

# Containers com *Docker*

Professor: José Eurípedes Ferreira de Jesus Filho  
*jeferreirajf@gmail.com*

Universidade Federal de Jataí – UFJ

# Aula anterior

- Copiar arquivos do computador para o container

- `docker cp <source> <destiny>`

- Volumes

- *Bind mount*

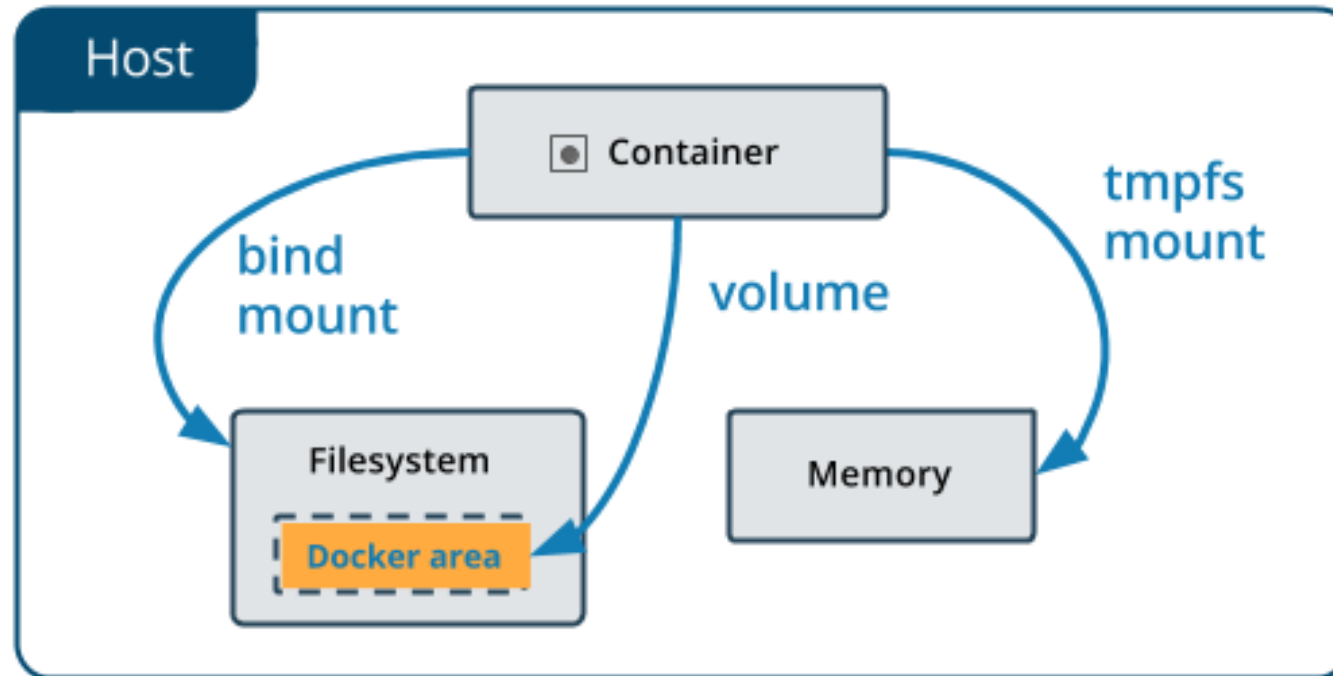
- `docker run --mount type=<type>,source=<source>,target=<target> <image>`

- Volume

- `docker run --volume <source>:<target> <image>`

**Qual a diferença?**

# Volumes no Docker



# Introdução

- Frequentemente disponibilizamos **serviços** em um container que gostaríamos que fosse **acessado externamente**.
  - Esse acesso poderia ser feito partindo de um outro container.
  - Esse acesso poderia ser feito partindo do nosso computador.
- Como liberar portas de acesso?

# Liberar porta de acesso

- Mundo interno:
  - As imagens já possuem definido as portas que elas irão expor ao mundo interno do Docker

```
docker run -itd --rm --name nginx nginx:1.25.1
```

- O Nginx por exemplo expõe internamente a porta 80.

# Liberar porta de acesso

- Mundo externo:

```
docker run -itd --rm --name nginx -p 8000:80 nginx:1.25.1
```

- Dessa forma estamos mapeando a **porta 80 do container** na porta **8000 do nosso computador**.

**Hora da prática!**