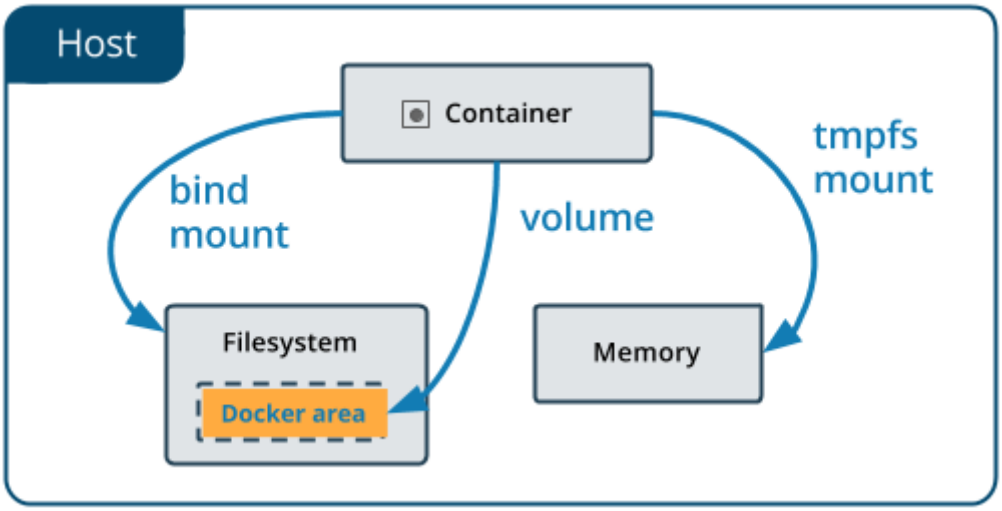
docker volumes



- bind mount: montaje clásico con `-v` sobre una estructura de directorios
- tmpfs mount: almacenar data no persistente
- volume:
  - gestionado desde cli docker
  - trasversal al OS

--volume o --mount

- desde la versión 17.06 `–mount` está también disponible en «standalone» contenedores (además de swarm)

cli

docker volume

- `create`
- `ls`
- `inspect`
- `rm`

docker run

- `–volume | -v`
  - `[ volume_name | local_path | <nothing-for-anonymous-volume ] : path-in-container [ : [ ro ] ]`
- `–mount | -m`
  - `type=[bind,volume,tmpfs], [ source|src= {name,path} ], destination|dst|target=<path-in-container>[,readonly][,volume-opt=<options]`

docker create

examples

```
docker run -d \
  --name devtest \
  --mount source=myvol2,target=/app \ # -v myvol2:/app \
  nginx:latest
```

```
docker run -d \
  --name=nginxtest \
  --mount source=nginx-vol,destination=/usr/share/nginx/html \ #  -v nginx-vol:/usr/share/nginx/html \
  nginx:latest
```

```
$ docker run -d \
  --name=nginxtest \
  --mount source=nginx-vol,destination=/usr/share/nginx/html,readonly \ # -v nginx-vol:/usr/share/nginx/html:ro \
  nginx:latest
```

volume drivers

vieux/sshfs

```
docker volume create --driver vieux/sshfs \
  -o sshcmd=test@node2:/home/test \
  -o password=testpassword \
  sshvolume

docker run -d \
  --name sshfs-container \
  --volume-driver vieux/sshfs \
  --mount src=sshvolume,target=/app,volume-opt=sshcmd=test@node2:/home/test,volume-opt=password=testpassword \
  nginx:latest
```

volume services

nfs

```
# NFS v3
docker service create -d \
  --name nfs-service \
  --mount 'type=volume,source=nfsvolume,target=/app,volume-driver=local,volume-opt=type=nfs,volume-opt=device=:/var/docker-nfs,volume-opt=o=addr=10.0.0.10' \
  nginx:latest

# NFS v4
```

```
docker service create -d \
    --name nfs-service \
    --mount 'type=volume,source=nfsvolume,target=/app,volume-driver=local,volume-
opt=type=nfs,volume-opt=device=:/,"volume-opt=o=10.0.0.10,rw,nfsvers=4,async"' \
    nginx:latest
```

backup volumes

backup

```
docker run -v /dbdata --name dbstore ubuntu /bin/bash
docker run --rm --volumes-from dbstore -v $(pwd):/backup ubuntu tar cvf
/backup/backup.tar /dbdata
```

restore

```
docker run -v /dbdata --name dbstore2 ubuntu /bin/bash
docker run --rm --volumes-from dbstore2 -v $(pwd):/backup ubuntu bash -c "cd
/dbdata && tar xvf /backup/backup.tar --strip 1"
```