
Scaleway
The cloud that makes sense

Containers 101

Scaleway

Overview

What is a container?

What problems does it solve?

How to get started?

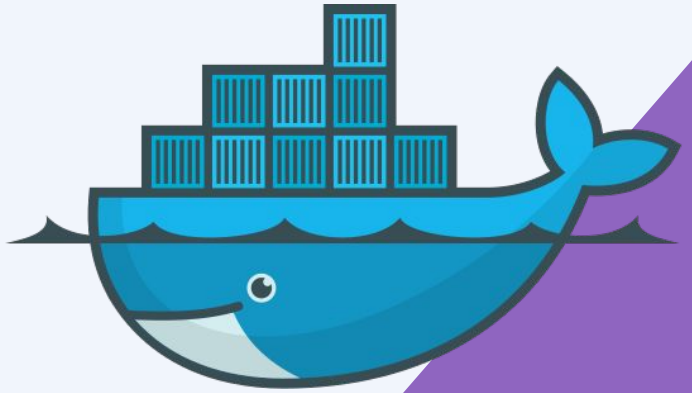
Compute

Storage

Networking

CONTAINERS : WHAT'S IN A NAME ?

- Coming from the shipping industry
- Cause aquatic theme for domain



docker

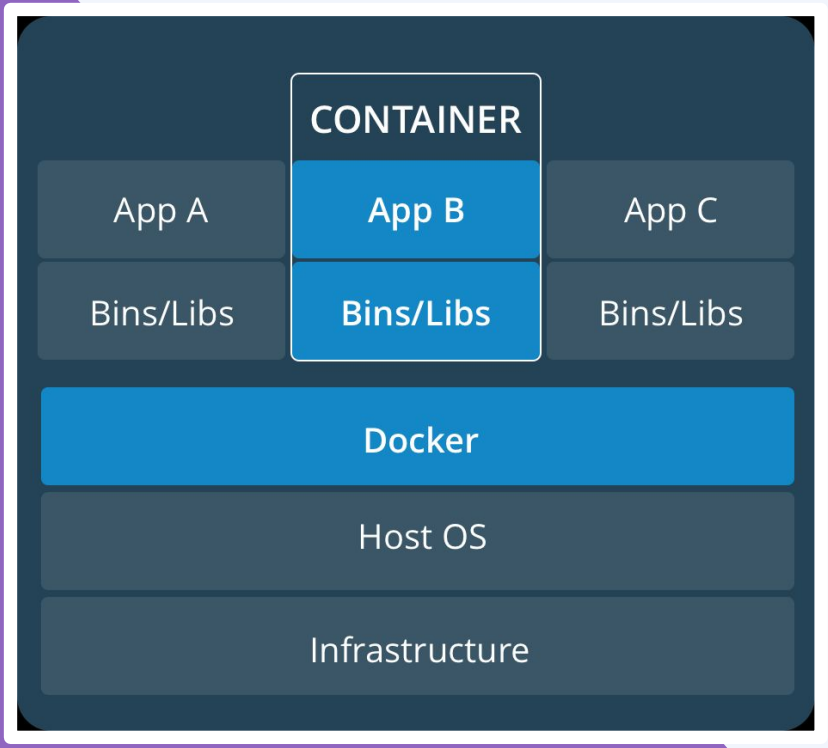
SHIPPING CONTAINERS

- Portability - can be used on any supported types of ships
- A wide variety of cargo can be packed inside
- Standard sizes - standard fittings on ships
- Many containers on a ship
- Isolates cargo from each other

TRANSLATED TO SOFTWARE

- Portability - can be used on any supported system (system with a container execution environment)
- A wide variety of software can be packed inside
- Standard format
- Many containers to a physical node
- Isolates execution of one container from another

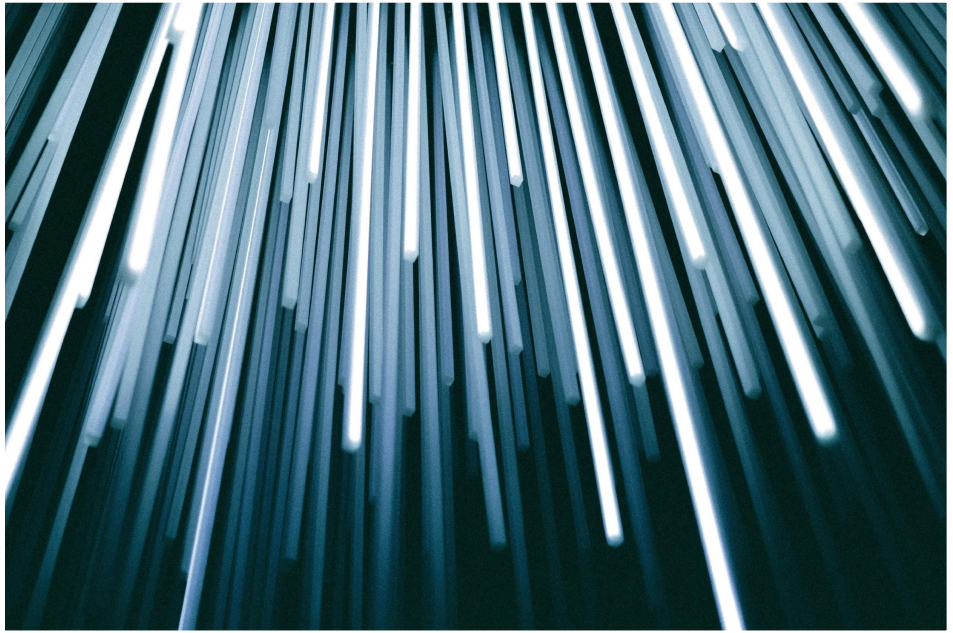
WHAT IS A CONTAINER ?



- **Pack code & dependencies together**
- **Can run anywhere**
- **Execute multiple containers to a physical machine**

SOUNDS FAMILIAR ?

- Same concept as virtual machines
- Pack OS and software together, to run in isolated instances
- Can run anywhere the specific hypervisor runs
- Multiple VMs to a physical machine

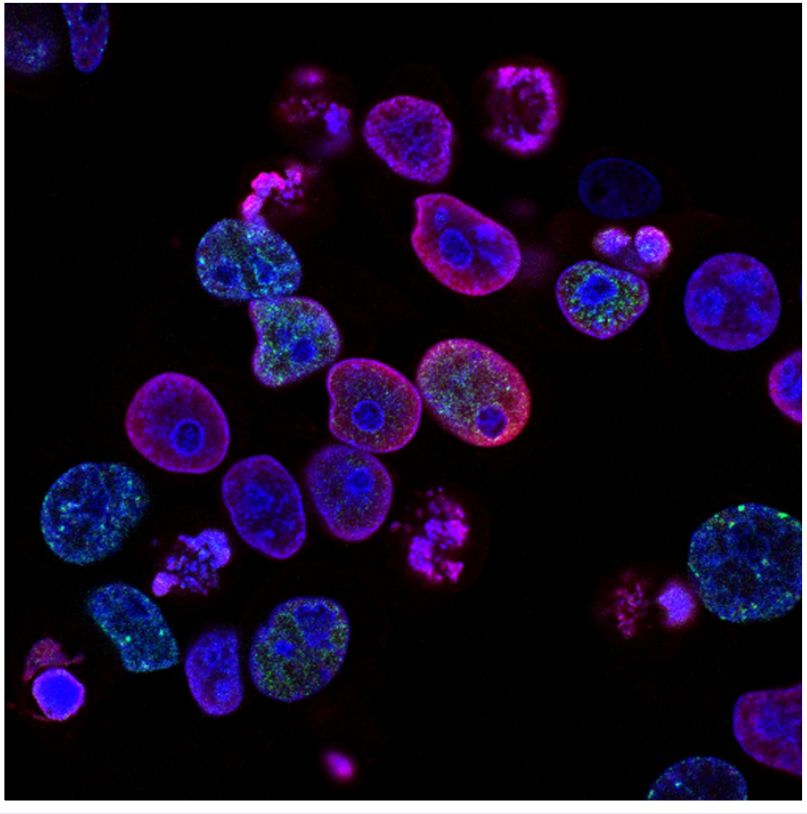


HOW DO VMs WORK ?

- Hypervisor = layer between VM and kernel
- Emulates system calls
- Allows multiple types of operating systems on a machine (Windows on Linux)
- Overhead for hypervisor

CONTAINERS ON THE OTHER HAND...

- Only contain applications and application-related libraries and frameworks, that run on the host machine's kernel
- Smaller
- Lower overhead
- Differences in OS distributions and dependencies are abstracted - same kernel



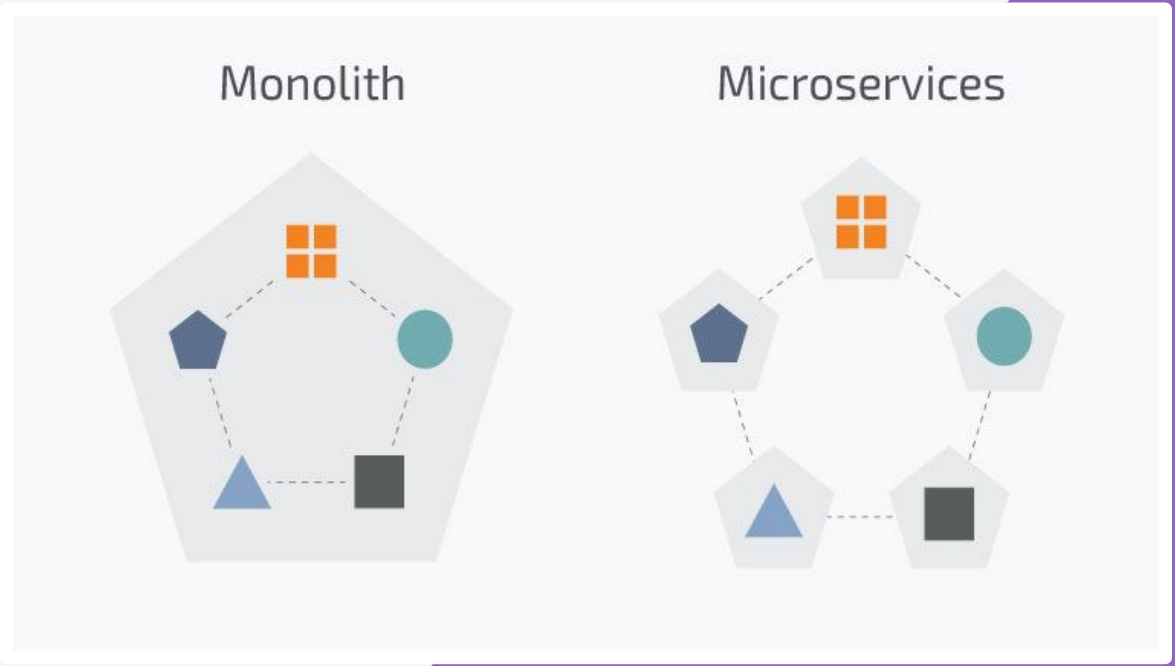
WORKING TOGETHER, NOT AGAINST EACH OTHER

- Windows on Linux possible only with VMs
- Older software needs to be adapted to be run as containers (and won't)
- Usage of VMs as a medium for containers (better isolation and easier scaling)

GREATER MODULARITY IN SOFTWARE

Monolithic application

Independent services that interact (microservices)

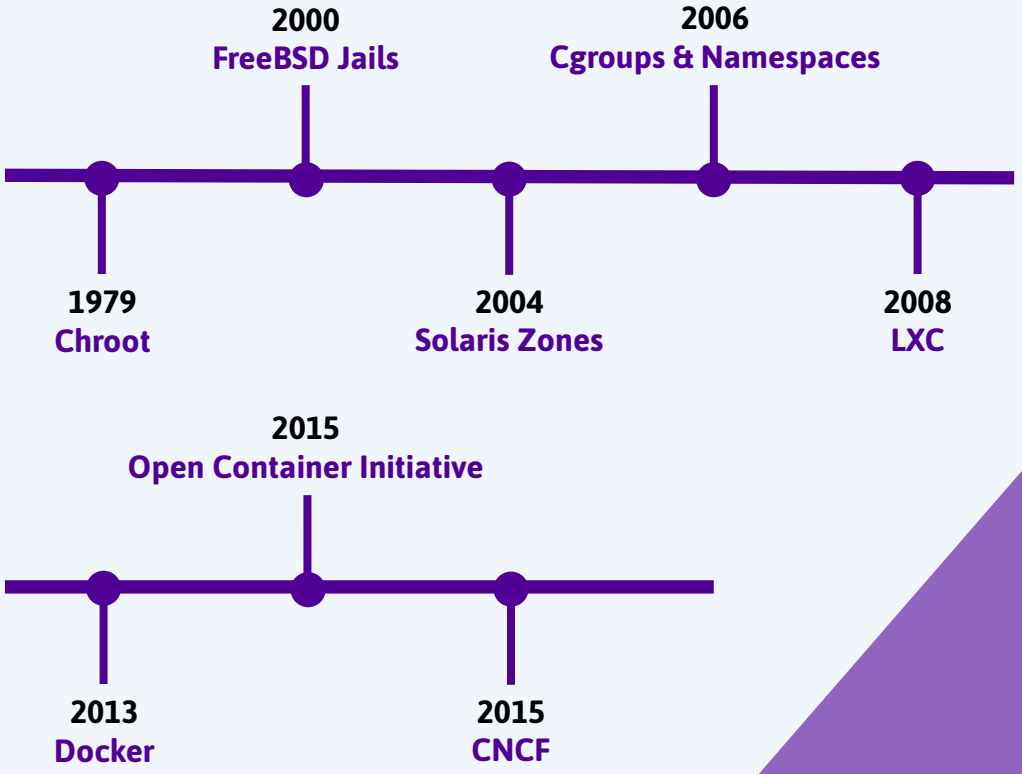


CONTAINERS HISTORY – EARLY DAYS

- **Need for resources to be shared among many users**
  - **multiple terminals connected to the same mainframe**

- **Main problem - execution can cause the main computer to crash**
  - **down for everybody**

CONTAINERS HISTORY – ISOLATING MORE & MORE



Key ideas - Containers

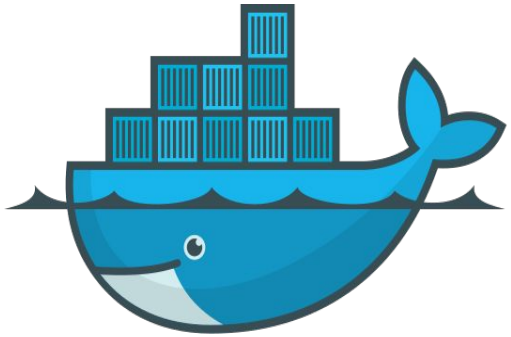
- Isolation
- Standard deployment
- Better resource efficiency

Any questions?

Setting up Docker

Scaleway

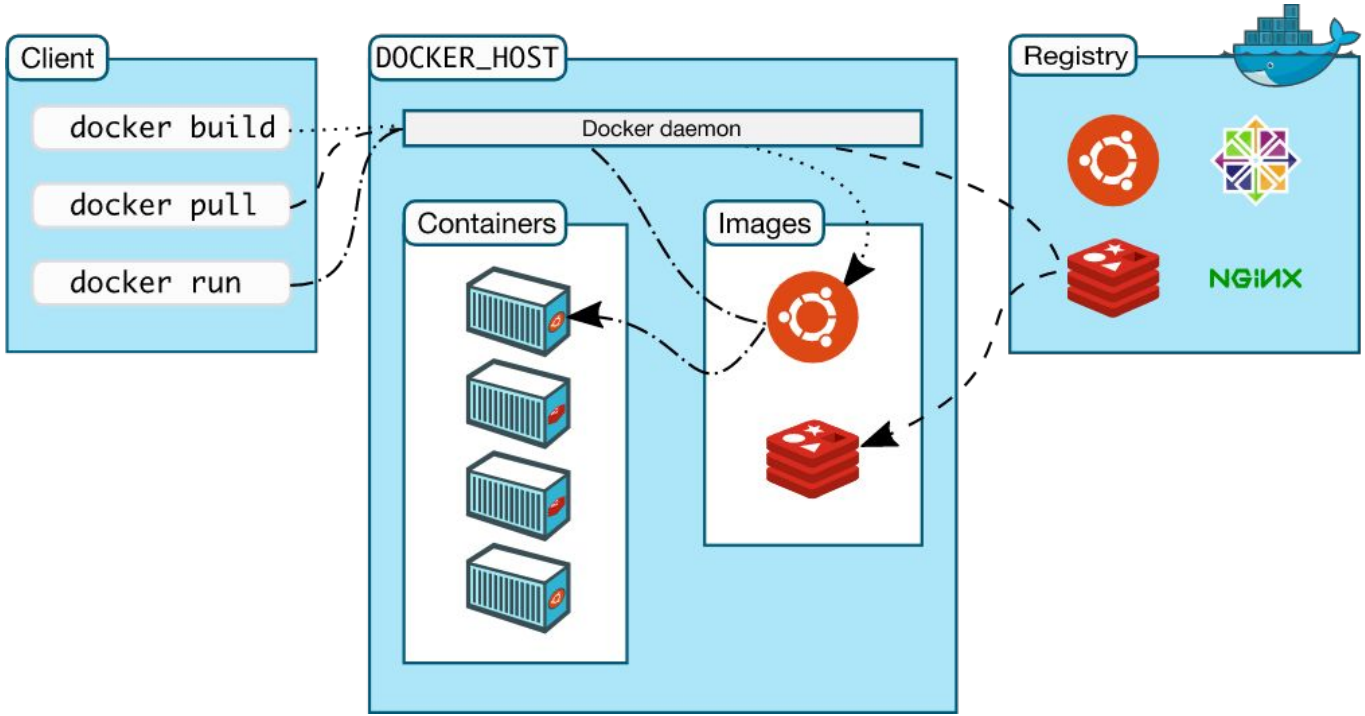
How to get started?



docker

- https://docs.docker.com/get-docker/
- Windows:
  - Enable WSL2
- Mac:
  - Docker for Mac
- Linux
  - Natively supported
- nvidia GPU supported (CUDA)

Big picture


Client
docker build
docker pull
docker run
DOCKER_HOST
Docker daemon
Containers
Images
Registry
NGINX

Compute

Scaleway

docker container

- Where the computation happen
- A container is a set of running processes
  - Isolated using Linux Kernel cgroups
- Stateless and could be restarted at will
- Can be interactive (-i) or non-interactive

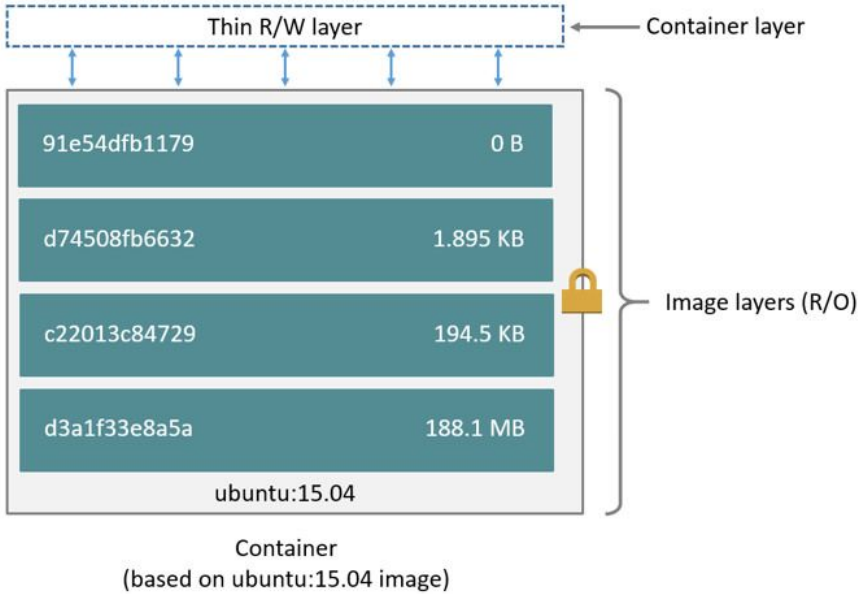
docker container - Available commands

```
$ docker container

Usage:  docker container COMMAND

Commands:
  attach      Attach local standard input, output, and error streams to a running container
  commit      Create a new image from a container's changes
  cp          Copy files/folders between a container and the local filesystem
  create      Create a new container
  diff        Inspect changes to files or directories on a container's filesystem
  exec        Run a command in a running container
  export      Export a container's filesystem as a tar archive
  inspect     Display detailed information on one or more containers
  kill        Kill one or more running containers
  logs        Fetch the logs of a container
  ls          List containers
  pause       Pause all processes within one or more containers
  port        List port mappings or a specific mapping for the container
  prune       Remove all stopped containers
  rename      Rename a container
  restart     Restart one or more containers
  rm          Remove one or more containers
  run         Run a command in a new container
  start       Start one or more stopped containers
  stats       Display a live stream of container(s) resource usage statistics
  stop        Stop one or more running containers
  top         Display the running processes of a container
  unpause     Unpause all processes within one or more containers
  update      Update configuration of one or more containers
  wait        Block until one or more containers stop, then print their exit codes
```

Storage

Scaleway

docker image - What problems does it solve?



- Dependencies managed in a reproducible fashion
- Image are blueprint for a container to be launch
- Images: Classes / Containers: Instance of that class
- Built on layers
- Images got tags, version and metadata such as labels

docker image - Dockerfile

- Recipe for building a Docker image
- Support multi-stage build to avoid having heavy weight images

docker image - Image Registry

- Registry are storing places for your images
- Can be:
  - Local
  - Public (hub.docker.com)
  - Private
- Very useful in CI/CD setups

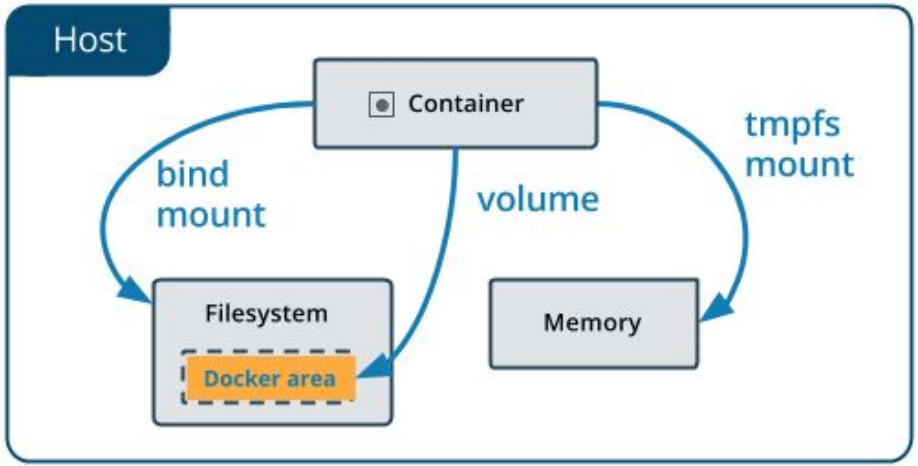
docker image - Available commands

```
$ docker image

Usage:  docker image COMMAND

Commands:
  build       Build an image from a Dockerfile
  history     Show the history of an image
  import      Import the contents from a tarball to create a filesystem image
  inspect     Display detailed information on one or more images
  load        Load an image from a tar archive or STDIN
  ls          List images
  prune       Remove unused images
  pull        Pull an image or a repository from a registry
  push        Push an image or a repository to a registry
  rm          Remove one or more images
  save        Save one or more images to a tar archive (streamed to STDOUT by default)
  tag         Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
```

docker volume - What problems does it solve?

- Images are static
- Containers are stateless by default (tmpfs)
- How do you manage data persistence with containers ?

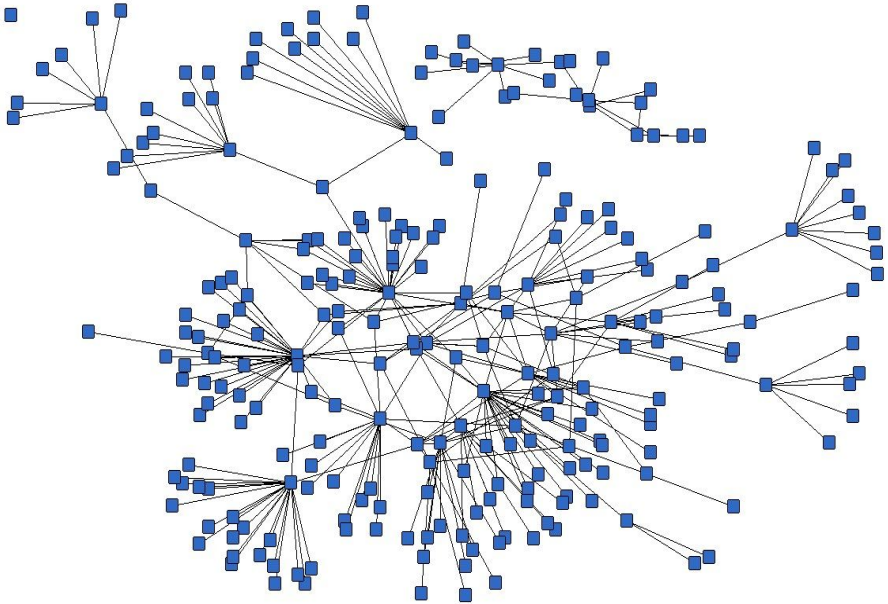
Solution: docker volume



- Copy on Write for changes
- Exist independently of containers
- Are mounted inside containers

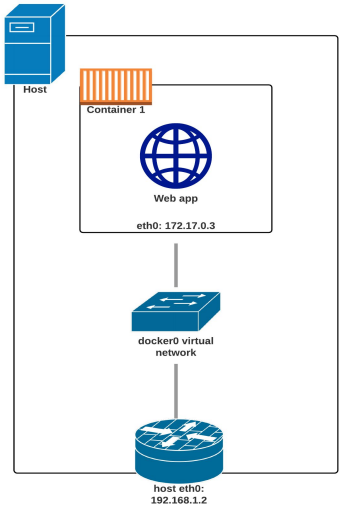
Container Networking

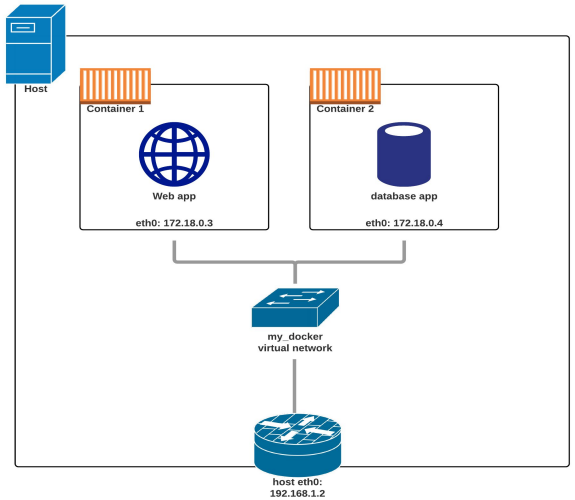
docker network - What problems does it solve?

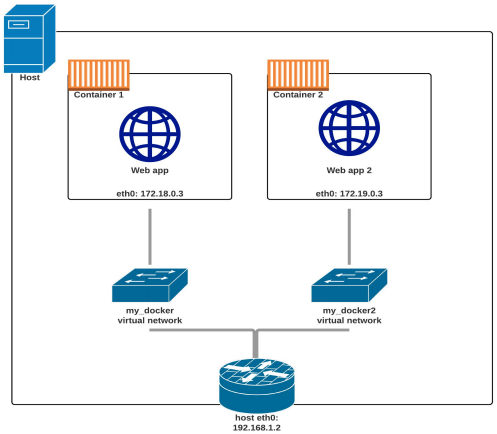


- Containers needs to do I/O locally
- Containers needs to do I/O remotely
- How do you control what a container can see or not see?

docker network - Possible setup







Scaleway

docker network

```
$ docker network

Commands:
  connect     Connect a container to a network
  create      Create a network
  disconnect  Disconnect a container from a network
  inspect     Display detailed information on one or more networks
  ls          List networks
  prune       Remove all unused networks
  rm          Remove one or more networks
```

Conclusion

Scaleway

Conclusion

- Containers:
  - Have been around for a while
  - Are a way to manage your application deployments
  - Are fundamentals of cloud native operations

Learn more

- https://container.training (Self-paced + Video)
- https://www.youtube.com/watch?v=fqMOX6JJhGo
- https://www.youtube.com/watch?v=3c-iBn73dDE

Any questions?

rleone@scaleway.com