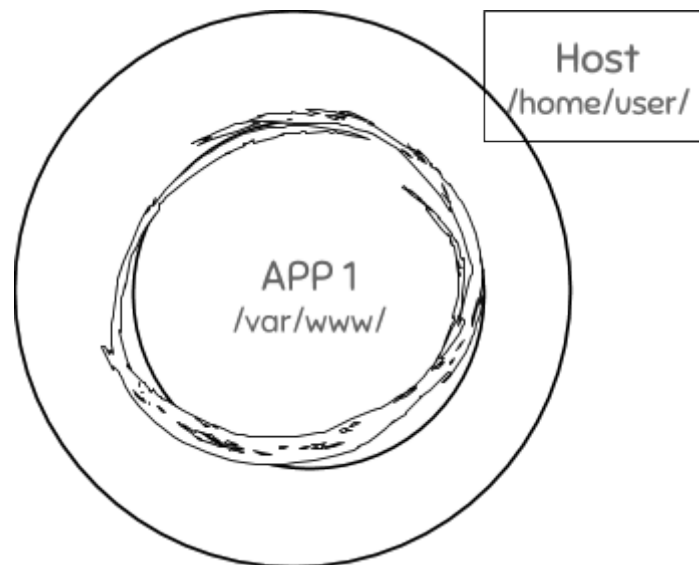


Seção 7 - Volumes Docker

Volumes

- Permite manter os dados, mesmo quando o container é deletado;
- É mapeado num diretório do host, em outro container ou em um volume;
- Volumes são mapeados quando um container é criado;

1. Volume entre o host e o container - “Host directory as a Data Volume”

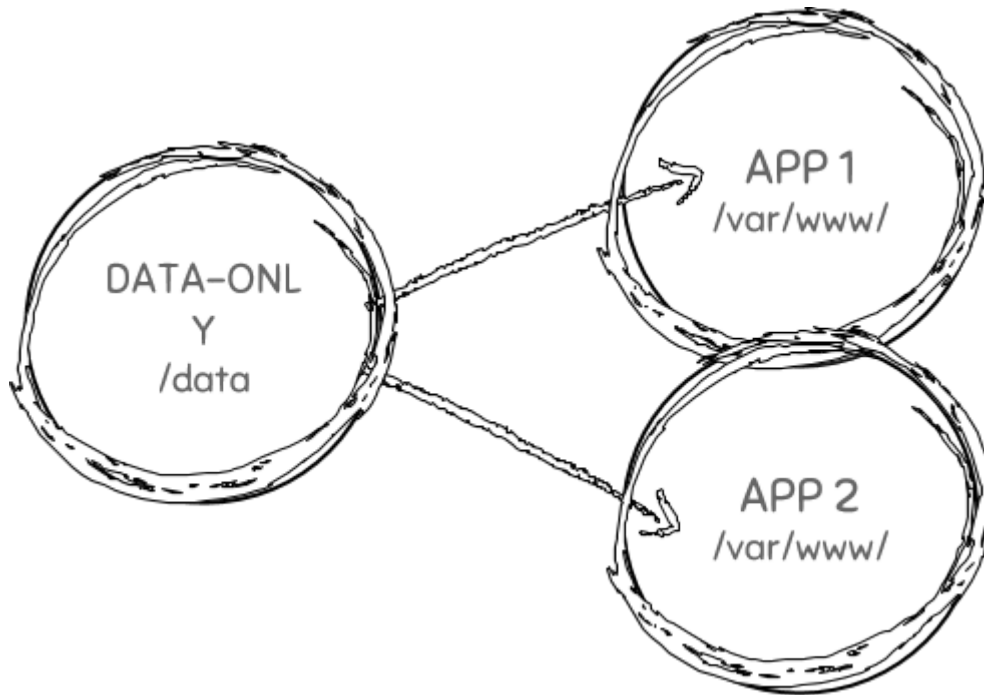


- + Volume salvo no host
- Não escalável

```
# docker run -d --name web_pasta -v
/home/ricardo/site:/usr/local/apache2/htdocs -p 80:80 httpd
```



2. Container como Volume - "Data-only container"



- + Portátil - não atrelado ao host
- Volátil - container se removido ou se falhar

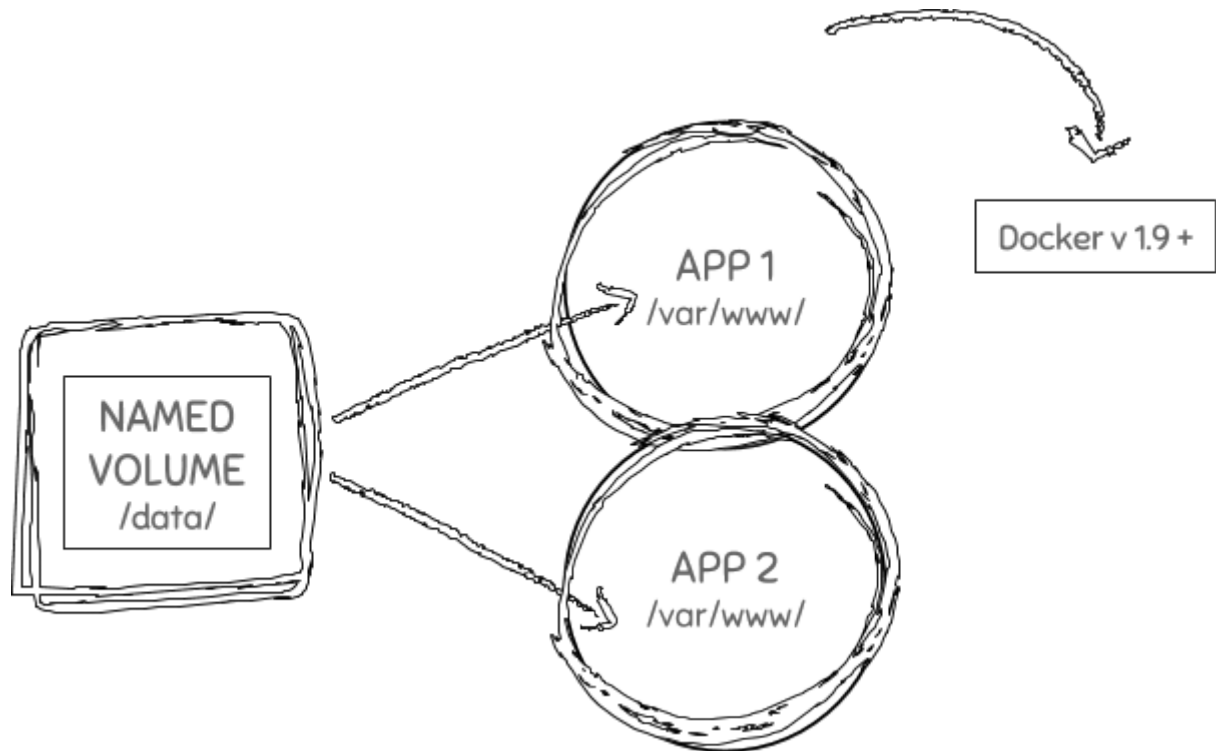
```
# docker create -v /usr/local/apache2/htdocs --name
datacontainer ubuntu
```

```
# docker run -d --volumes-from datacontainer -p 80:80 httpd
```

```
# docker rm -v datacontainer
```



3. Docker Volume - "Shared-storage Volume"



- + Compartilhável - 'plugin' local, NFS, CIFS, Cluster, etc
- + Maior segurança dos dados
- Maior 'overhead' de disco

```
# docker volume create --name datastore
```

```
# docker run -v datastore:/tmp debian
```

```
# docker volume ls
```



“Persistindo os dados de uma aplicação”

Criar uma pasta dentro de ~/docker chamada site:

```
mkdir ~/docker/site
```

** Você deve armazenar todos os arquivos do seu site nesta pasta!*

Criar um arquivo básico index.html de exemplo:

```
<h1>Minha primeira página web </h1>
```

Utilizar o primeiro método de mapeamento de volumes Docker (método "host volume"):

```
sudo docker run -d --name web_pasta -v ~/docker/site:/usr/local/apache2/htdocs -P  
httpd
```

```
sudo docker port web_pasta
```

** Observe que o 'docker port' é utilizado para descobrir qual porta foi liberada no host automaticamente; já que foi usado o parâmetro '-P'*

Acessar o navegador web no endereço [http://localhost:\[PORTA\]](http://localhost:[PORTA]):

** Observe que o arquivo index.html será carregado. Fique a vontade para colocar quaisquer arquivos html, css ou js na pasta ~/docker/site. Depois, veja o resultado atualizando a página!*

Criar um container somente volume, para poder usar o segundo método de mapeamento de volumes Docker ("data-only container"):

```
sudo docker create -v /usr/local/apache2/htdocs --name datacontainer ubuntu
```

** Observe que ao invés do 'docker run' foi usado o 'docker create'. A diferença básica é que o 'run' cria e executa um container. Enquanto o 'create' apenas cria e mantém parado o container.*

Criar container da aplicação a partir desse contaneir de dados:

```
sudo docker run -d --name web_dataonly --volumes-from datacontainer -p 8085:80  
httpd
```



Copiar arquivo index.html para dentro do container de dados usando o 'docker cp':

```
sudo docker cp ~/docker/site/index.html web_dataonly:/usr/local/apache2/htdocs
```

** Depois, prossiga para o webbrowser do seu computador e acesse <http://localhost:8085> e veja o arquivo index.html sendo carregado*

08 - Remover o container somente volume:

```
sudo docker stop web_dataonly
sudo docker rm web_dataonly
sudo docker rm -v datacontainer
```

** Observe que primeiro deve ser parado e removido o container de aplicação que usa esse volume. Depois, assim, remover o container de dados com o parâmetro '-v'*

Para usar Docker Volume crie seu primeiro volume:

```
sudo docker volume create --name datastore
```

Crie um docker de aplicação que use esse volume:

```
sudo docker run --name containervolume -v datastore:/tmp debian
sudo docker exec containervolume touch /tmp/teste
```

** Observe que foi criado um arquivo vazio chamado teste na pasta teste usando o 'docker exec' justamente na pasta (/tmp) onde o volume está montado. Assim, se perder ou remover esse container, o conteúdo dessa pasta será mantida até a remoção do volume.*

Listar e remover volume:

```
sudo docker volume ls
sudo docker stop containervolume
sudo docker rm containervolume
sudo docker volume rm datastore
```

** Observe que primeiro deve ser parado e removido o container de aplicação que usa esse volume. Depois, assim, remover o volume docker*



Referências

- ▶ <https://docs.docker.com/engine/admin/volumes/volumes/>
- ▶ https://docs.docker.com/engine/reference/commandline/volume_create/

