Playing with Containers

Marco Mambelli
Fermilab
5/22/2024

Container

Large metal box used for the transportation of freight by road, rail, sea, or air

https://dictionary.cambridge.org/dictionary/english/container

Container

Discrete environment set up within an operating system in which one or more applications may be run, typically assigned only those resources necessary for the application to function correctly.

Standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another. (Docker 2013)

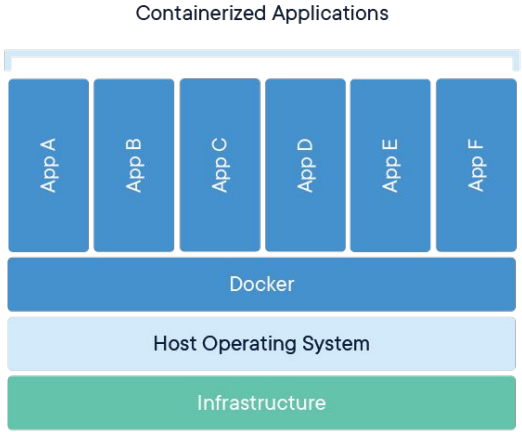
It's a Linux thing

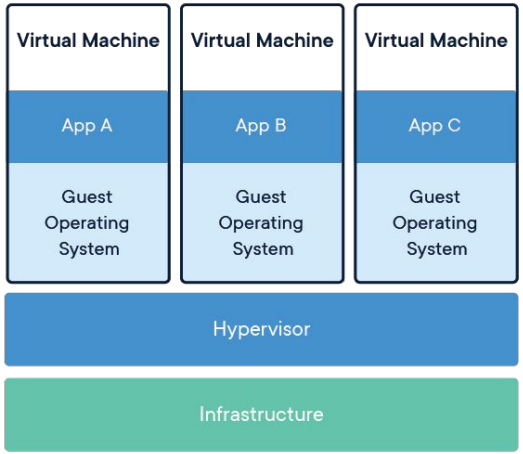
Shared kernel space

- Kernel of the host OS

Separate user space

- App + Library



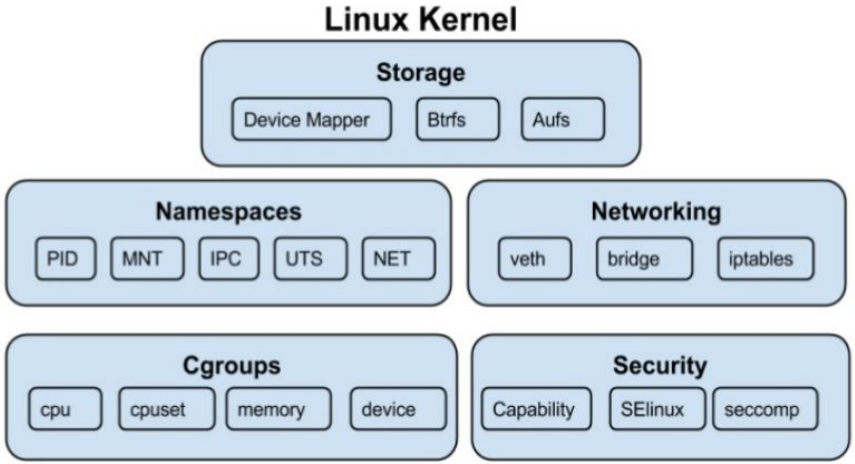


User mode	**User applications**	For example, bash, LibreOffice, Apache OpenOffice, Blender, 0 A.D., Mozilla Firefox, etc.				
	Low-level system components:	**System daemons**: *systemd, runit, loginд, networkd, soundd...*	**Windowing system**: *X11, Wayland, Mir, SurfaceFlinger (Android)*	**Other libraries:** *GTK+, Qt, EFL, SDL, SFML, FLTK, GNUstep*, etc.		**Graphics**: *Mesa 3D, AMD Catalyst, ...*
	C standard library	`open()`, `exec()`, `sbrk()`, `socket()`, `fopen()`, `calloc()`, ... (up to 2000 subroutines) *glibc* aims to be POSIX/SUS-compatible, *uClibc* targets embedded systems, *bionic* written for Android, etc.				
Kernel mode	**Linux kernel**	`stat`, `splice`, `dup`, `read`, `open`, `ioctl`, `write`, `mmap`, `close`, `exit`, etc. (about 380 system calls) The Linux kernel System Call Interface (SCI, aims to be POSIX/SUS-compatible)				
		Process scheduling subsystem	IPC subsystem	Memory management subsystem	Virtual files subsystem	Network subsystem
		Other components: ALSA, DRI, evdev, LVM, device mapper, Linux Network Scheduler, Netfilter Linux Security Modules: *SELinux, TOMOYO, AppArmor, Smack*				
Hardware (CPU, main memory, data storage devices, etc.)						

Use Linux VM on other OSes

https://www.docker.com/resources/what-container/
https://i.stack.imgur.com/2mDPs.png
https://dockerlabs.collabnix.com/beginners/difference-vm-containers.html

Container abstraction

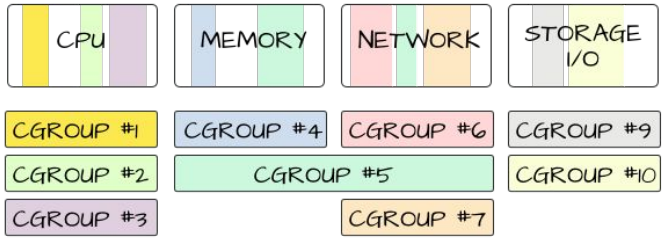


Isolation with Namespaces
(2002, EW Biederman, P Emelyanov, A Viro, and C Gorcunov)

- PID namespace for process isolation.
- NET namespace for managing network interfaces.
- IPC namespace for managing access to IPC resources.
- MNT namespace for managing filesystem mount points.
- UTS namespace for isolating kernel and version identifiers.

Resource Limitation with cgroup (2008, P Menage and R Seth)

Packaging files and dependencies with rootfs



https://q15928.github.io/2021/01/09/container-101/
https://mairin.wordpress.com/2011/05/13/ideas-for-a-cgroups-ui/
https://en.wikipedia.org/wiki/Linux_kernel_interfaces
https://gcore.com/learning/containers-vs-virtual-machines/

Container runtime

Software package that knows how to leverage specific features on a supported operating system to run the containers.

High level

- Docker engine
- Podman
- CRI-O (Container Runtime Interface -- CRI -- specification)
- Apptainer/Singularity

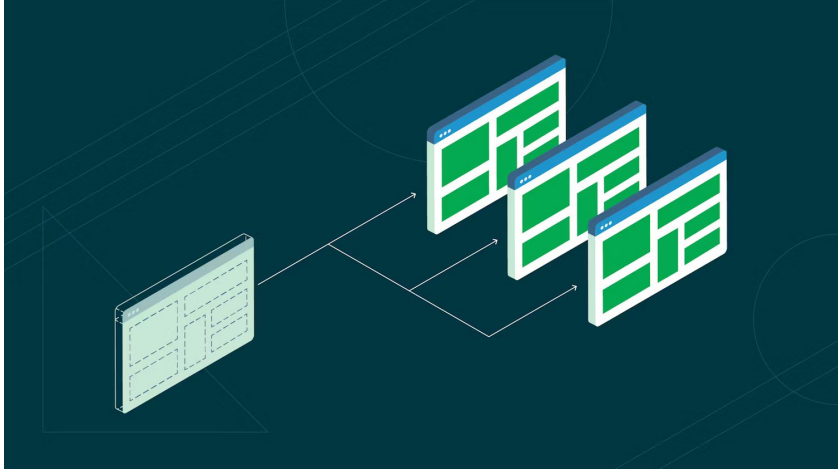
Low level

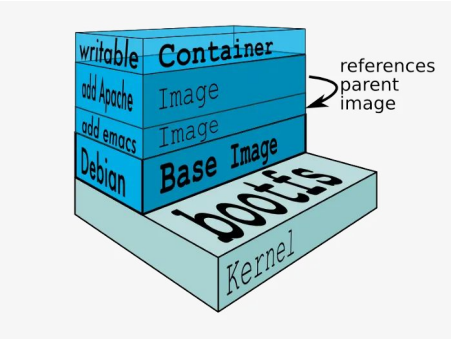
- runc (Open Container Initiative -- OCI)
- crun
- runhcs (with VM Hypervisor)
- containerd

https://www.cloudraft.io/blog/container-runtimes

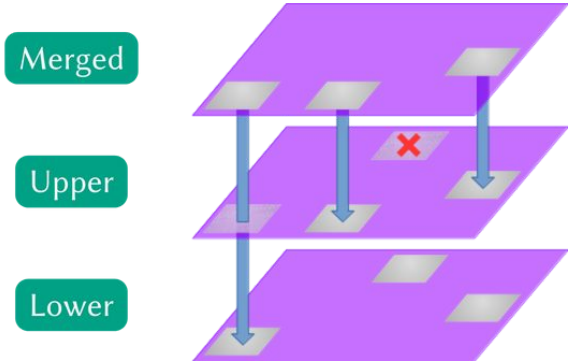
Container Image

Read only template used to create containers





Overlays and underlays (UnionFS/aufs)



https://circleci.com/blog/docker-image-vs-container/
https://velog.velcdn.com/images/koo8624/post/3e431335-53b5-4f0e-90bd-eb85b6c3c4fa/ufs.jpeg
https://www.nemunai.re/post/unveiling-whiteout-files/
https://embeddedcomputing.com/technology/processing/understand-what-an-overlayfs-is-and-how-it-works

Container/Image Definition file (Dockerfile)

Starting from another image

One layer for each RUN line

One entry point (CMD)

```
# syntax=docker/dockerfile:1

FROM ubuntu:22.04
COPY . /app
RUN make /app
CMD python /app/app.py
```

https://docs.docker.com/reference/dockerfile/

Containers/Images Registry and Repository

Repository: storage for your containerized application images
Registry: both a collection of repositories and a searchable catalogue where you manage and deploy images

Docker Hub (https://hub.docker.com/)

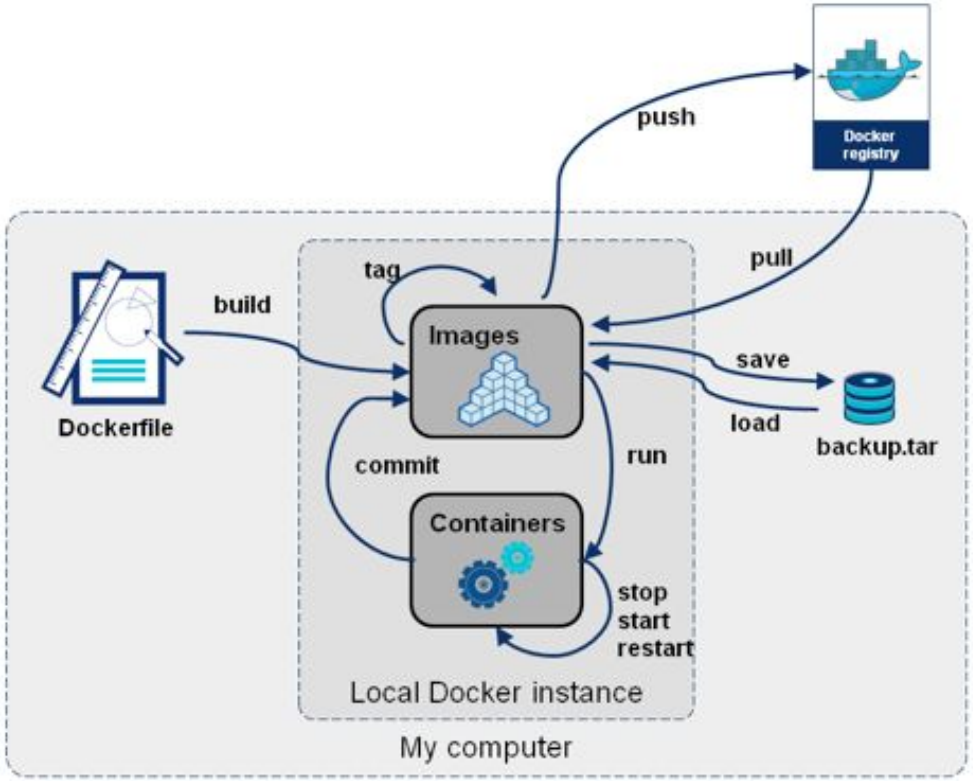
- Public, well-known, but rates and users limitations in the free tier
- Docker-Sponsored Open Source projects
  - https://www.docker.com/community/open-source/application/

Harbor (https://goharbor.io/)

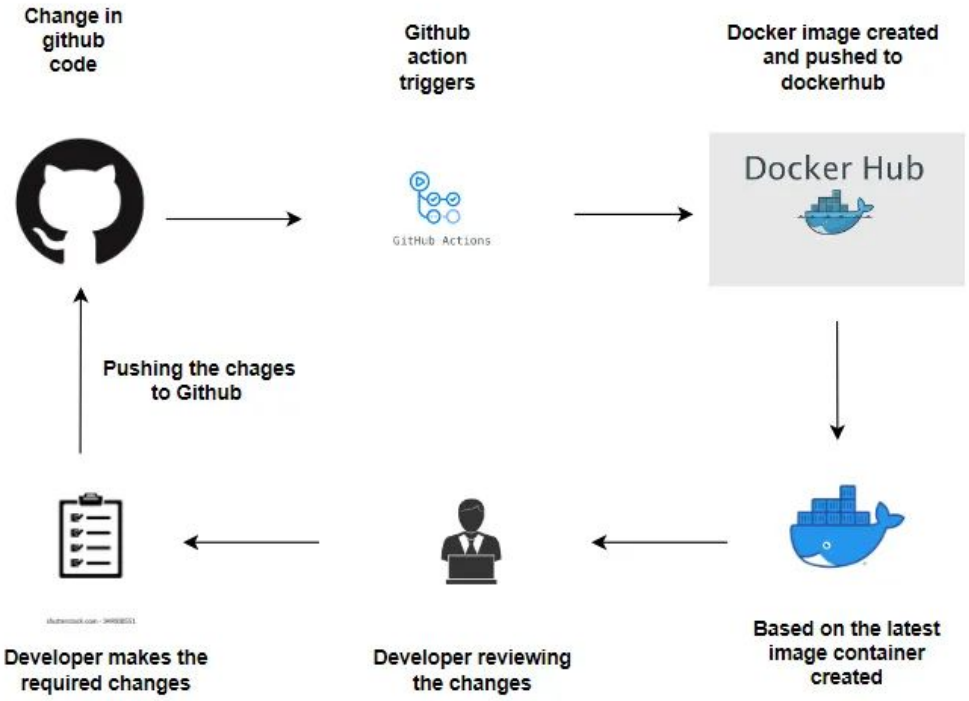
- Designed for Kubernetes, open source, self hosted
- No rate limitations
- Available at Fermilab
  - https://ssiwiki.fnal.gov/wiki/Container_Build_Service_Home

More: Amazon's Elastic Container Registry (ECR), Azure Container Registry (ACR), Google Cloud's Container Registry (GCR), GitHub's Container Registry (GitHub Packages), JFrog Container Registry

Build, push, pull, run, stop, …



Building with GitHub actions

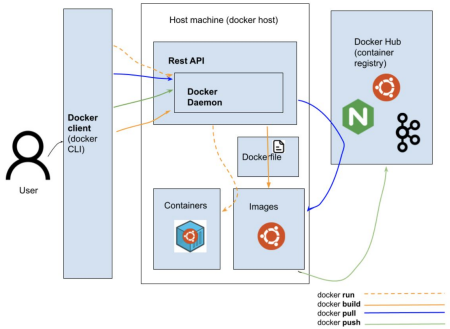


Docker

Open-source platform used to build, ship, and run applications inside containers

Docker Engine, a service that runs on your system and manages the creation and running of containers. It builds containers from images, starts and stops them, and manages their resources

Apache 2.0. But commercial use of Docker Engine obtained via Docker Desktop within larger enterprises (> 250 employees OR revenue > $10 million USD), a paid subscription is required.



https://docs.docker.com/engine/
https://hsf-training.github.io/hsf-training-docker/index.html

Podman

podman

Open-source, daemonless container engine designed for managing containerized applications on Linux systems. It offers a lightweight and efficient alternative to Docker while maintaining compatibility with the Open Containers Initiative (OCI) specifications

- Daemonless
- Rootless
- Docker CLI compatible

https://mydeveloperplanet.com/2023/05/24/is-podman-a-drop-in-replacement-for-docker/?ref=oramind.com

COntainers with LInux vms on a MAc

Colima

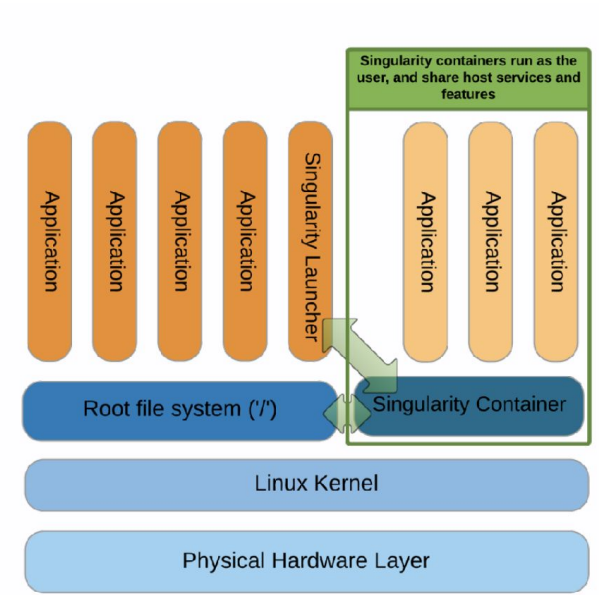
- Install colima, docker (CLI), and/or podman (CLI) via Homebrew
- All via command line
- Multiple runtimes (docker-compatible by default)
- Easy support for both arm64 and amd84 VMs

Apptainer (Singularity)

Open-source platform for building and running high-performance containers designed for scientific computing and High-Performance Computing (HPC) environments

Sylabs Singularity Pro/CE

- Daemonless, rootless (unprivileged mode)
- Same user inside and outside the container
- Single file SIF container format
- Direct file system access
- Minimum overhead -- execv()



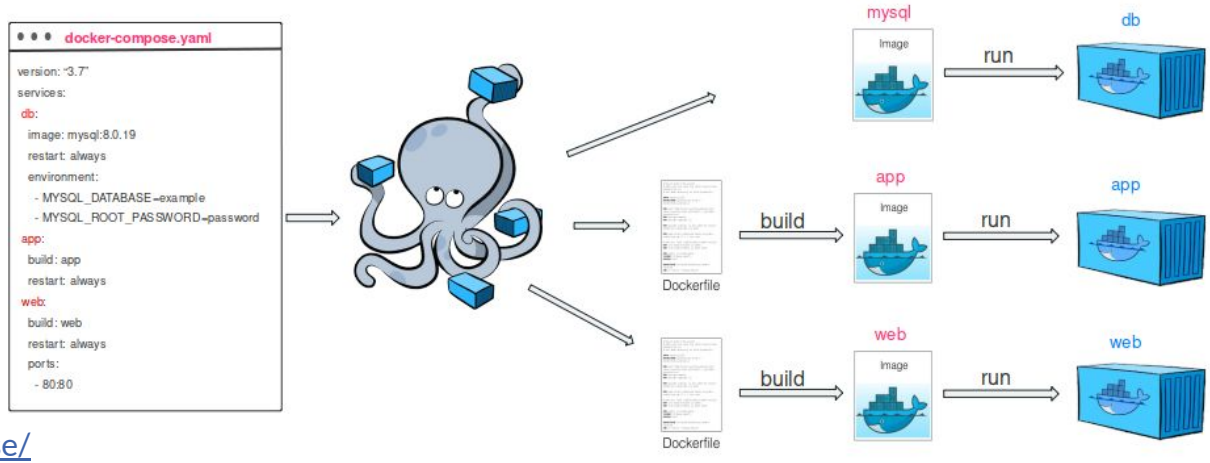
https://apptainer.org/
https://sylabs.io/
https://hsf-training.github.io/hsf-training-singularity-webpage/

Composition/Orchestration

Automation of provisioning, deployment, scaling, and management of containerized applications

Docker/Podman Compose

- Tool for defining and running Docker containers by reading configuration data from a YAML file



https://www.biaudelle.fr/docker-compose/

Alnoda Workspaces

Alnoda Hub is a library of portable, containerized workspaces and workspace applications

- https://alnoda.org/
- https://alnoda.org/registry/workspaces/

Why workspaces, what is different

- Use a container as development VM
- ITB for testing
- Fast onboarding
- init, supervisor (http://supervisord.org/), systemd problematic in containers
- systemctl replacement
    - https://github.com/gdraheim/docker-systemctl-replacement

Demo time!

GlideinWMS containers and workspaces

- https://github.com/glideinWMS/containers
- https://github.com/glideinWMS/containers/tree/main/workspaces

On a Linux terminal (Podman required)

```
mkdir -p $HOME/ws-test/gwms; cd $HOME/ws-test
git clone https://github.com/glideinWMS/containers.git
git clone https://github.com/glideinWMS/glideinwms.git gwms/glideinwms
GMWS_PATH=$HOME/ws-test/gwms/ podman-compose up -d
podman exec -it ce-workspace.glideinwms.org ~/scripts/startup.sh
podman exec -it factory-workspace.glideinwms.org ~/scripts/startup.sh
podman exec -it frontend-workspace.glideinwms.org ~/scripts/startup.sh
# Complete CI-Logon authentication
podman exec -it frontend-workspace.glideinwms.org ~/scripts/run-test.sh
# Play more w/ the containers: podman exec -it HOST /bin/bash
```