

# containerd, What does it mean for me ?



Scott Coulton  
Principal Software Engineer  
Puppet



@scotty-c



@scottcoulton



My other computer  
is Amazon EC2  
[aws.amazon.com](http://aws.amazon.com)



# This Talk

What we will cover

# Agenda

## **What is containerd?**

Why is ContainerD a critical part of the container ecosystem ?

## **What upstream projects use containerd**

How is containerD changing the way we build container products

## **Tying it all together in a real world example**

Let's explore what we can achieve

# What is ContainerD?

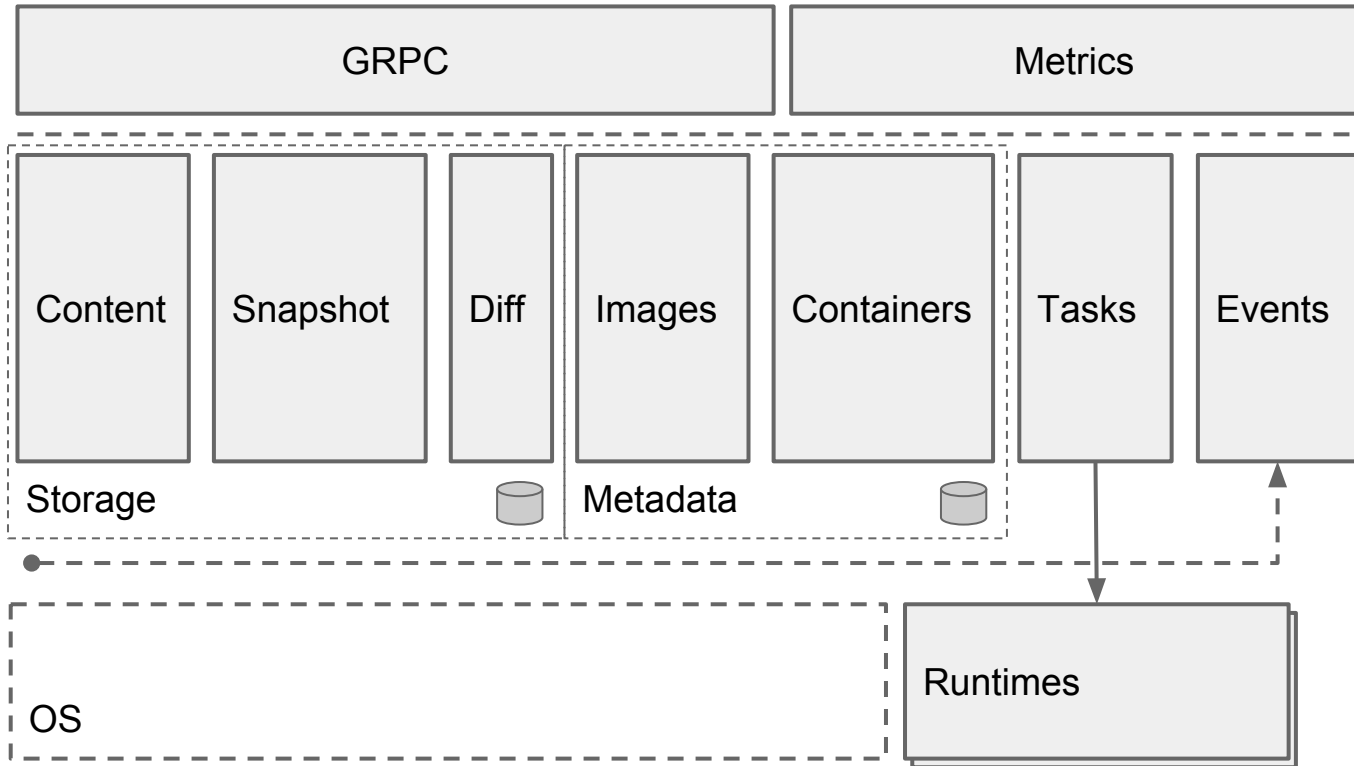
“ContainerD is an industry-standard core container runtime with an **emphasis** on simplicity, robustness and portability.

**containerd** can manage the complete container lifecycle of its host system: image transfer and storage, container execution and supervision, low-level storage and network attachments.”

# Let's breakdown **containerd**

- CTR (ContainerD CLI)
- A daemon exposing gRPC API over a local UNIX socket
- Protobuf specs between components

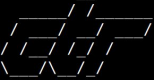
# Let's breakdown containerd





# Let's breakdown containerd

```
NAME:
  ctr -



containerd CLI

USAGE:
  ctr [global options] command [command options] [arguments...]

VERSION:
  v1.1.0

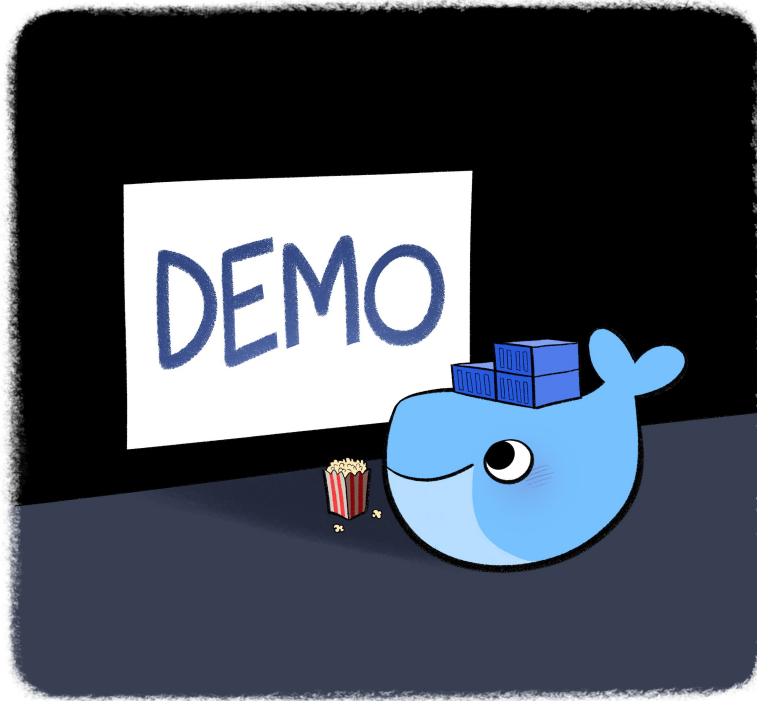
COMMANDS:
  plugins, plugin      provides information about containerd plugins
  version              print the client and server versions
  containers, c, container  manage containers
  content              manage content
  events, event        display containerd events
  images, image, i     manage images
  namespaces, namespace  manage namespaces
  pprof                provide go lang pprof outputs for containerd
  run                  run a container
  snapshots, snapshot  manage snapshots
  tasks, t, task       manage tasks
  shim                 interact with a shim directly
  cri                  interact with cri plugin
  help, h              Shows a list of commands or help for one command

GLOBAL OPTIONS:
  --debug                enable debug output in logs
  --address value, -a value  address for containerd's GRPC server (default: "/run/containerd/containerd.sock")
  --timeout value         total timeout for ctr commands (default: 0s)
  --connect-timeout value  timeout for connecting to containerd (default: 0s)
  --namespace value, -n value  namespace to use with commands (default: "default") [$CONTAINERD_NAMESPACE]
  --help, -h              show help
  --version, -v           print the version
```

# containerd gives us

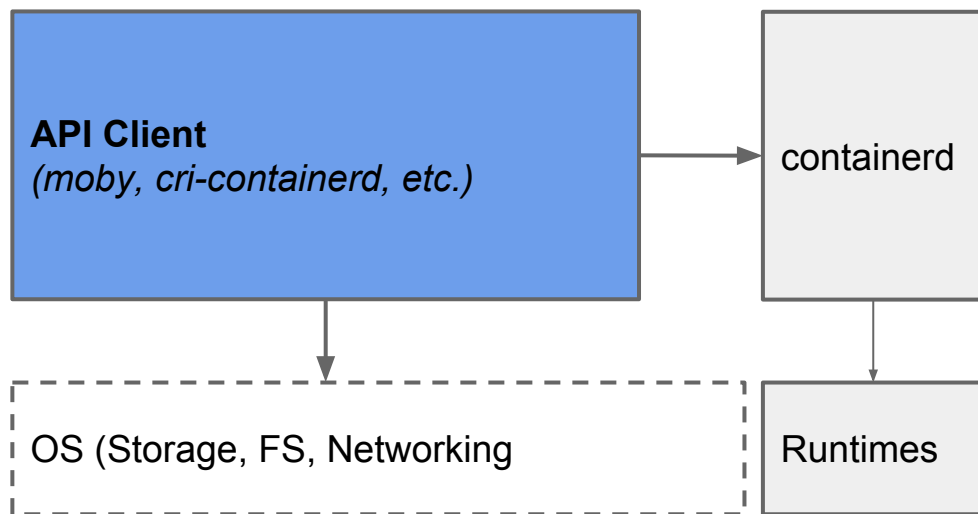
- OCI Image Spec support
- OCI Runtime Spec support (aka runC)
- Image push and pull support
- Network primitives for creation, modification, and deletion of interfaces
- Management of network namespaces containers to join existing namespaces

# Lets see it in action

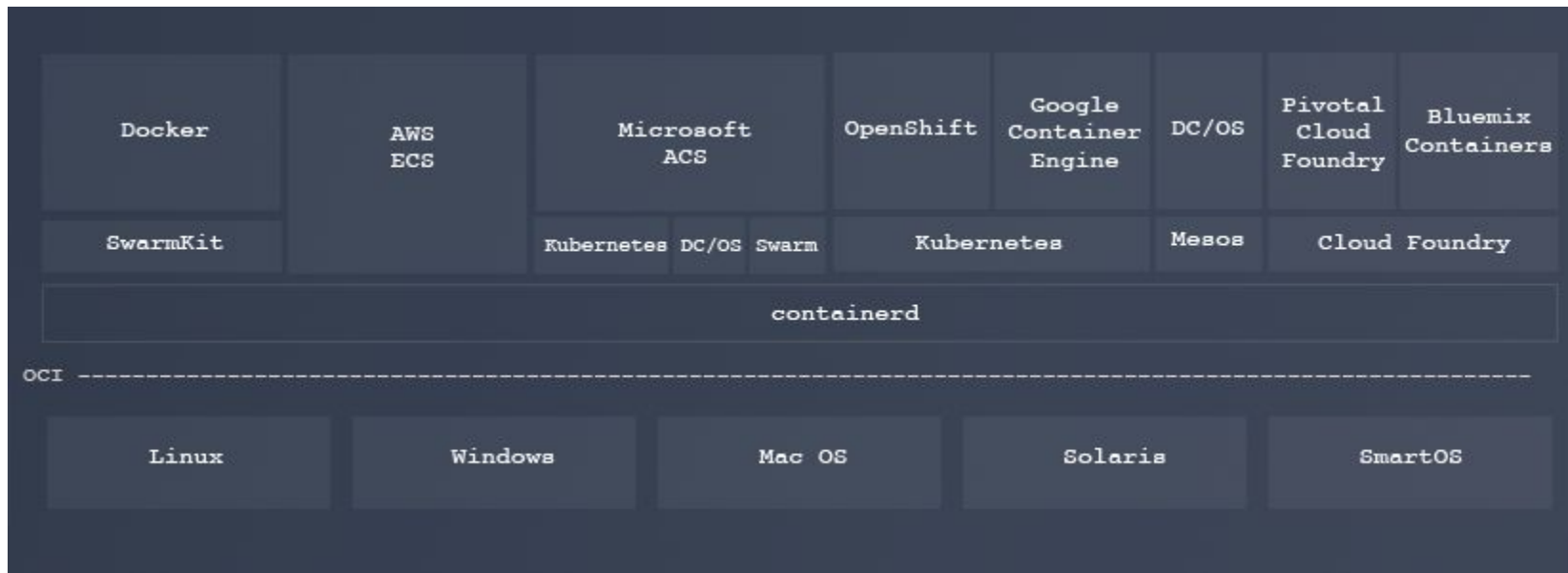


# What upstream projects use containerd

# How does containerd architecture allow other projects to interact with it



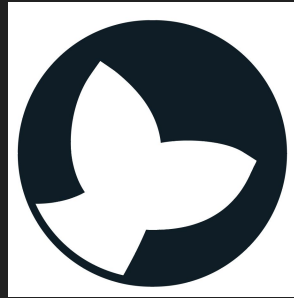
# How does containerd architecture allow other projects to interact with it



# containerd is used in

- Moby
- Moby (LinuxKit)
- Kubernetes
- LCOW (Linux containers on Windows)

# Moby Project

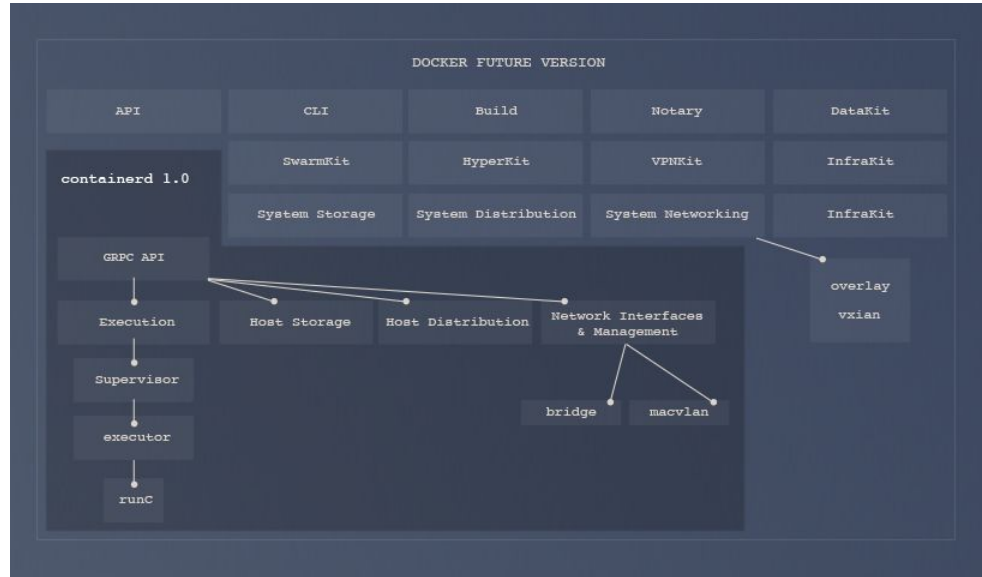




# The Moby project is made up of

- The Docker Engine
- SwarmKit
- HyperKit
- RunC
- Also LinuxKit (I want to cover that separately)

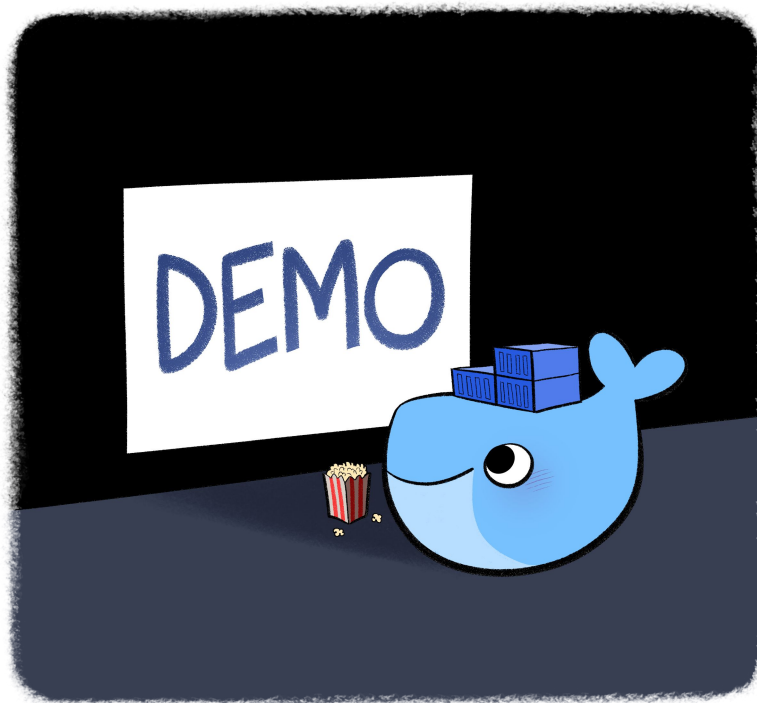
# How does containerd interact with Moby



# How is containerd different from Docker



# Lets see it in action !



# LinuxKit

# What is LinuxKit ?

So what does LinuxKit give us ?

- Lean OS. Minimal size, minimal boot time
- 4.9 Kernel
- Allows you to run any container runtimes
- Batteries included but can be replaced
- All system services are containers

# What is LinuxKit ?

Why is it different to a traditional OS ?

- Smaller attack surface
- Immutable infrastructure
- Sandboxed system services
- Specialized patches and configurations
- You have full control over the build
- The configuration is all yaml

# How is containerd integrated with LinuxKit ?

- Runs at init time
- Is defined in your configuration yaml

`init:`

- `linuxkit/init:63eed9ca7a09d2ce4c0c5e7238ac005fa44f564b`
- `linuxkit/runc:2649198589ef0020d99f613adaeda45ce0093a38`
- `linuxkit/containerd:18eaf72f3f4f9a9f29ca1951f66df701f873060b`
- `linuxkit/ca-certificates:3344cdca1bc59fdfa17bd7f0fcfbf491b9dbaa288`



**Every** containerd demo we have done  
has run in **LinuxKit**





# Kubernetes



# How does **containerd** and **Kubernetes** interact ?



# To follow this project

<https://github.com/containerd/cri>

containerd / cri

Watch 50 Star 276 Fork 76

Code Issues 42 Pull requests 6 Projects 0 Wiki Insights

Containerd Plugin for Kubernetes Container Runtime Interface

containerd container-runtime-interface kubernetes

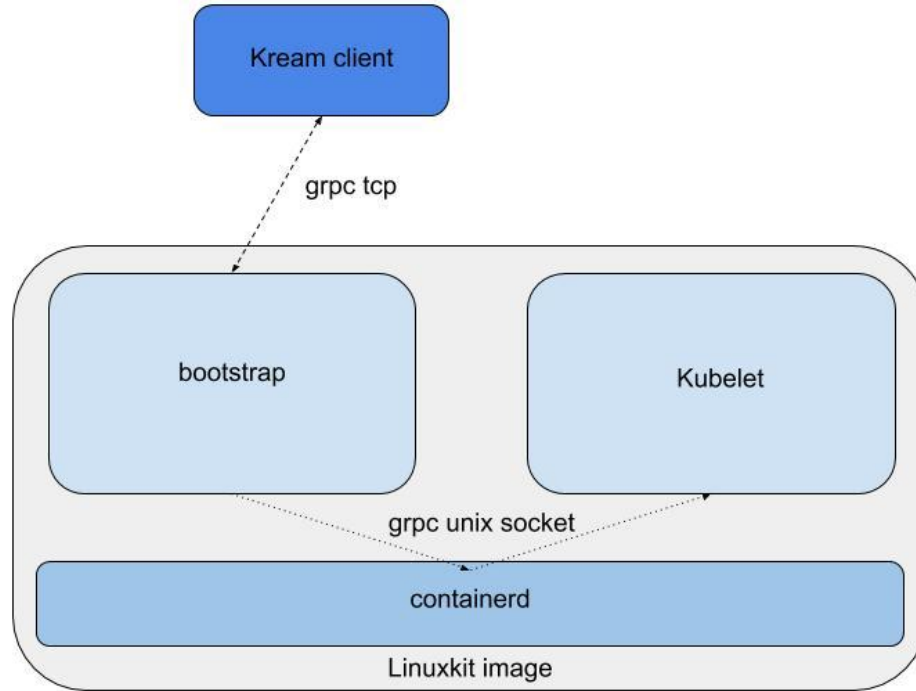
1,117 commits 4 branches 12 releases 34 contributors Apache-2.0

Branch: master New pull request Create new file Upload files Find file Clone or download

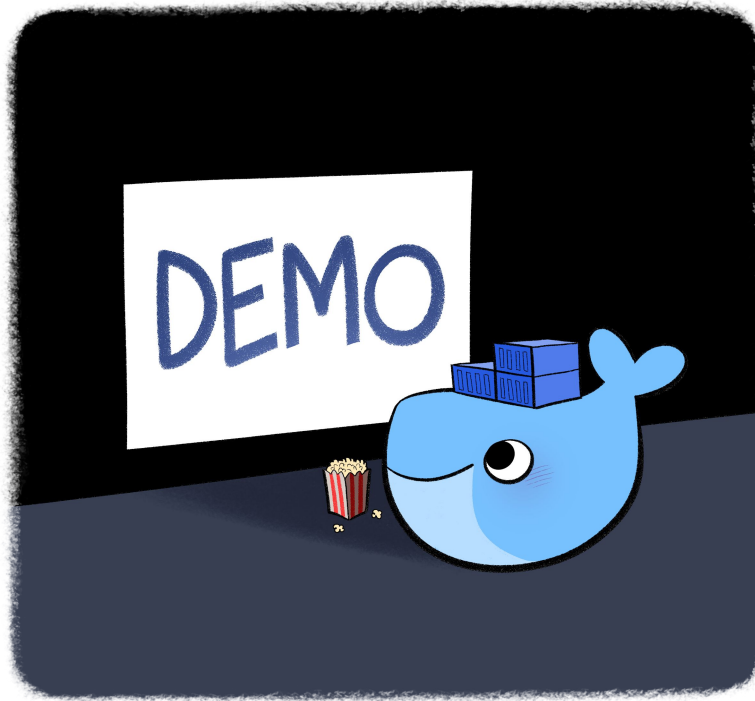
cli	Rename all variables to remove "cricontainerd".	a month ago
cluster	Add 'unix://' prefix for socket addresses used by CRI remote client.	11 days ago
cmd	Change for new containerd.	a month ago
contrib	Add 'unix://' prefix for socket addresses used by CRI remote client.	11 days ago
docs	Down containerd binaries from official release.	7 days ago
hack	Update cri-tools to fix 'cricri logs' output.	5 days ago
integration	Add 'unix://' prefix for socket addresses used by CRI remote client.	11 days ago
pkg	Always mount sysfs as 'rw'.	5 days ago
test	Add log level support.	13 days ago
vendor	Update containerd before release.	8 days ago
.gitignore	initial makefile (#7)	a year ago
.travis.yml	start tracking golang updates for 1.10.x	15 days ago
CONTRIBUTING.md	update readme after move to containerd	4 months ago
LICENSE	Squashed 'hack/repro-infra' content from commit 684e550	a year ago
Makefile	Fix for kube-up.sh and update several documents.	a month ago
OWNERS	Add new approvers/reviewers to the OWNERS file.	5 months ago
README.md	Add 'unix://' prefix for socket addresses used by CRI remote client.	11 days ago

**Let's tie it all together in a real world  
example**

# Enter kream v2



# Lets see it in action !





# Questions.

