



Kubernetes & Orquestração

Mas não é a mesma coisa?

Agenda

1. Quem sou eu
2. Revisão containers
3. Orquestração
4. Kubernetes
5. Mãos na massa!
6. Considerações finais

Quem sou eu

Diego Ferreira

DevOps Engineer @ Blip.pt

28 anos, engenheiro
eletrônico de formação
pela UFPE e DevOps
Engineer por paixão.



<https://bit.ly/dieodevops>



Revisão Containers

Motivação

Como eu posso garantir
que o meu app vai estar
~~quase~~ **sempre rodando e**
disponível?

Spoiler: usando uma ferramenta para
reiniciar em caso de falhas e outros
problemas inesperados



Containers

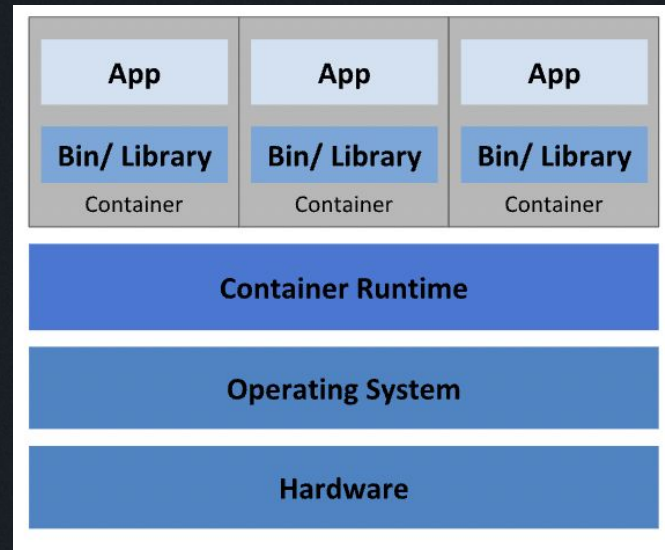
O que são? Onde vivem? De onde vem?
Para que servem?

- São semelhante a máquinas virtuais;
- São criados para um trabalhos específicos, com dependências específicas;
- Os containers “alugam” um espaço no sistema operacional do computador físico e compartilham vários recursos, com um isolamento menor;
- Existem imagens & repositórios para armazenar as imagens.

Containers

Camadas de execução

Container



Containers

Containers \neq Docker

Container Engine

Docker, CRI-O, RKT e LXD

Container Daemon

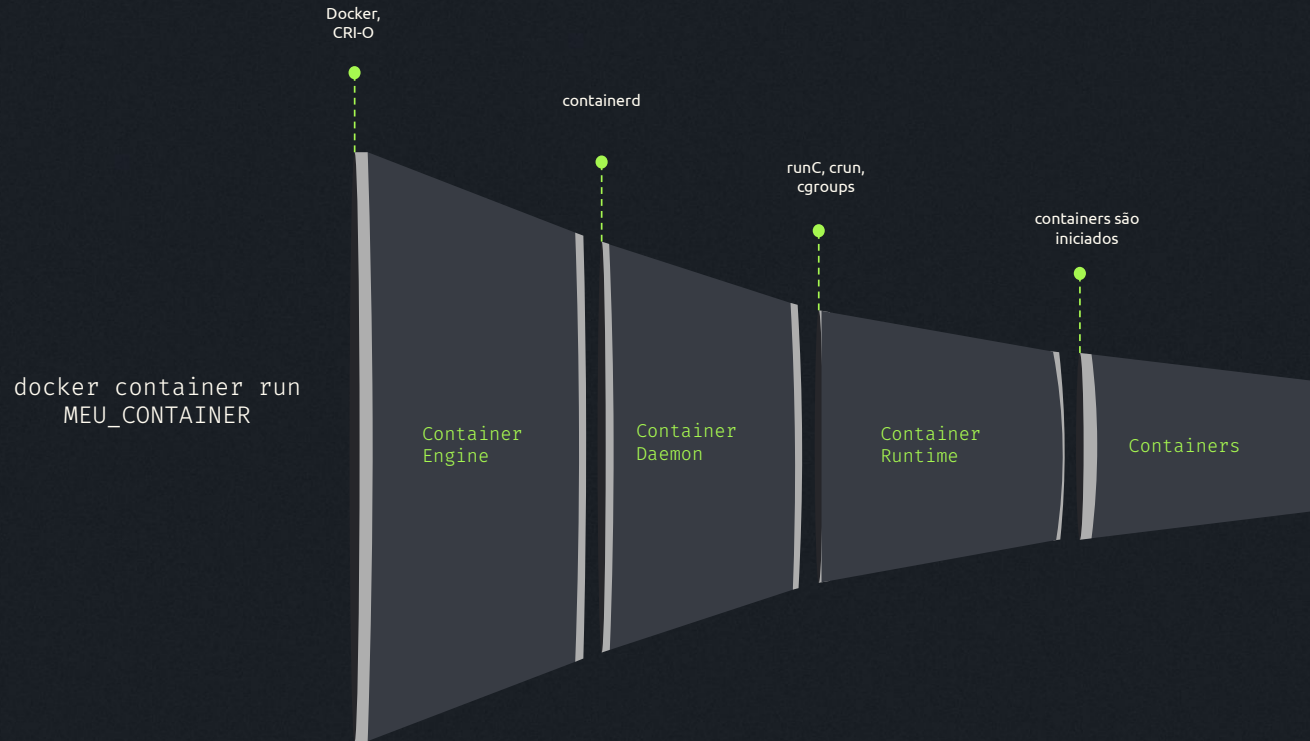
containerd

Container Runtime

runC, cgroups, crun

Containers

Containers \neq Docker





Orquestração

Orquestração

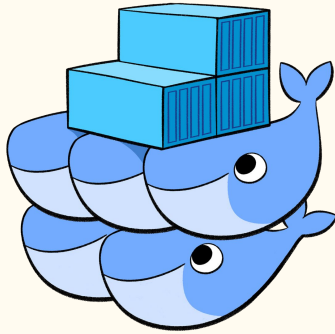
Qual o conceito por trás



- Um maestro é a pessoa responsável por “orquestrar” e gerenciar uma orquestra de músicos, no mundo dos containers é da mesma forma;
- A orquestração de containers é feita com base na inicialização, execução e gerenciamento;
- Um orquestrador de containers faz o papel de gerenciamento e **garante que tudo ocorra como planejado.**

Orquestração

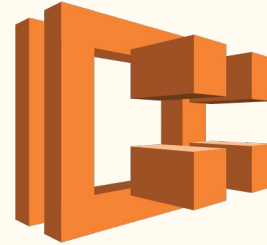
Orquestração \neq Kubernetes



Docker Swarm



Kubernetes

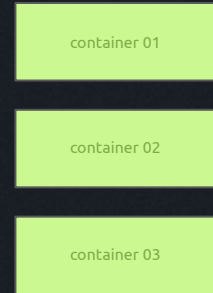
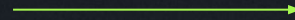
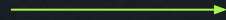


AWS ECS


Orquestração

Como funciona?

Executar o container X,
com a imagem Y em 3
réplicas



- baixa as imagens
- cria os containers
- faz o agendamento
- configura a execução



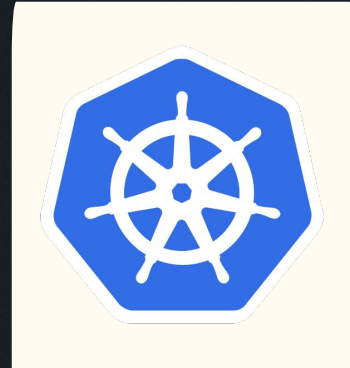
Kubernetes

Kubernetes

O que é? De onde vem? De que se alimenta? Onde vive?

"Kubernetes (K8s) é um produto Open Source utilizado para automatizar a implantação, o dimensionamento e o gerenciamento de aplicativos em contêiner." - kubernetes.io

- Funciona em cluster com o funcionamento de master-slave, em que uma ou mais máquinas são as líderes (ou **control plane**) do cluster e as outras são os slaves (ou **workers**);
- Possui a funcionalidade de criar pods on demand, fazer agendamento, configurar tolerância a falhas, etc.



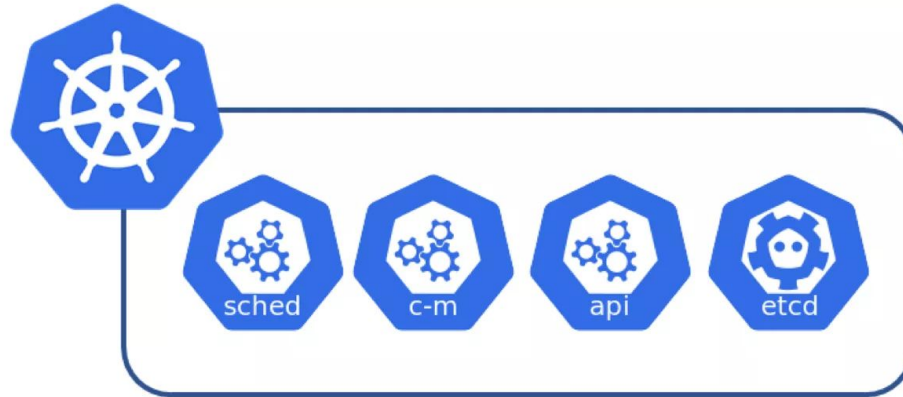
Kubernetes

Como os kubernetes trata os elementos

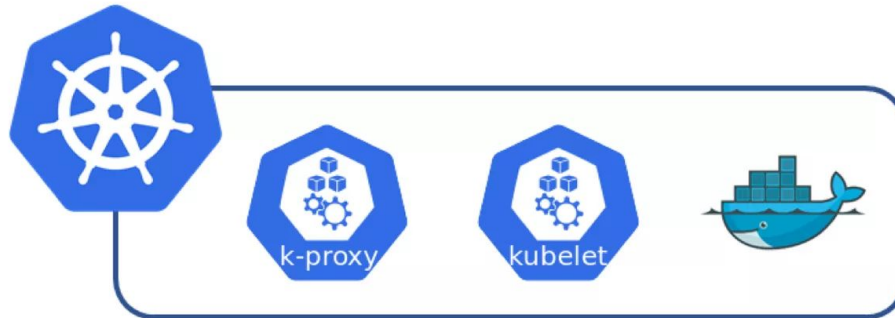
- Os principais elementos constituintes do Kubernetes são:
 - **Pods:** são a unidade mínima responsável por executar os containers;
 - **Deployments:** são as definições de um ou mais pods;
 - **Services:** responsáveis pelas interfaces de rede dos pods;
 - **ConfigMaps:** são objetos lógicos que armazenam informação no formato chave-valor;
 - **Jobs e CronJobs:** elementos que padronizam a execução de pods e fazem agendamentos.

Kubernetes

Master vs Worker



Master Node

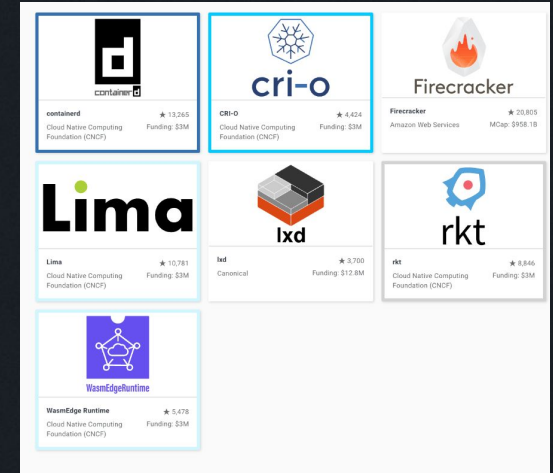


Worker Node

Kubernetes

E aquela confusão do k8s e o docker?

- Não é bem assim... o k8s a partir da versão v1.24 parou de suportar o Docker com container engine;
- Passou a adotar apenas engines que fossem compatíveis com o **Container Runtime Interface (CRI)**;



mais informações: <https://kubernetes.io/blog/2020/12/02/dont-panic-kubernetes-and-docker/>

Kubernetes

Kubernetes por debaixo dos panos -
Master node

Scheduler

É o processo que faz o agendamento dos pods nos nós do cluster.

API Server

É o ponto de contato com a base de dados etcd e serve como uma ponte entre o usuário e a configuração de estado do cluster.

etcd

É uma base de dados do tipo chave valor que armazena todo o estado atual do cluster e é utilizada como fonte de verdade.

Controller Manager

É um processo que roda constantemente e que verifica o estado atual do cluster (consultando no etcd) e aplicando as ações necessárias em caso de divergências.

Kubernetes

Kubernetes por debaixo dos panos -
Worker node

Kubelet

É um agente instalado em cada nó *worker* que é responsável por garantir que os pods sejam devidamente executados;

Kubernetete Proxy

É um serviço que roda em cada nó *worker* que assegura que a comunicação entre todos os elementos ocorra sem problemas. Por exemplo:

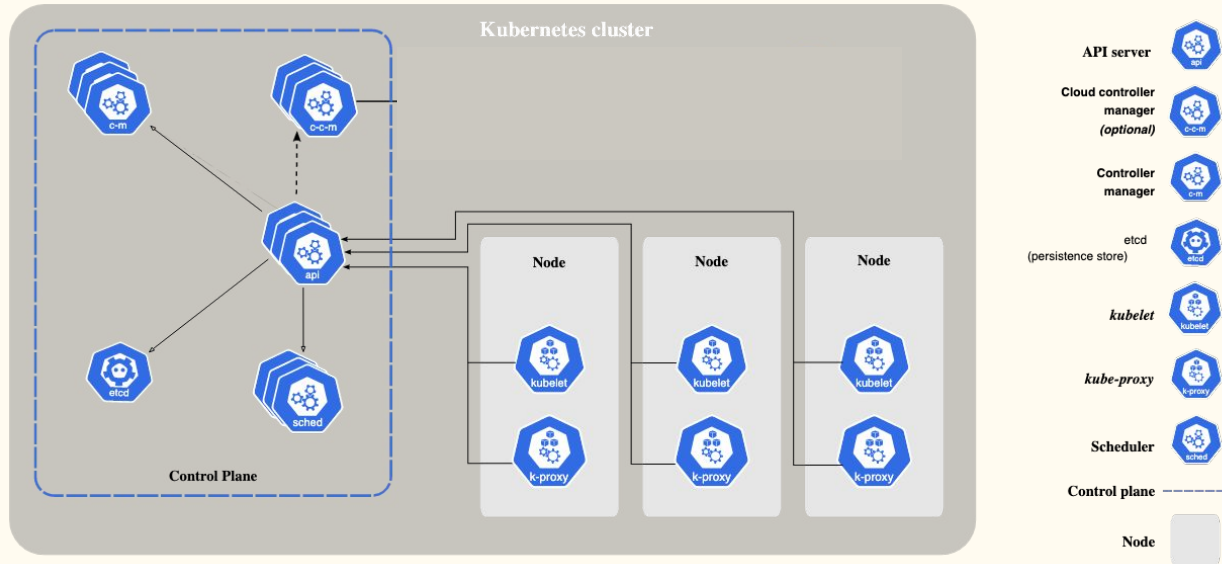
- pod → pod
- node → node

Container Runtime

Aplicação instalada em cada nó *worker* que tem como principal objetivo lidar com a execução e configuração dos pods (containers) que estão sendo executados no nó.

Kubernetes

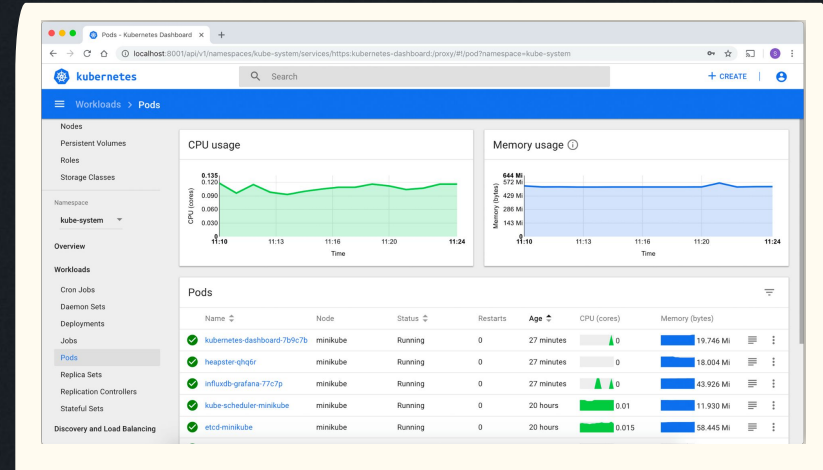
Diagrama Arquitetural do kubernetes



Kubernetes

Outras informações interessantes

- O k8s possui um gerenciador de pacotes próprio, o Helm;
- A configuração dos objetos é feita via arquivo YAML;
- A interação com o cluster pode ser feita via CLI, com o **kubectl**;
- É possível instalar um dashboard para visualizar os elementos de maneira mais visual;
- Existem programas que “simulam” um cluster de k8s para testes locais. (kind, minikube)



Mãos na massa!

99%

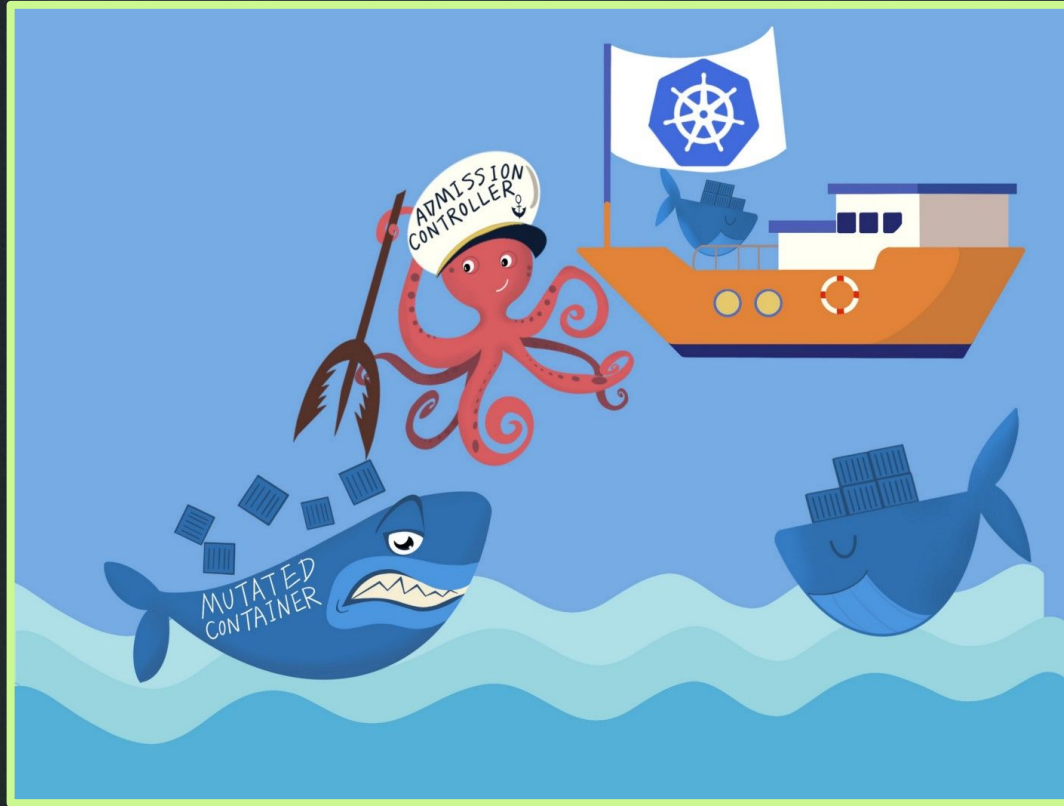


código: <https://github.com/desferreira/meetup-k8s>

Considerações finais

Considerações finais

Q&A



Obrigado!