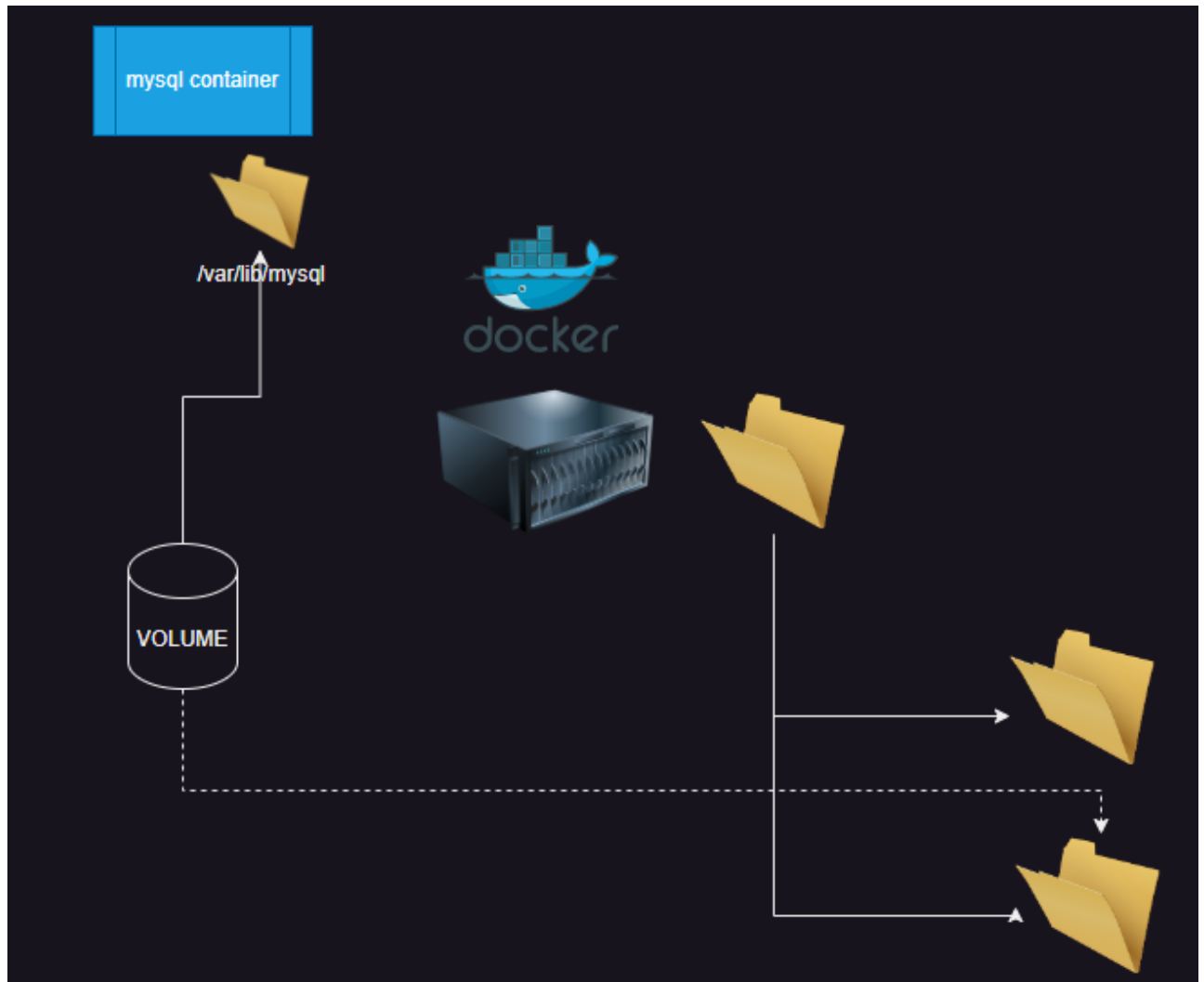


## Docker Volumes

- Overview: [Refer Here](#) for article on volume



- Create docker volume

```
[node1] (local) root@192.168.0.28 ~
$ docker volume create --help

Usage:  docker volume create [OPTIONS] [VOLUME]

Create a volume

Options:
  -d, --driver string    Specify volume driver name (default "local")
  --label list           Set metadata for a volume
  -o, --opt map          Set driver specific options (default map[])
[node1] (local) root@192.168.0.28 ~
$ docker volume create booksdb
booksdb
[node1] (local) root@192.168.0.28 ~
$ docker volume ls
DRIVER      VOLUME NAME
local       booksdb
[node1] (local) root@192.168.0.28 ~
$ |
```

- command

```
docker container run -d -P -e 'MYSQL_ROOT_PASSWORD=admin' -e
'MYSQL_DATABASE=books' -e 'MYSQL_USER=ltdevops' -e 'MYSQL_PASSWORD=admin' --name
booksdb -v booksdb:/var/lib/mysql mysql:8.0-debian
```

```
$ docker container run -d -P -e 'MYSQL_ROOT_PASSWORD=admin' -e 'MYSQL_DATABASE=books' -e 'MYSQL_USER=ltdevops' -e
'MYSQL_PASSWORD=admin' --name booksdb -v booksdb:/var/lib/mysql mysql:8.0-debian
Unable to find image 'mysql:8.0-debian' locally
8.0-debian: Pulling from library/mysql
09f376ebb190: Pull complete
029cd480822b: Pull complete
225aa208e69a: Pull complete
d239b7cd9357: Pull complete
3a0ff6da3549: Pull complete
ae062e186090: Pull complete
040a0a6075c4: Pull complete
efe50e9a5499: Pull complete
b8e648b481d7: Pull complete
c8839099acf6: Pull complete
d5e3c8c90598: Pull complete
13925c5104b8: Pull complete
Digest: sha256:1579fe3a97a436cc10824fc771a07fcedc92213e7ab7604eb5d2976ca419abc8
Status: Downloaded newer image for mysql:8.0-debian
25003384da524704512aac432dd22eb665228af19cc7a0d99c75195093a6d8ed
[node1] (local) root@192.168.0.28 ~
$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
25003384da52   mysql:8.0-debian "docker-entrypoint.s..." 7 seconds ago  Up 5 seconds  0.0.0.0:32769->3306/tcp
, 0.0.0.0:32768->33060/tcp booksdb
[node1] (local) root@192.168.0.28 ~
$
```

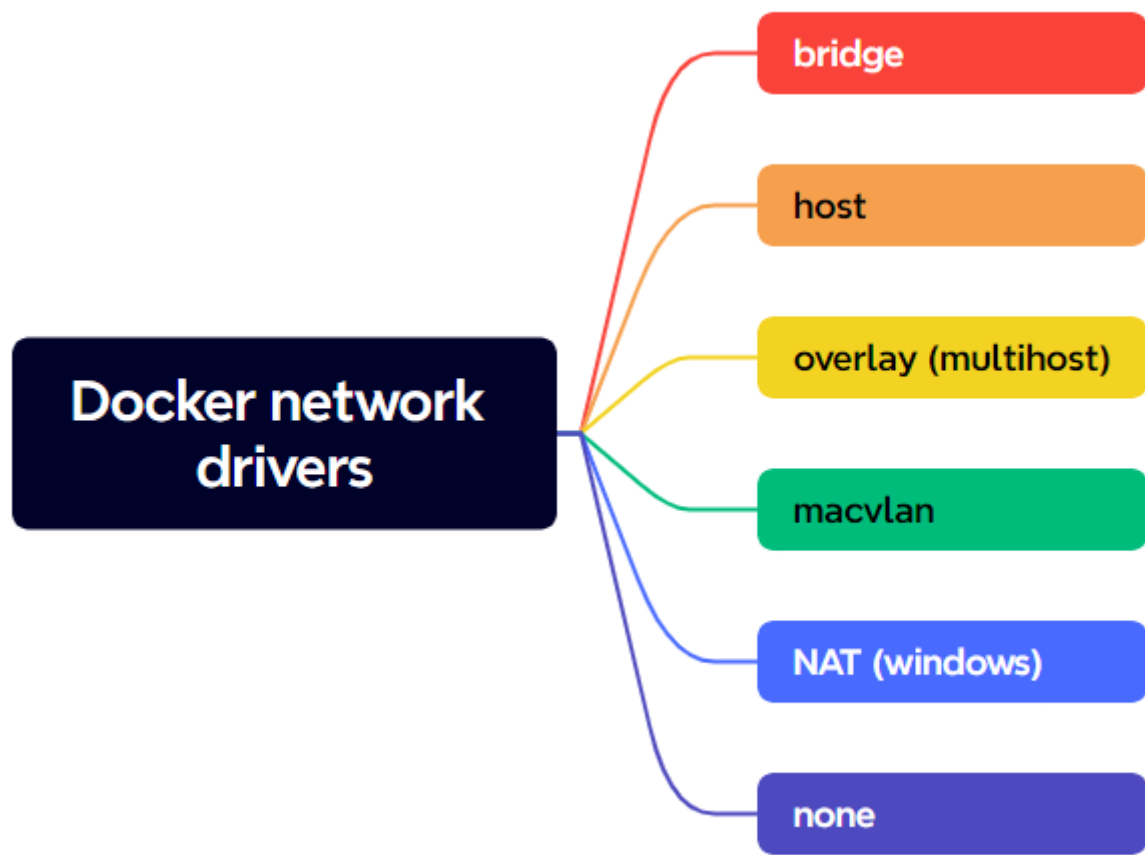
- Lets insert some data, delete the container and recreate another mysql container with same volume and verify data
- As we have discussed in the class the data exists as long as you dont delete volumes.

- VOLUME instruction will create an unnamed volume if the user doesnot pass the `-v`

```
[node1] (local) root@192.168.0.28 ~
$ docker container run -d -p -e 'MYSQL_ROOT_PASSWORD=admin' -e 'MYSQL_DATABASE=books' -e 'MYSQL_USER=ltdevops' -e
'MYSQL_PASSWORD=admin' --name booksdb2 mysql:8.0-debian
bb9215cald72568aa2e2aa2b53f7ab30875d5aa06ee81df740000b34735799d6
[node1] (local) root@192.168.0.28 ~
$ docker volume ls
DRIVER      VOLUME NAME
local       booksdb
local       ee0a821718ef113d1deea1ef5ef676e8d69f483e60d2c638d968b5d6e26c7158
[node1] (local) root@192.168.0.28 ~
$ docker volume inspect ee0a821718ef113d1deea1ef5ef676e8d69f483e60d2c638d968b5d6e26c7158
[
  {
    "CreatedAt": "2024-06-05T02:15:45Z",
    "Driver": "local",
    "Labels": {
      "com.docker.volume.anonymous": ""
    },
    "Mountpoint": "/var/lib/docker/volumes/ee0a821718ef113d1deea1ef5ef676e8d69f483e60d2c638d968b5d6e26c7158/_
data",
    "Name": "ee0a821718ef113d1deea1ef5ef676e8d69f483e60d2c638d968b5d6e26c7158",
    "Options": null,
    "Scope": "local"
  }
]
[node1] (local) root@192.168.0.28 ~
$ docker volume inspect booksdb
[
  {
    "CreatedAt": "2024-06-05T02:04:32Z",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/booksdb/_data",
    "Name": "booksdb",
    "Options": null,
    "Scope": "local"
  }
]
[node1] (local) root@192.168.0.28 ~
$
```

## Docker networks

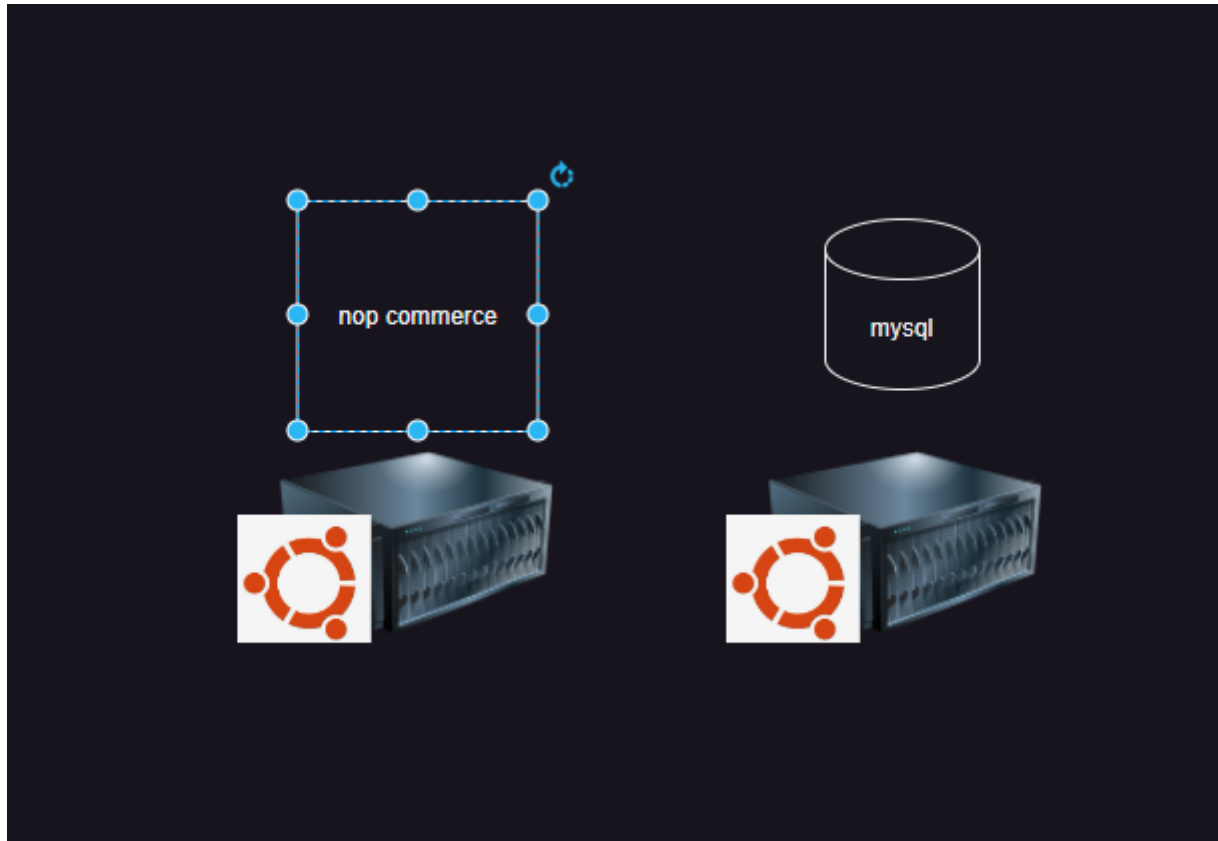
- overview



- [Refer Here](#) for docker networking

Application

- Consider a scenario of containerizing the following architecture



- To do this docker (Every developer, devops etc..) we need to do the following
  - create a new bridge network
  - create a volume
  - create a mysql container and then nop application with env variables, volumes mounted in a new network
  - We might also need to build the nop application
- To simplify this docker had created docker compose
- For this we need to fill the `docker-compose.yml`
- [Refer Here](#) for the changes done to create a Dockerfile