Um node é uma máquina, física ou virtual, onde o Kubernetes está instalado.

No node criamos os containers (com docker) com as nossas aplicações.

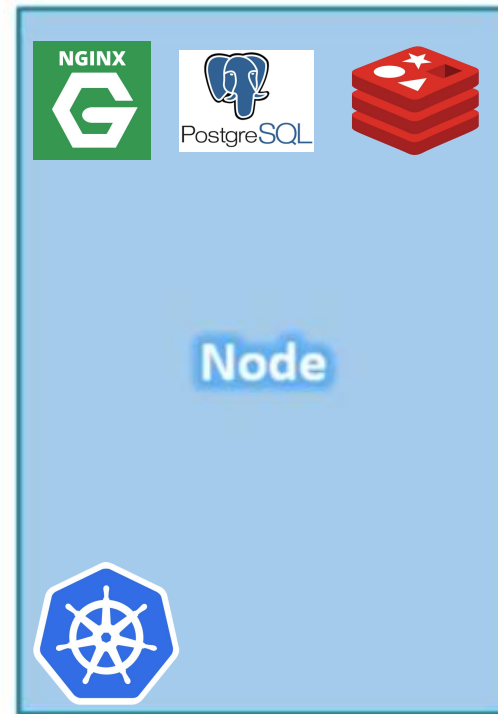




Introdução á Kubernetes

Node

Um node é uma máquina, física ou virtual, onde o Kubernetes está instalado.



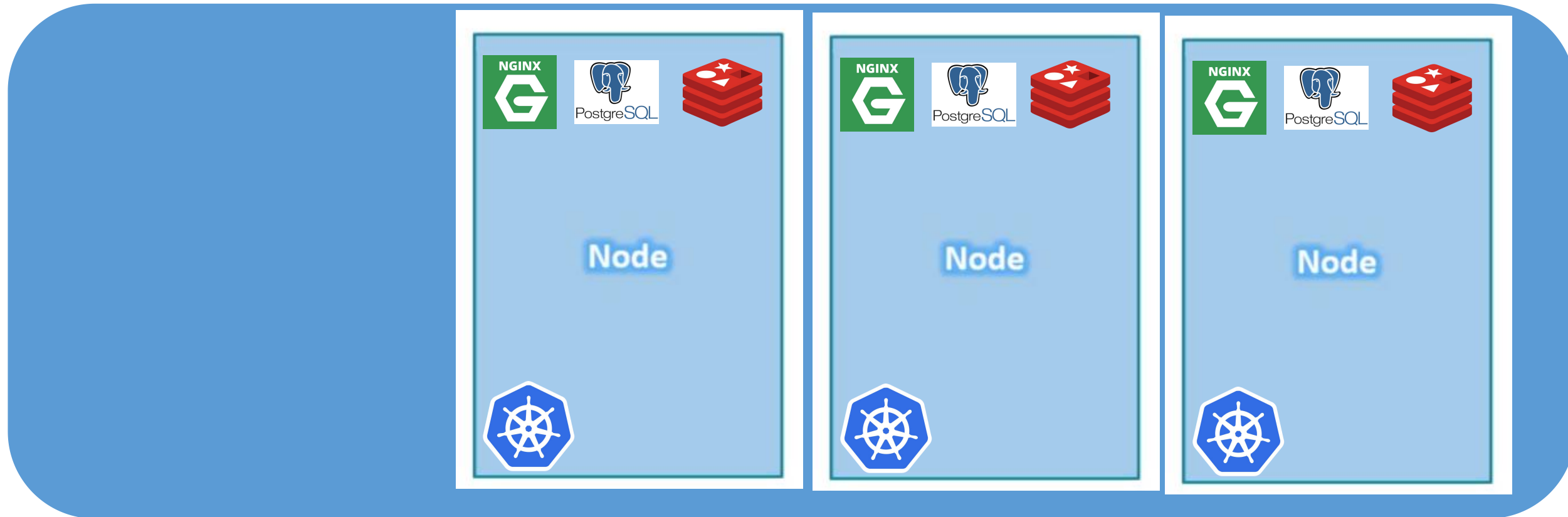
OBS: Nos primórdios do Kubernentes, os Nodes eram chamados de **Minions**.



Introdução á Kubernetes

Cluster

Um cluster é um conjunto de nodes agrupados.

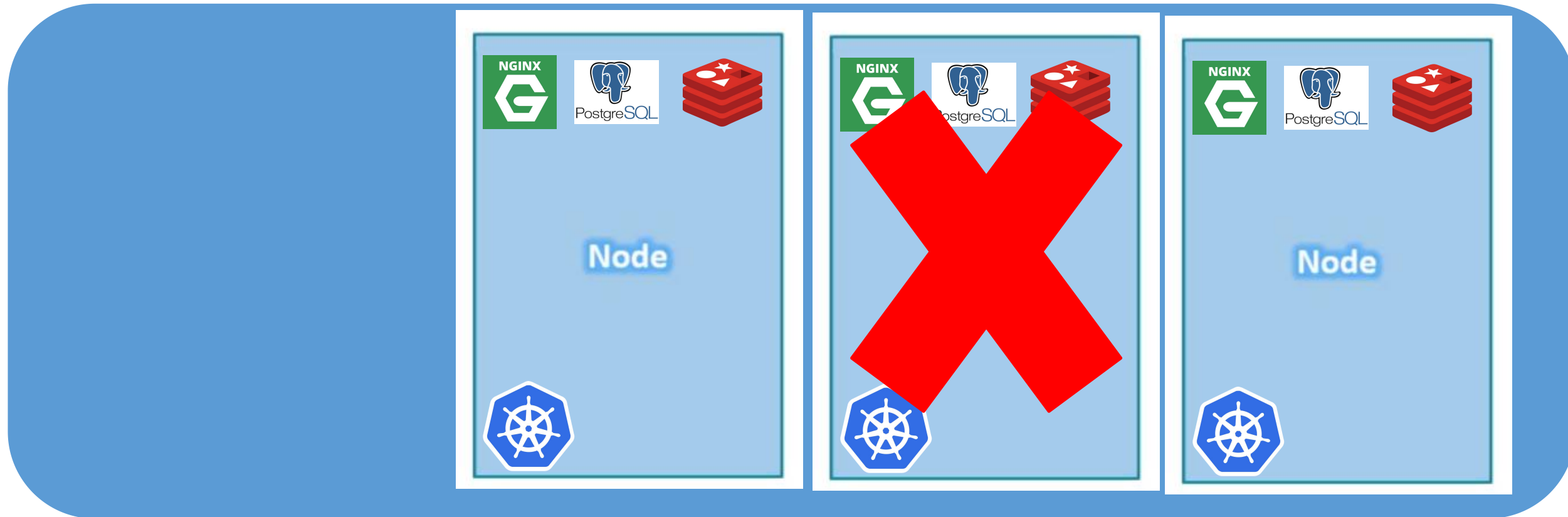




Introdução á Kubernetes

Cluster

Um cluster é um conjunto de nodes agrupados.



Desta forma mesmo que um dos nodes falhe, sua aplicação continua sendo acessada através dos outros nodes.



Introdução á Kubernetes

Master

A partir do momento que temos um cluster, alguém tem que ser responsável:

- Por gerenciá-lo;
- Para manter informações dos membros do cluster;
- Para monitorar os nodes;
- Para fazer algo (ativar um novo node ou não) quando um node falha;

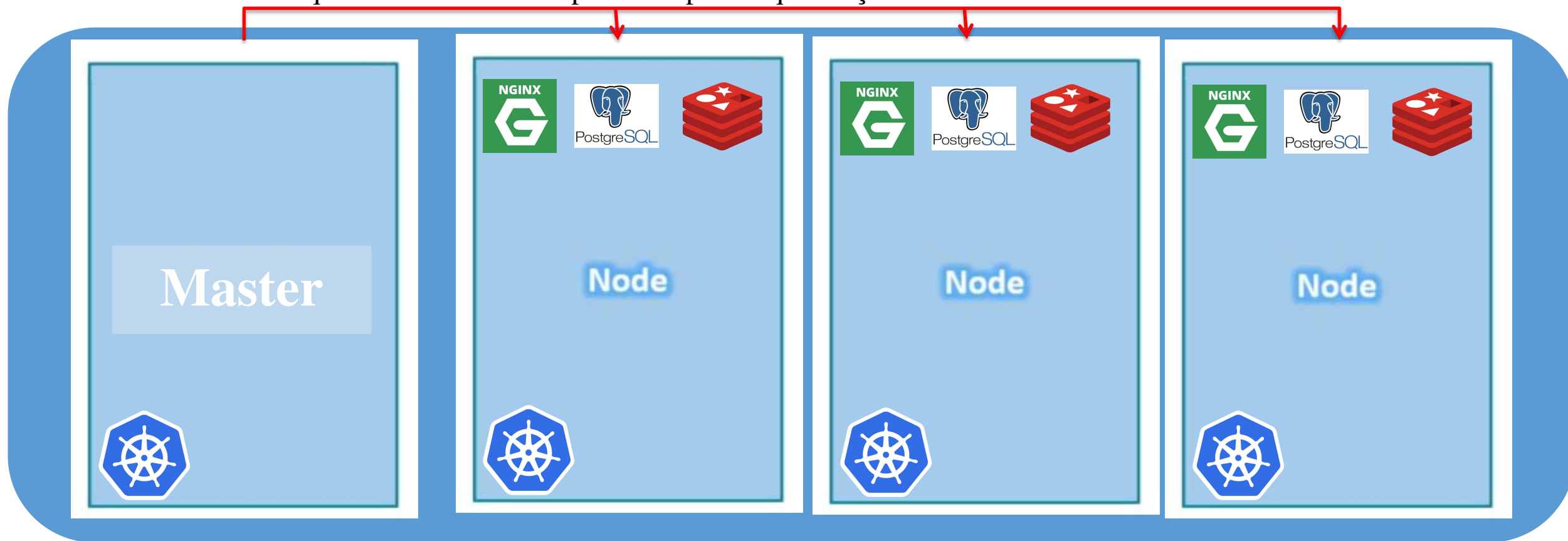
Esta é a função do Master (mestre), **e o mestre não é você.**



Introdução á Kubernetes

Master

O master é um node que é o verdadeiro responsável pela orquestração dos containers.



OBS: Os nodes que não são “master” são chamados também de **Worker Nodes**.

Introdução á Kubernetes

Componentes do Kubernetes





Introdução á Kubernetes

Componentes do Kubernetes

Quando fazemos a instalação do Kubernetes vários componentes são instalados em conjunto

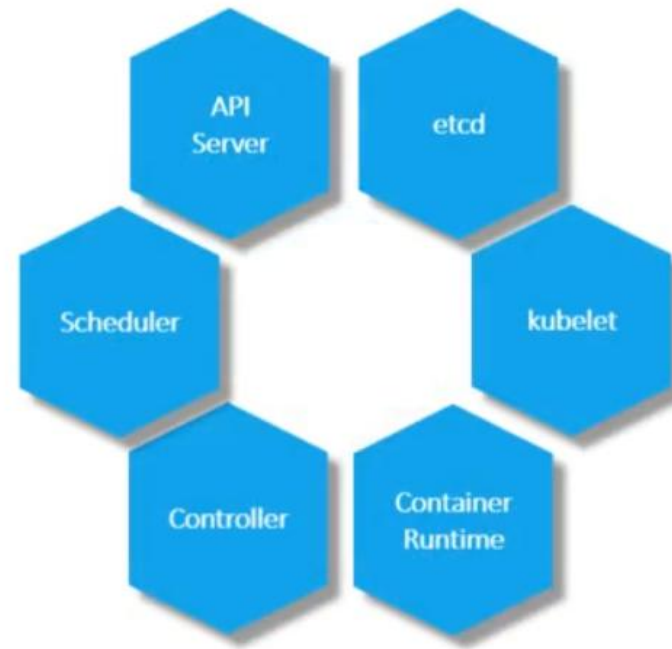




Introdução á Kubernetes

Componentes do Kubernetes

Quando fazemos a instalação do Kubernetes vários componentes são instalados em conjunto





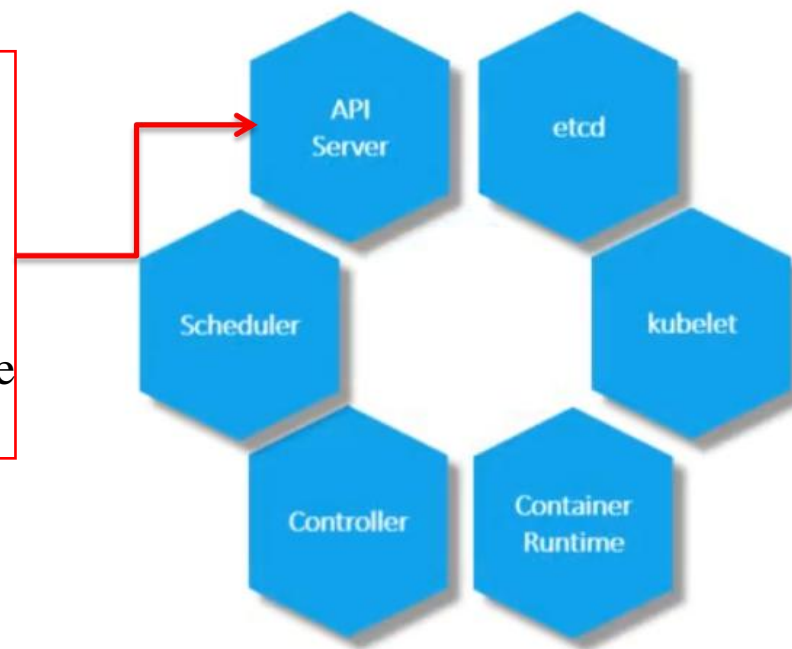
Introdução á Kubernetes

Componentes do Kubernetes

Quando fazemos a instalação do Kubernetes vários componentes são instalados em conjunto

A API Server funciona como um frontend para o Kubernetes.

Ou seja, é através dele que fazemos uso do gerenciamos usuários, dispositivos, interface de linha de comando.

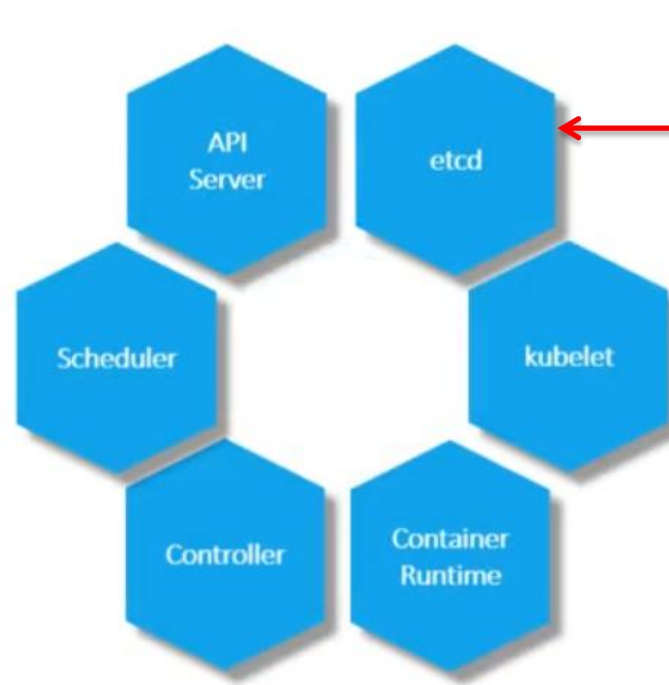




Introdução á Kubernetes

Componentes do Kubernetes

Quando fazemos a instalação do Kubernetes vários componentes são instalados em conjunto



O etcd é usado para armazenar dados, de forma distribuída, no formato chave/valor para configuração e gerenciamento dos clusters.

Sim, podemos ter múltiplos clusters e cada cluster com seu master, e o etcd guardando dados de forma distribuída nos nodes.

Além disso os logs dos clusters são gravados aqui.



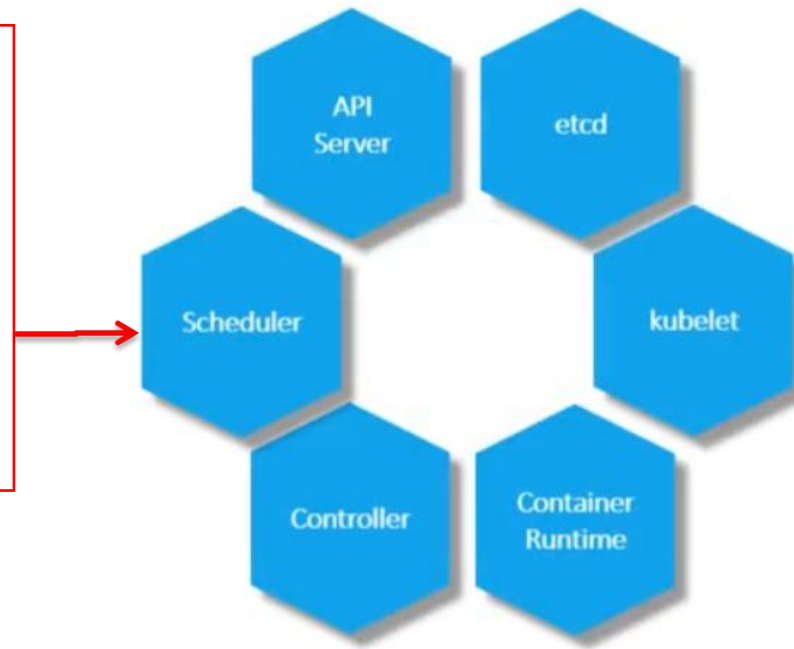
Introdução á Kubernetes

Componentes do Kubernetes

Quando fazemos a instalação do Kubernetes vários componentes são instalados em conjunto

O scheduler (agendador) é responsável por distribuir o trabalho para os containers através dos múltiplos nodes.

Ele também “busca” por novos containers criados e anexa eles aos nodes.





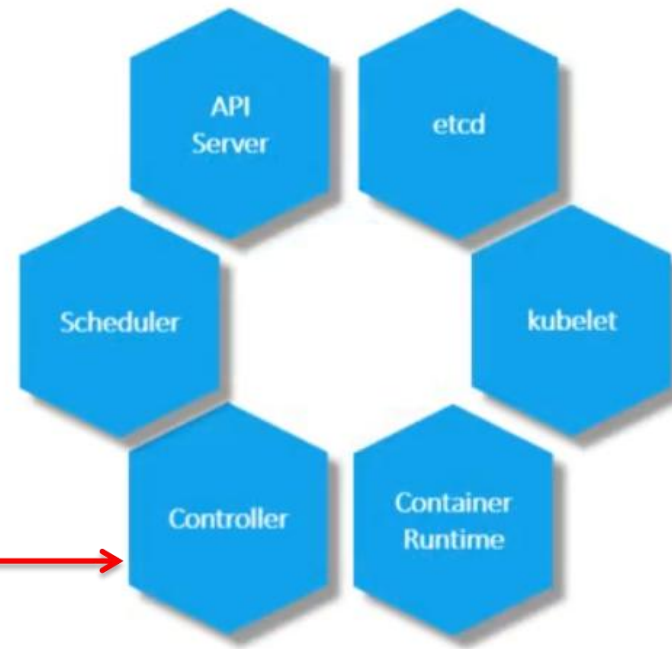
Introdução á Kubernetes

Componentes do Kubernetes

Quando fazemos a instalação do Kubernetes vários componentes são instalados em conjunto

O controller é o cérebro por traz da orquestração.

Ele é responsável por tomar as decisões quando um node falha ou apresenta problemas e pode ativar novos nodes para substituir os problemáticos.

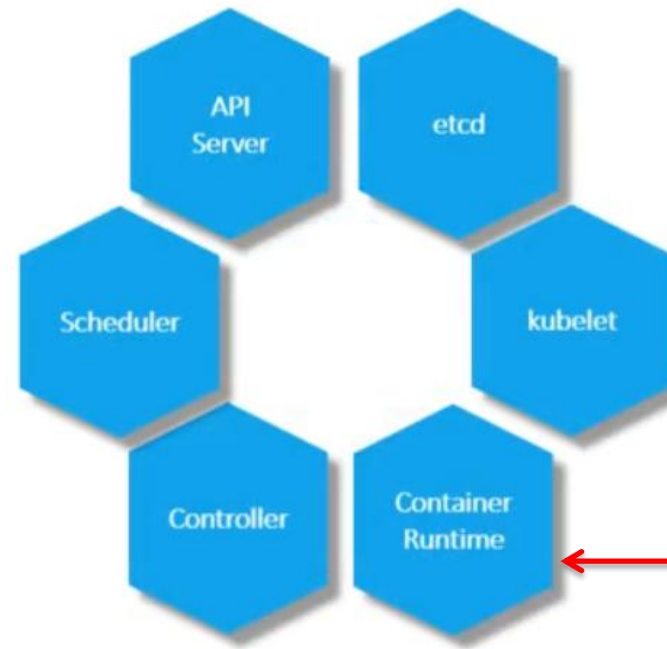




Introdução á Kubernetes

Componentes do Kubernetes

Quando fazemos a instalação do Kubernetes vários componentes são instalados em conjunto



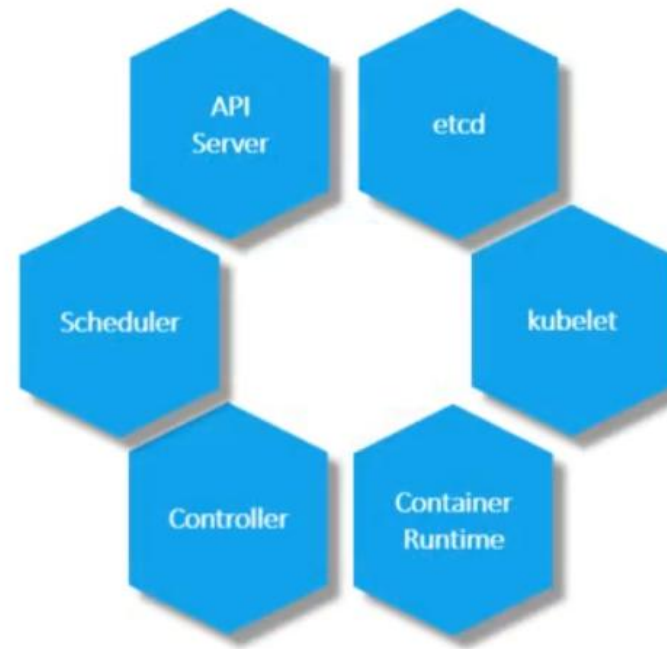
O container runtime faz interface com o software usado para criação dos containers, no nosso caso o docker engine, mas pode ser utilizado com outros softwares com o mesmo propósito.



Introdução á Kubernetes

Componentes do Kubernetes

Quando fazemos a instalação do Kubernetes vários componentes são instalados em conjunto



O kubelet é o agente que é executado em cada node em um cluster.

O agente é responsável por checar se os containers estão sendo executado no node conforme o esperado.



Introdução á Kubernetes

Master versus Worker Nodes

O que difere um master node de um worker node?



Introdução á Kubernetes

Master versus Worker Nodes

O que difere um master node de um worker node?



Note os componentes presentes no node master, enquanto o worker node possui menos componentes.



Introdução á Kubernetes

kubectl

Por fim, ao instalar e configurar o Kubernetes, coisa que fazemos na próxima seção, teremos a disposição um programa de linha de comando chamado kubectl (Kube Control Tool) na qual iremos utilizar, em muitos momentos, para executar comandos do kubernetes.

Uma das utilizações desta ferramenta é para realizar deploy (publicação) de aplicações em um cluster do kubernetes.

Exemplos de comandos:

kubectl run nome-da-aplicacao

kubectl cluster-info

kubectl get nodes



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br