

# Container Runtimes

Ryan Tolboom

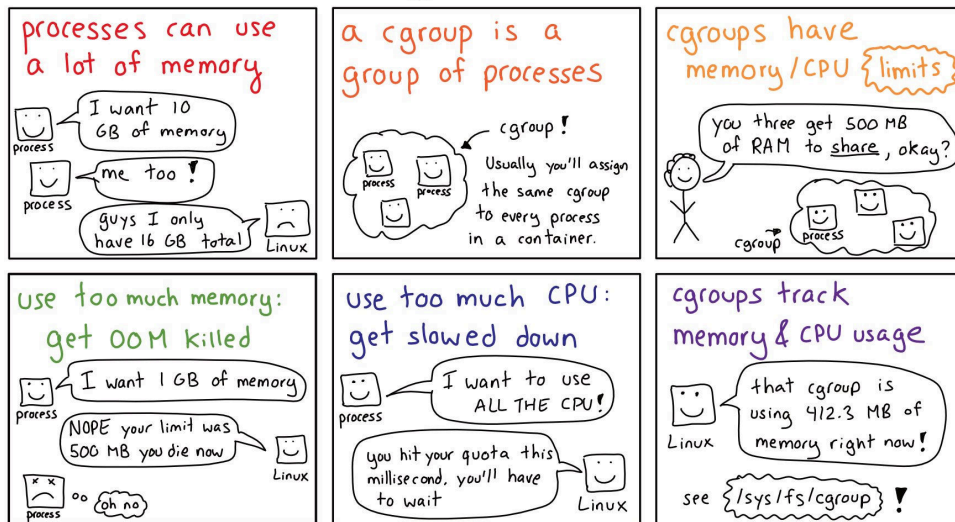
New Jersey Institute of Technology



# Background

JULIA EVANS  
@b0rk

## cgroups



"cgroups" by [Julia Evans](#) is licensed under [CC BY-NC-SA 4.0](#)

- Recall that containers are largely just [namespaces](#) and [cgroups](#)
- A container runtime is the software that is used to support the containerization of software

# What specifically does a runtime do?

- image management
- lifecycle management
- creation
- resource management

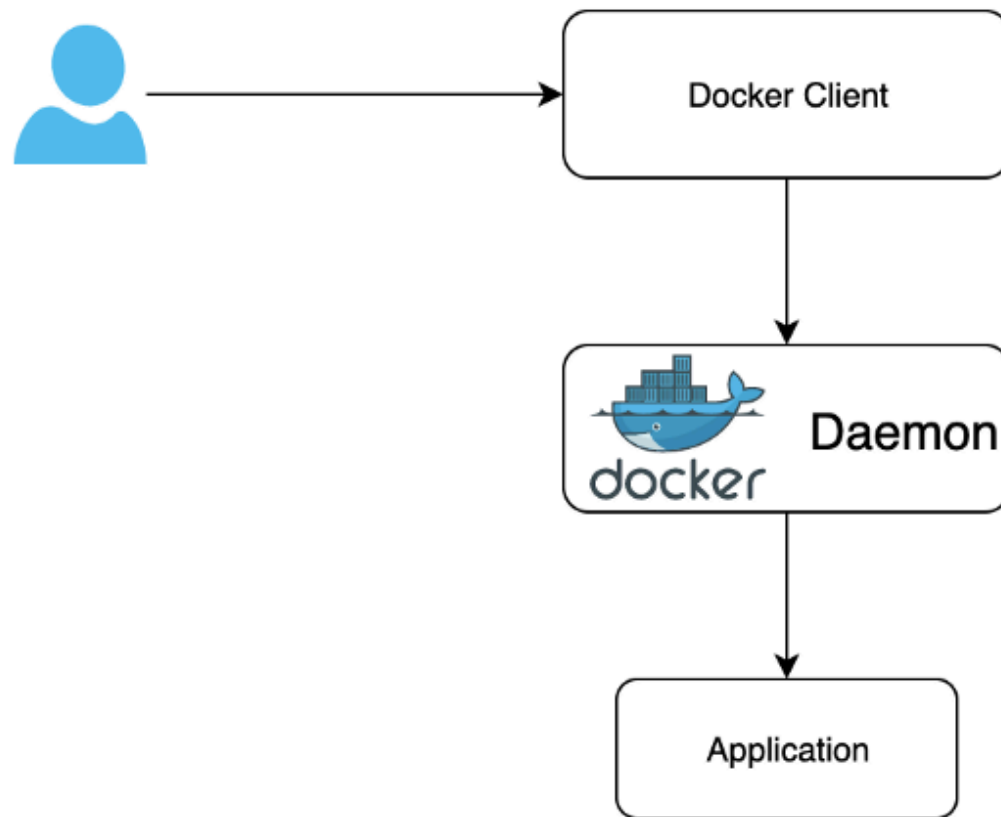


[Cloud Native Landscape](#) is used under fair use

# Docker (original)

- One of the first runtimes, and as such many things strive to be Docker compatible.
- Runs on many platforms and is easy to use
- Extremely popular *not just for the runtime* but also for the tools that work with it such as their container registry Docker Hub.
- Supports app containers
- Requires a daemon to be running as root
- Losing ground in the production environment, but still common on dev machines

# Old Docker



Docker Version < 1.11.0

[Docker and OCI Runtimes](#) is used under fair use



[containerd logo](#) is used under fair use

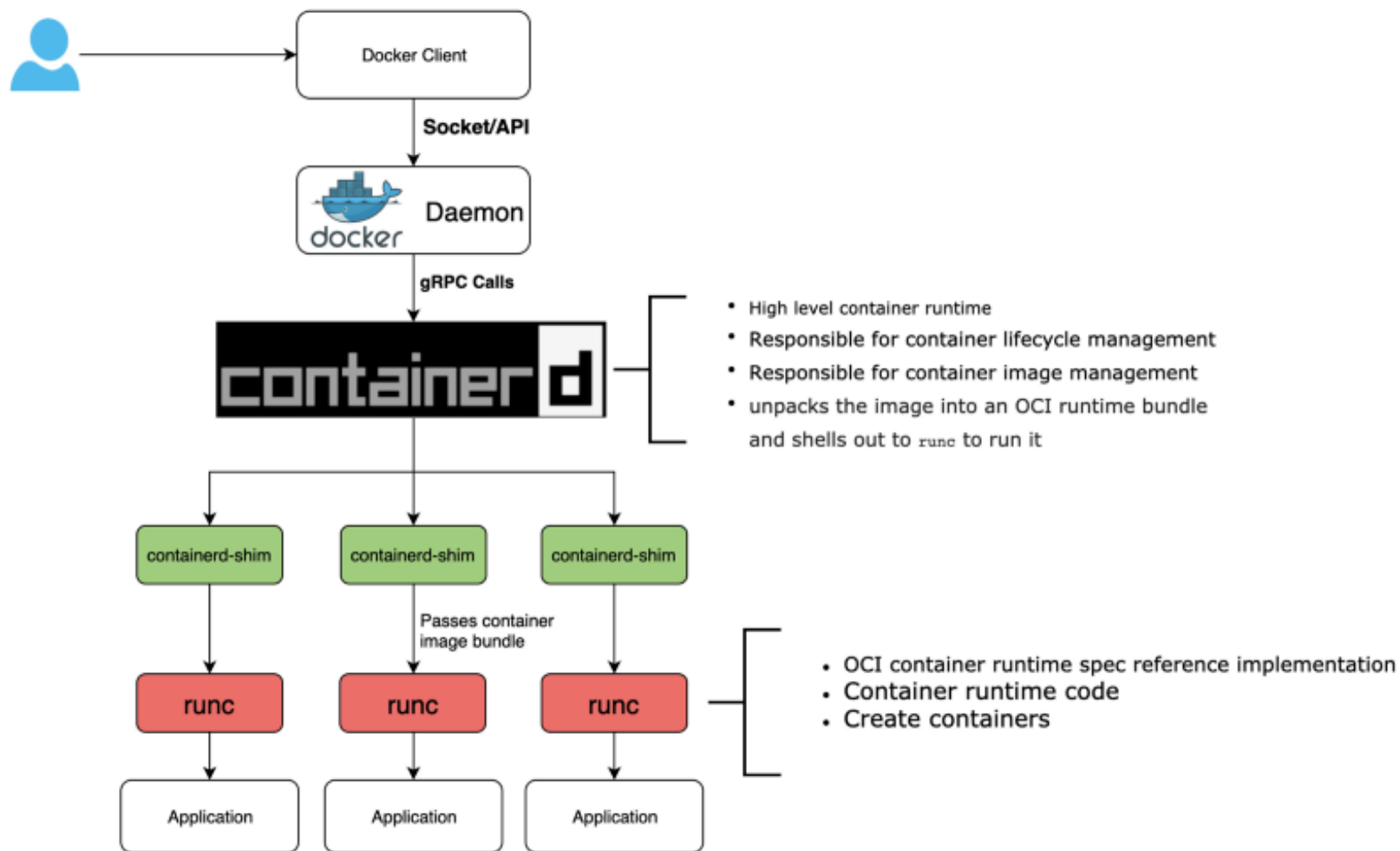
- Born from the Open Container Initiative (OCI) which was created to make open standards for containers
- Daemon that runs in the background and supports app containers (sound familiar?)
- Relies on a lower-level tool called runc (written by Docker)
- Current versions of Docker run on containerd



[runc logo](#) is used under fair use

- Universal, lightweight container runtime
- Designed for security
- No Docker dependencies
- Supports a very simple model of container execution

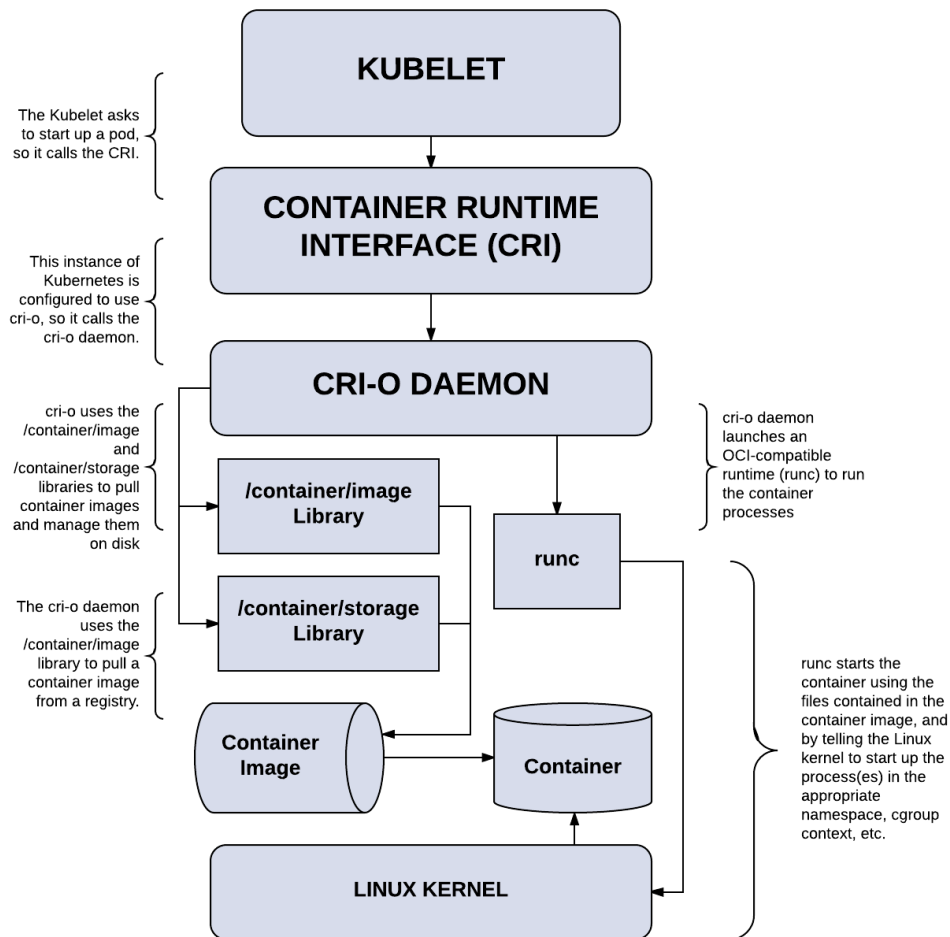
# New Docker



[Docker and OCI Runtimes](#) is used under fair use



# CRI-O



[Introducing CRI-O 1.0](#) is used under fair use

- Implements Kubernetes Container Runtime Interface (CRI) with OCI standards
- Runs as a daemon
- Support app containers
- Bridges the gap between K8s and containers, no need to run a **docker** command in-between
- Not meant for use by devs

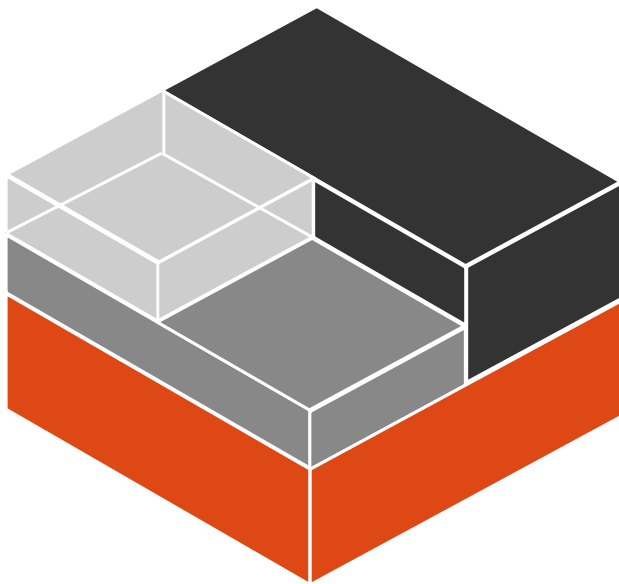
# Podman

- daemonless tool to use OCI containers
- has a similar CLI to `docker`
- supports similar features as `docker`
- Linux only



podman

[podman logo](#) is used under fair use



[“LXC logo”](#) by [LinuxContainers](#) is licensed under [CC BY-NC-SA 4.0](#)

- Container engine designed for *system* containers
- Rootless, daemonless, and fast
- Keep this in mind if you encounter a *tough* virtualization feature
- Works with tools Linux admins are already familiar with
- Linux only

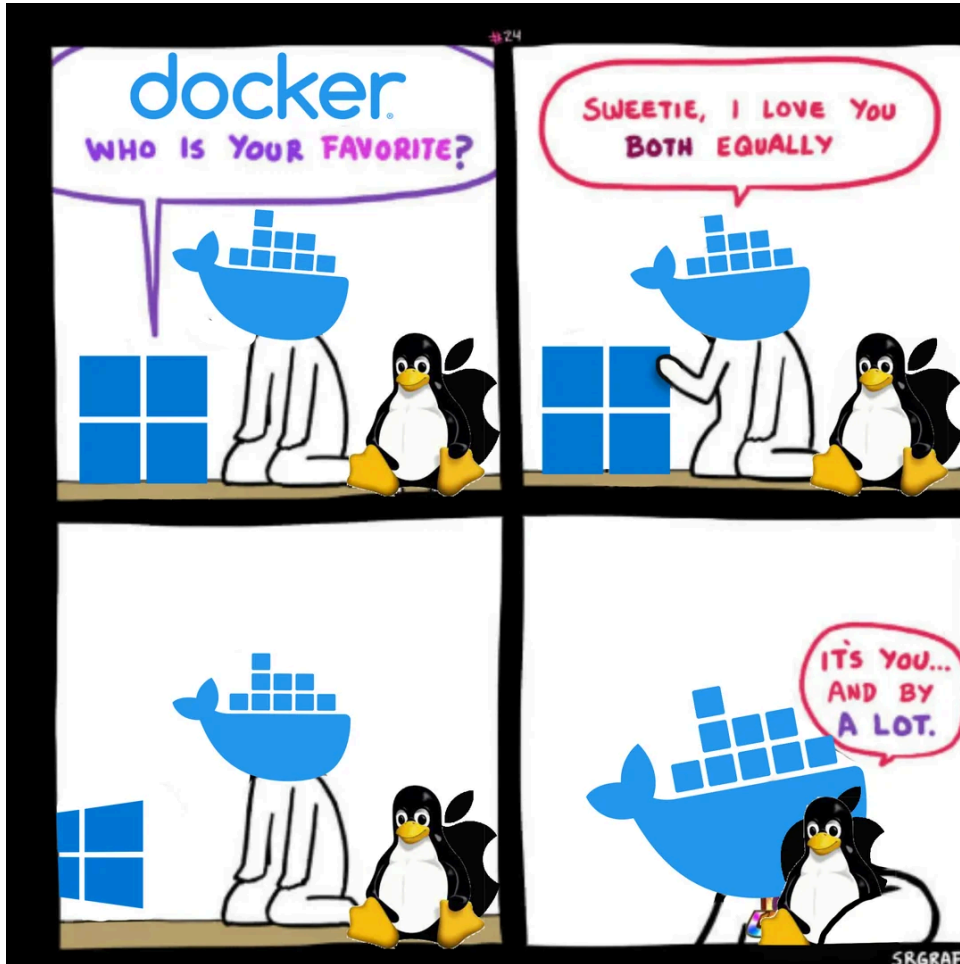
# Just for fun: Bocker

- A lot of what Docker does can be implemented in around 100 lines of BASH!
- Remember these are things that are *already* built in to the Linux kernel
- Obviously Linux only



[You should use docker - it's so light weight!](#) is used under fair use

# What does this mean?

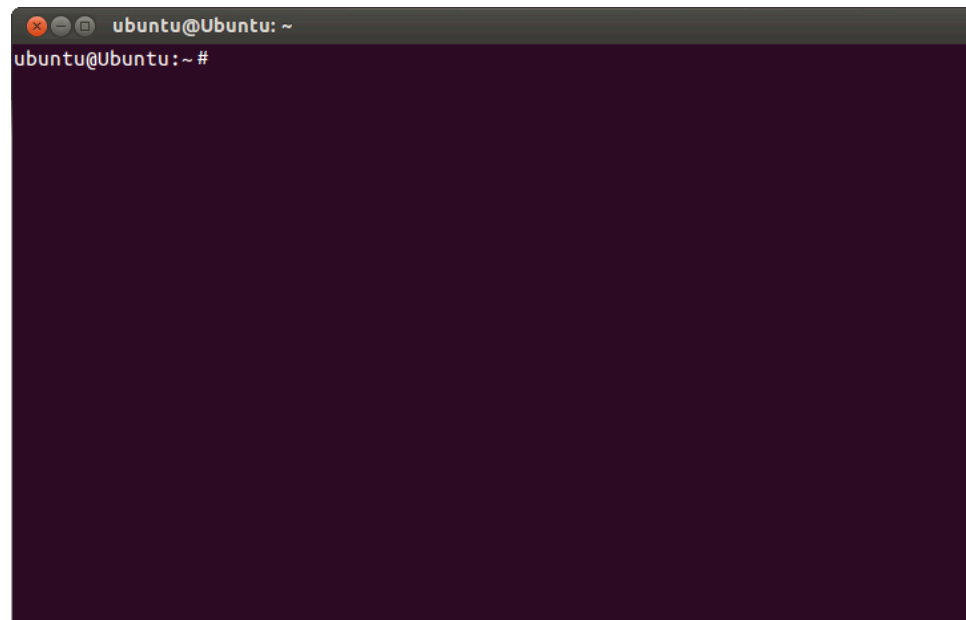


[Docker does have some favorites though...](#) is used under fair use

- As a sysadmin, you have options
- Almost all Linux-based
- A few kernel features can spawn a massive shift in software deployment

# Don't forget to demo...

- the Docker daemon running
- a `docker save` to show the structure of the filesystem



[“Terminal-linux-ubuntu.png”](#) by [Ricardoborges](#) is licensed under [CC BY-SA 4.0](#)