

## **2º Trabalho de C115 – Docker**

Alexânder Augusto Silva Fernandes – 1333 – GEC

Vanessa Swerts Esteves – 1397 – GEC

### **Definição**

O Docker é uma ferramenta para criação de containers, com ela podemos separar toda configuração e lógica de inicialização das nossas aplicações do nosso ambiente, evitando que uma aplicação influencie no comportamento de outra. Os contêineres são isolados uns dos outros e agrupam seus próprios softwares, bibliotecas e arquivos de configuração.

### **Comparação entre Docker e Máquina Virtual**

#### **1. Vantagens:**

Uma Máquina Virtual possibilita que um disco rígido virtual seja criado dentro de um ambiente físico por meio de um software específico e opere independentemente dele. Ter um hardware dentro de outro possibilita rodar sistemas operacionais e todo o tipo de programas. Esta é uma das grandes vantagens de ter uma Máquina Virtual, uma vez que você pode usar as aplicações e os programas que precisar sem depender do sistema operacional que tem no disco físico. Outro benefício deste modelo é que o isolamento de recursos permite que a Máquina Virtual “engane” o software, que não detecta que está ativo porque não está rodando em um sistema físico. Como não há interação entre a atuação física e a virtual, essa dinâmica permite que os recursos originais, que não pertencem à máquina virtual, sejam poupados. Sem ter de criar partições no HD para rodar múltiplos sistemas, o disco rígido físico permanece intacto e você pode testar e criar no ambiente virtual sem se preocupar em esgotar os recursos do seu dispositivo.

Por outro lado, os Containers são tipos de ambiente nos quais é possível agrupar aplicações e seus elementos para melhorar tanto o trabalho da equipe de TI, que consegue trabalhar focada em um ambiente específico de uma aplicação, quanto para o funcionamento do software e do servidor, já que os recursos utilizados por cada Container são totalmente isolados. Uma das principais vantagens do Container é a possibilidade de criar serviços e códigos independentes, que podem ser movidos sem dificuldade entre máquinas e ambientes diferentes sem a perda de dados. Além disso, sua operação tende a ser mais rápida, uma vez que só as aplicações precisam ser iniciadas, e não todo o sistema operacional.

Outra vantagem, é que um Container pode ser implantado de forma ágil, além de fornecer uma infraestrutura imutável. Ou seja, raramente o que for configurado no container sofrerá alteração, e caso isso ocorra, ele pode ser facilmente reiniciado. No

que diz respeito ao processo de correção, eles superam o processo tradicional, pois possibilitam que as organizações diminuam o tempo de resposta às falhas. Com isto, eleva consideravelmente a qualidade dos aplicativos, bem como sua manutenção.

## **2. Desvantagens**

Uma desvantagem do Docker se encontra no fato de que, como este divide recursos entre outros containers e outros processos do sistema que o executa, uma aplicação que necessite por exemplo de um alto consumo de CPU terá seu desempenho prejudicado, comparado a executá-la em uma Máquina Virtual ou uma máquina comum. Além disso, a comunicação entre o container e seu host, e o mapeamento de rede necessário para enviar os pacotes aos seus destinos corretos também impacta na performance de seus processos.

Outra desvantagem é a dificuldade na persistência dos dados, pois containers são elaborados para apagar totalmente seus arquivos quando são desligados, portanto qualquer armazenamento de dados para uso futuro teria que ser feito em algum outro lugar. Além disso, o Docker foi criado com o intuito de executar aplicações de servidor que não necessitam de interfaces gráficas, portanto uma aplicação que precisa ser visualmente utilizada não seria própria para ser usada com a plataforma e quando consideramos a segurança, o Docker é um pouco menos robusto que a Máquina Virtual, que ainda garante mais proteção ao usuário.

## **3. Uso de hardware**

Uma mesma máquina física pode contar com diferentes Máquinas Virtuais(MVs). Cada uma opera um Sistema Operacional (SO) exclusivo e possui seu próprio kernel, binários, aplicativos e bibliotecas, o que significa que ocupam um espaço grande no servidor. Os containers se localizam em cima de um servidor físico e do SO hospedeiro. Cada um deles compartilha o kernel do SO host e costuma partilhar também as bibliotecas e binários. A diferença é que os itens compartilhados servem somente para leitura, o que torna o container muito leve, especialmente se comparado à MV. Para ter uma ideia, o tamanho do primeiro é de Mb e, por isso, ele é iniciado em poucos segundos. Já a máquina virtual contém vários Gb e demora minutos para ser executada.

### **O que pode ser “dockerizado”?**

Podemos dizer que todas as aplicações podem se beneficiar do Docker, ainda mais se essas aplicações necessitarem de recursos que dependem de instalações pelo Sistema Operacional, como bancos de dados, arquivos de configuração, APIs, etc. Com o Docker, todas essas configurações de software são isoladas em containers, aumentando a flexibilidade de uma aplicação, pois com esses containers, podemos criar, implantar, copiar e migrar de um ambiente para outro facilmente.

