

## Container

Na pesquisa realizada o exemplo sobre container que achei claro foi no site da microsoft:

*Da mesma forma que empresas de logística usam contêineres físicos para isolar diferentes cargas para transporte em navios e trens, as tecnologias de desenvolvimento de software cada vez mais usam um conceito chamado de "transporte em contêineres"*

*Um único pacote de software, conhecido como um contêiner, agrupa o código de um aplicativo e os respectivos arquivos de configuração e bibliotecas necessários para a execução do aplicativo*

Um container é um ambiente isolado utilizado para empacotar aplicações, têm o objetivo de segregar e facilitar a portabilidade de aplicações em diferentes ambientes.

Um container contém um conjunto de processos que são executados a partir de uma imagem, imagem esta que fornece todos os arquivos necessários. Os containers compartilham o mesmo kernel e isolam os processos da aplicação do restante do sistema operacional.

A ideia é que cada container assuma apenas uma responsabilidade. Nos containers, você divide a responsabilidade isolando os processos de cada aplicação, garantindo assim que nenhum processo possa influenciar no funcionamento dos demais processos.

O Agrupamento de Container é conhecido como cluster: consegue compartilhar recursos como armazenamento, tornando possível a execução de dezenas e até centenas de containers de maneira simultânea a partir do mesmo ambiente.

### Gerenciamento de containers

É preciso organizar, gerenciar e monitorar os containers em execução, esse gerenciamento se chama orquestração. Existem ferramentas de orquestração que fazem o gerenciamento de múltiplos containers. Os mais conhecidos são o OpenShift e o Kubernetes.

Com a orquestração de containers, você consegue automatizar e gerenciar diversas tarefas como o provisionamento e implantação, configuração, alocação de recursos, disponibilidade de containers, escalada, balanceamento, roteamento de tráfego e remoção de containers, de acordo com parâmetros como a demanda de requisições e o consumo de recursos, por exemplo.

O objetivo principal dos orquestradores é cuidar do ciclo de vida dos containers, distribuindo-os conforme as suas especificações ou de acordo com as demandas.

## Dockers

O Docker é uma plataforma open source para construir, armazenar, distribuir e rodar contêineres, foi criado por uma empresa homônima e lançado em 2013.

De desktop a cloud, o Docker é considerado padrão e a ferramenta mais popular: é usado por milhões de desenvolvedores para construir e compartilhar aplicações containerizadas. Para termos uma ideia de volume, 30% das empresas usavam Docker em ambientes AWS, em 2019.

Embora seja uma das ferramentas de containerização mais usadas e conhecidas, o Docker não está sozinho. Há outros fornecedores no ecossistema: ContainerD, CoreOS, Canonical, LXC Linux Containers, CRIO-D e Mesos Containerizer.

## **Kubernetes**

Aplicações de produção abrangem múltiplos containers. Eles devem ser implantados em vários hosts do servidor. A segurança dos containers tem várias camadas e pode ser complexa. É aí que o Kubernetes entra em cena. Ele oferece os recursos de orquestração e gerenciamento necessários para implantar containers em escala para essas cargas de trabalho. Com a orquestração do Kubernetes, é possível criar serviços de aplicações que abrangem múltiplos containers, programar o uso deles no cluster, escalá-los e gerenciar a integridade deles com o passar do tempo. Com o Kubernetes, você toma medidas reais para aprimorar a segurança da TI. Também é necessário integrar o Kubernetes com os serviços de rede, armazenamento, segurança, telemetria e outros para oferecer uma infraestrutura de containers global.