

# Getting started with containers

Managing containers and images

# Two important concepts

## Image

## Container

Zutaten	
Hauptgericht	
Für 4 ✓ Personen ⓘ	
250 g	Brot, z. B. Weggli
2 dl	Bouillon
250 g	junger Spinat
2	Eier
5 EL	Paniermehl
50 g	Reibkäse, z.B. Greyerzer
	Salz
1	Schalotte
2	Knoblauchzehen
400 g	gemischte Pilze, z. B. Eierschwämmchen und Steinpilze
½ Bund	Thymian
50 g	Butter
	rosa Pfeffer
20 g	Brunnenkresse

### So gehts

**Zubereitung:** ca. 40 Minuten

**Ruhen lassen:** ca. 30 Minuten

**Gesamt:** 1 Std. 10 Min.

①

Brot in Würfelchen schneiden und in Bouillon einweichen. Spinat waschen und tropfss in eine Pfanne geben. Erhitzen, bis er zusammenfällt. Mit kaltem Wasser abschrecken. Spinat abgießen und gut ausdrücken. Spinat und Eier mit einem Stabmixer pürieren. Mit Paniermehl und Käse zur Brotmasse geben und gut verkneten. Mit Salz abschmecken. Masse ca. 30 Minuten ruhen lassen. Aus der Masse mit nassen Händen Knödel à ca. 50 g formen.

②

Inzwischen Schalotte und Knoblauch hacken. Pilze rüsten und nach Belieben klein schneiden. Kräuterblättchen abzupfen. Reichlich Wasser in einer weiten Pfanne aufkochen. Knödel portionweise darin knapp unter dem Siedepunkt ca. 5 Minuten ziehen lassen, bis sie an der Oberfläche schwimmen und fest werden. Herausnehmen und abtropfen lassen. Butter in einer weiten, beschichteten Bratpfanne erhitzen. Schalotte, Knoblauch und Pilze darin bei mittlerer Hitze ca. 3 Minuten braten. Knödel dazugeben und kurz mitbraten. Mit Thymianblättchen und Pfeffer bestreuen. Mit Brunnenkresse servieren.

Rezept: Daniel Tinembart

FAST  
FERTIG



- read-only description
- stored on longer term
- can be used as a base

- based on the image
- short-lived
- usually only minor adjustments

# The concept of layers

- In docker: images have a starting point
- User makes changes/installations
- Stored in a layer on top of existing
- Creating an image:
  - From a container: `docker commit`
  - From a Dockerfile

# The docker engine

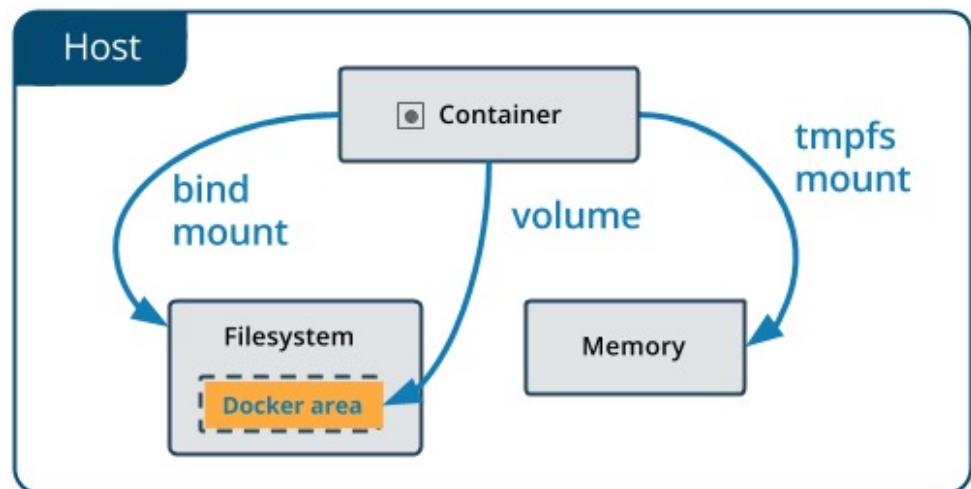
- Manages both the
  - images
  - containers
- Therefore layers are efficiently handled
- Both images and containers will remain available unless specified
- Two important commands:
  - `docker image ls`
  - `docker container ls`

# Sharing an image

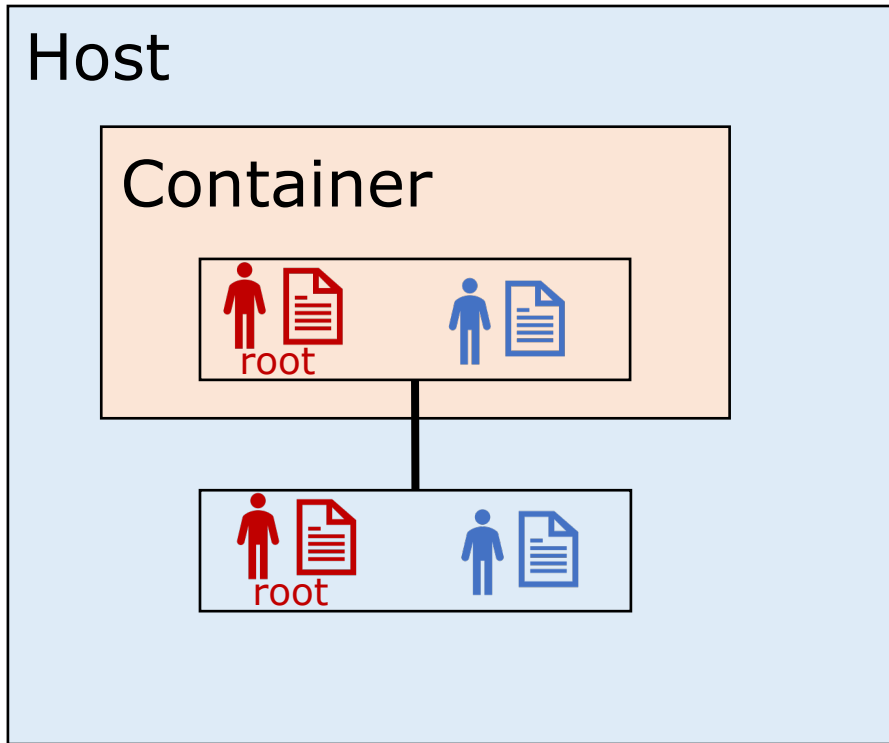
- docker hub (open to the world)
  - Alternatives: quay.io, singularityhub, gitlab and github container repositories ...
- command `docker save`
- Dockerfile

# Mounting

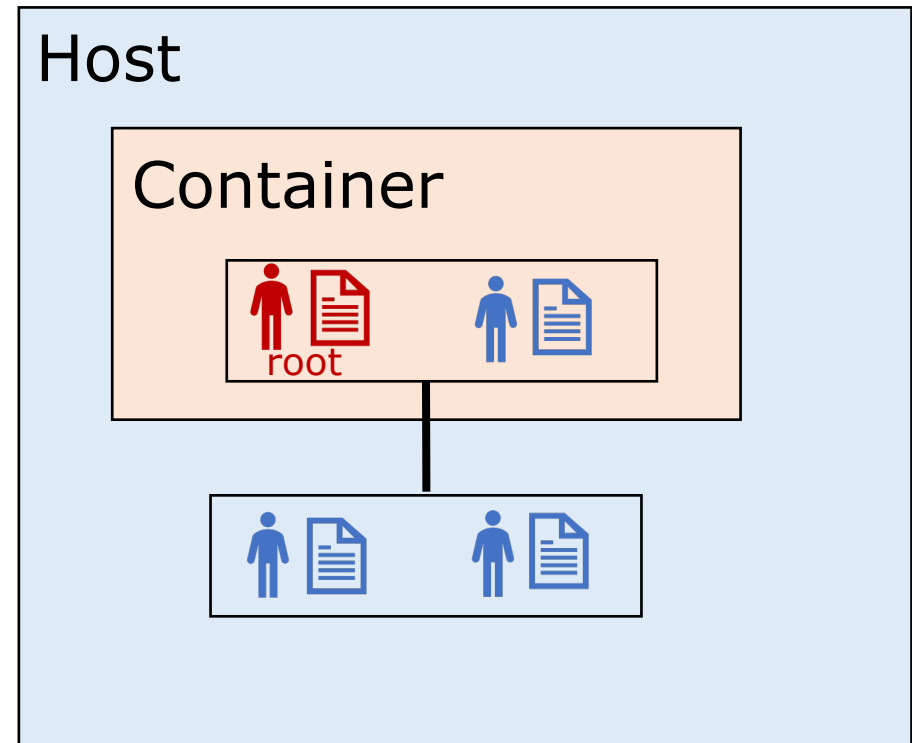
- **Bind-mount:** Make a directory on the host available to the container
- **Volume:** Disk space reserved and managed by docker (isolated)



# Identity



Linux



MacOS

```
docker run -u "$(id -u):$(id -g)"
```