

* Docker Network

Docker network allows containers to communicate to each other and with the outside world.

Note! - When containers talk to each other, we explicitly ~~create~~ or use a docker network (like mynet) so they can communicate by name.

When containers talk to the outside world, they don't need a docker network because docker automatically connects them to a special default network interface that routes traffic to your host machine & then out to the internet.

Two Networking Worlds in Docker

↳ Internal (Container ↔ Container)

↳ Needs a docker bridge network (mynet, mom-network, etc).

↳ Purpose: → private communication between containers.

Example

yaml

networks:

mynet:

driver: bridge

- In docker-compose.yml, both backend and db are on mynet.

External (containers ↔ Outside World)

- Uses Docker's built-in NAT network to reach Internet outbound.
- Else port publishing (-P or ports:) to allow inbound requests.
- You don't manually create this network - Docker provides it automatically.

* Example in a Mern Setup

Structure my-mern-project

 └── backend/
 └── dockerfile
 └── package.json

 └── frontend/
 └── dockerfile
 └── package.json

 └── docker-compose.yml (root level)

Docker network

List network

```
docker network ls
```

Create a network

```
docker network create my-network
```

Connect container to a network

```
docker network connect my-network container-name
```

Disconnect container from a network.

```
docker network disconnect my-network container-name
```

* There are two ways to put a container on a network.

1. At container creation (Preferred)

When you start a container, you can directly attach it to a network using:

```
docker run -d --name myapp -network myst-image-name
```

\Leftarrow --network mynet \rightarrow Attaches the containers to the mynet network at the creation.

\Leftarrow You don't need docker network connect later because it's already in that network.

Ex in mean Stack

```
docker network create mean-network
```

```
docker run -d --name mongo mongo --network mean-network mongo.
```

```
docker run -d --name backendel --network mean-network -p 5000:8000 my-backendel-image.
```

Now backendel can reach mongodb at mongodb:
27017.

2. After Container is already running

If you forgot to attach a container to a network, you connect it later.

```
docker network connect mynet myapp
```

\Leftarrow mynet \rightarrow network name

\Leftarrow myapp \rightarrow container name.