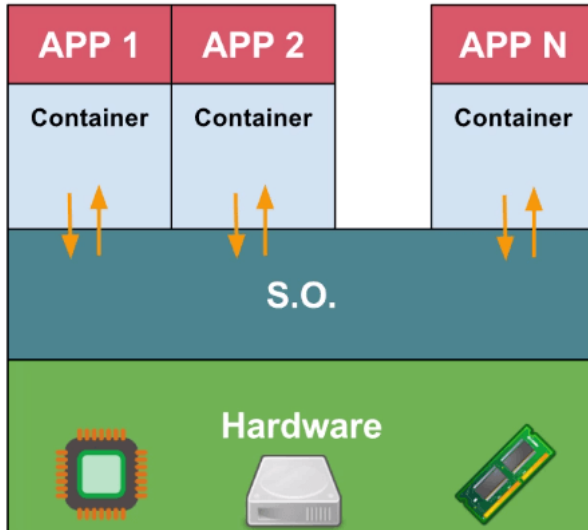


## A era dos containers

### Transcrição

Um **container** funcionará junto do nosso sistema operacional base, e conterá a nossa aplicação, ou seja, a aplicação será executada dentro dele. Criamos um **container** para cada aplicação, e esses **containers** vão **dividir** as funcionalidades do sistema operacional:



Não temos mais um sistema operacional para cada aplicação, já que agora as aplicações estão dividindo o mesmo sistema operacional, que está em cima do nosso hardware. Os próprios **containers** terão a lógica que se encarregará dessa divisão.

Assim, com somente um sistema operacional, reduzimos os custos de manutenção e de infraestrutura como um todo.

### Vantagens de um container

Por não ter que possuir um sistema operacional, o **container** é muito mais leve e não possui o custo de manter múltiplos sistemas operacionais, já que só teremos um sistema operacional, que será dividido entre os **containers**.

Além disso, por ser mais leve, o **container** é muito rápido de subir, subindo em questão de segundos. Logo, o **container** é uma solução para suprir o problema de múltiplas máquinas virtuais em um hardware físico, já que com o **container**, nós dividimos o sistema operacional entre as nossas aplicações.

### Necessidade do uso de containers

Mas por que precisamos dos **containers**, não podemos simplesmente instalar as aplicações no nosso próprio sistema operacional? Até por que já fazemos isso, já que no nosso sistema operacional temos um editor de texto, terminal, navegador, etc.

No caso das nossas aplicações, essa abordagem pode ter alguns problemas. Por exemplo, se dois aplicativos precisarem utilizar a mesma porta de rede? Precisaremos de algo para isolar uma aplicação da outra. E se uma aplicação consumir toda a CPU, a ponto de prejudicar o funcionamento das outras aplicações? Isso acontece se não limitarmos a aplicação. Outro problema que pode ocorrer é cada aplicação precisar de uma versão específica de uma linguagem, por exemplo, uma

aplicação precisa do Java 7 e outra do Java 8. Além disso, uma aplicação pode acabar congelando todo o sistema. Por isso é bom ter essa separação das aplicações, isolar uma da outra, e isso pode ser feito com os **containers**.

Com os *containers*, conseguimos limitar o consumo de CPU das aplicações, melhorando o controle sobre o uso de cada recurso do nosso sistema (CPU, rede, etc). Também temos uma facilidade maior em trabalhar com versões específicas de linguagens/bibliotecas, além de ter uma agilidade maior na hora de criar e subir *containers*, já que eles são mais leves que as máquinas virtuais.