

# CONTAINER RUNTIMES

DRAW SOME LINES

Vincent Batts @vbatts

```
$> finger $(whoami)
```

**Login:** vbatts

**Name:** Vincent Batts

**Directory:** /home/vbatts

**Shell:** /bin/bash

Such mail.

**Plan:**

OHMAN

```
$> id -Gn
```

```
devel opencontainers docker appc redhat golang slackware
```



# CONTAINERS



(Cite: the internet)

# CONTAINERS



(Cite: The Internet)

# CONTAINER RUNTIMES

## A HISTORY

chroot + unshare (mnt ns) + cgroups

LXC

- first release Aug 2008
- alive and well today
- <https://linuxcontainers.org/>

systemd-nspawn

- first commit in 2011
- alive and well today
- originally just a simple test and debug utility

# CONTAINER RUNTIMES

## A HISTORY

### Docker

- all the things
- 2013 python script became golang
- (not a single good link to the source)

### Imctfy

- 2013
- academic offering
- dead, but still a good example of form over function

### rkt

- Dec 2014
- spec-first design (appC spec)

# CONTAINER RUNTIMES

## A HISTORY

### libct

- from odis/openVZ folks
- C library of container helpers (not docker or lxc related)

### libcontainer (now runc)

- 2014
- to have something go lang native, rather than shell-out to lxc
- (eventually meant more shelling-out)
- the original OCI runtime

### lxd

- 2014
- container manager built on lxc
- (came after Docker made asks of lxc, then abandoned it for libcontainer)

# CONTAINER RUNTIMES

## A HISTORY

### OCI Specification

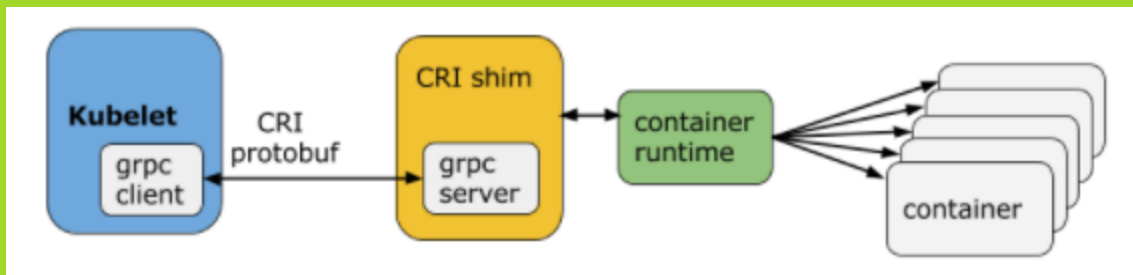
- 2015
- (image and runtime) v1.0.0 in 2017
- distribution-spec began 2018 (formerly docker registry API)



# CONTAINER RUNTIME INTERFACE

## Kubernetes

- 2014
- was (is?) entirely too docker-centric
- with rkt it was clear there were big changes needed
- Enter CRI (2016)
  - gRPC interface for ImageService and RuntimeService



([source](#))

## migration via dockershim

- completely switched to docker via CRI in v1.8

# CONTAINER RUNTIME INTERFACE

## cri-o (originally oci-d)

- kubernetes only use-case
- OCI images and runtime
- primarily called runc, but is flexible to call other runtimes

## containerd

- primarily docker use-case, but includes a CRI shim
- OCI and Docker images

## alibaba/pouch

- swiss arm knife of sorts
- uses kvm pieces
- supports 2.6.32+ kernel
- imports from runc, docker, p2p backend, and expose a CRI socket

# CONTAINER RUNTIME INTERFACE

## cri-tools

- `crictl`
- the cli for working or debugging directly with CRI layer

# OCI RUNTIMES

## [opencontainers/runc](#)

- 2014
- to have something go lang native, rather than shell-out to lxc
- (eventually meant more shelling-out)
- the original OCI runtime

## [oracle/railcar](#)

- 2017
- an OCI runtime
- written in rust

## [kata-containers](#) (formerly clear containers and hyperV)

- mixed years
- an OCI runtime
- thinVM (qemu with custom machine type)

# OCI RUNTIMES

## nabla-containers (IBM research)

- has an OCI runtime
- unikernel approach
- requires tailored container images with their executor

## nvdiA fork of runc

- inherently an OCI runtime
- exposes GPU specific configuration

## google/gvisor (`runsc`)

- has an OCI runtime
- syscall emulation layer
- uses a bit of kvm
- feels a bit like thrown over the wall

## windows/hcsshim (`runhcs`)

- has an OCI runtime
- windows native container API

# OCI RUNTIMES

- [giuseppe/crun](#)
  - thin OCI runtime written in C
- [projectatomic/bwrap-oci](#)
  - OCI runtime wrapper around bubblewrap
- [vbatts/nspawn-oci](#)
  - OCI runtime wrapper around systemd-nspawn
- [wking/ccon](#)
  - OCI pet project of community member

systemd-nspawn native support ([pr9762](#))

- preferable than wrapper
- still has a couple of pieces missing like hooks and compatible cli

# NON-DOCKER OPTIONS

## umoci

- utility for working with OCI container images
- unpack, modify, repack

## skopeo

- initially just remote inspect images
- now copies (local  $\leftrightarrow$  remote) and translates formats

## buildah

- drop-in for `docker build`, as well as shell subcommands
- direct invocation
- now supports non-root builds

## podman

- easy alias for `docker`
- direct invocation
- now supports non-root

VINCENT BATTS

@VBATTS | VBATTS@REDHAT.COM

[GITHUB.COM/VBATTS/TALKS](https://github.com/vbatts/talks)

THANKS!