



Podman, Buildah and Quarkus

The Latest in Linux Containers Technology

Daniel Oh

Open Source Summit Japan 2019



About Me



Daniel Oh

- Principal Technical Product Marketing Manager at Red Hat
 - Cloud Native App Development
 - Agile & DevOps practices
- CNCF Ambassador
- Java Developer
- Opensource.com Moderator
- Public Speaker & Writer



doh@redhat.com



@danieloh30



danieloh30

Container tools landscape is changing. Why?



Since Open Container Initiative (OCI) there are several new projects
What needs do these projects address?
What are these projects and when should I use them?
What specific security concern does each address?



Early concerns with Docker

Since the early days enterprise users of Docker had concerns

- Build requires a “big fat” daemon on every host
- Regression for integration with container platforms
Kubernetes/OpenShift
- Build has secret handling issues
- Root/privileged concerns at runtime
- Root/privileged concerns with daemon
- Build requires a running container



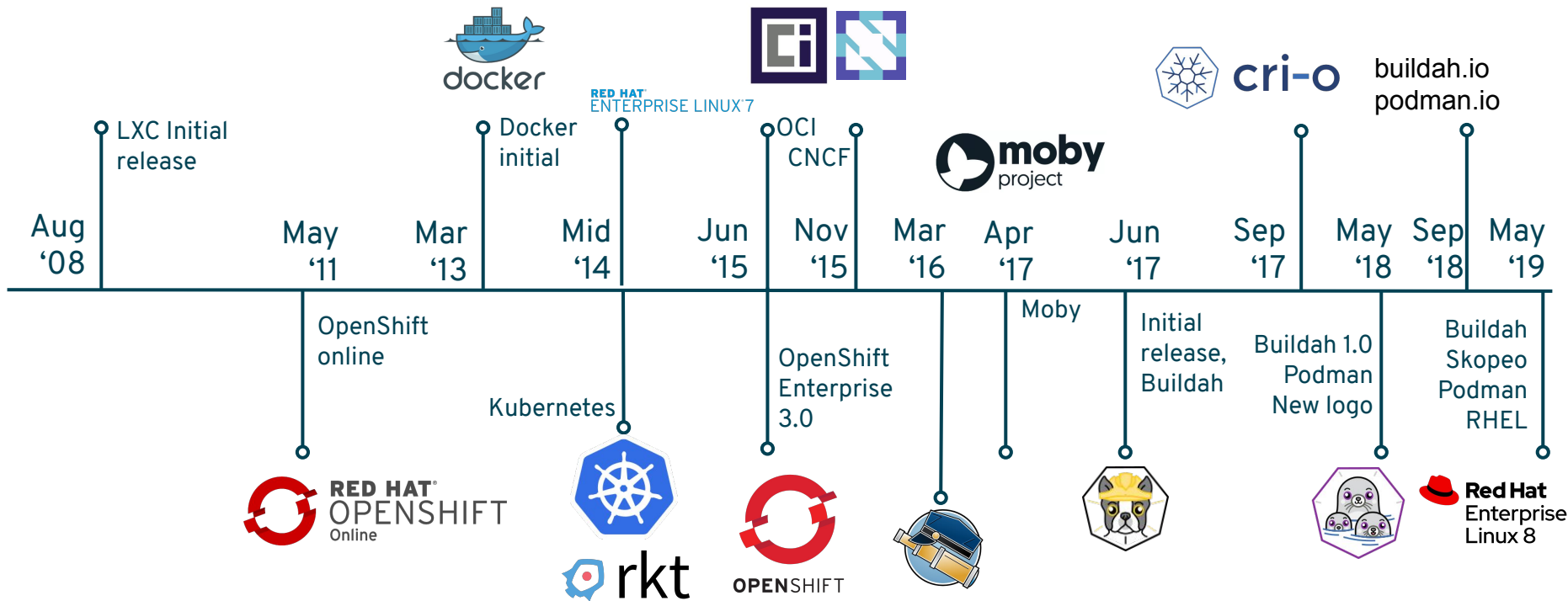


OPEN CONTAINER
INITIATIVE

- Docker, Red Hat et al. June 2015
- Two specifications
 - Image format
 - How to package an OCI Image with sufficient information to launch the application on the target platform
 - Runtime
 - How to launch a “filesystem bundle” that is unpacked on disk
- Version 1.0 of each released July 19th 2017
- Distribution spec started in April, 2018.

Containers are Linux

Container innovation continues



How did Docker change containers?

Docker Daemon

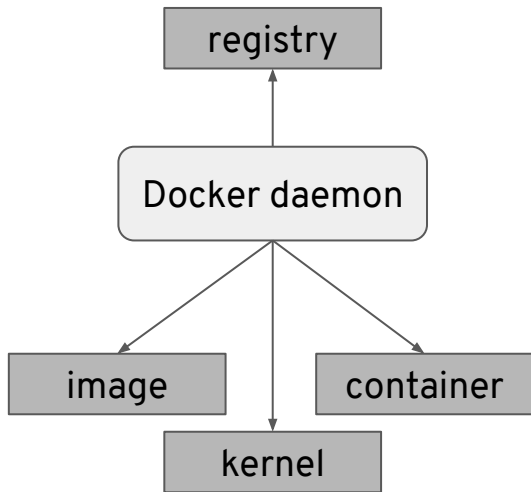
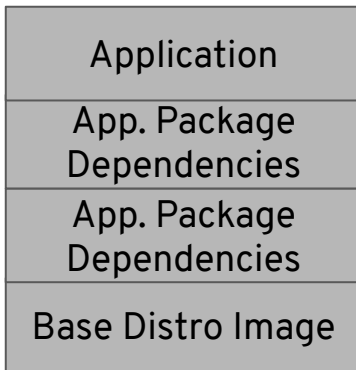
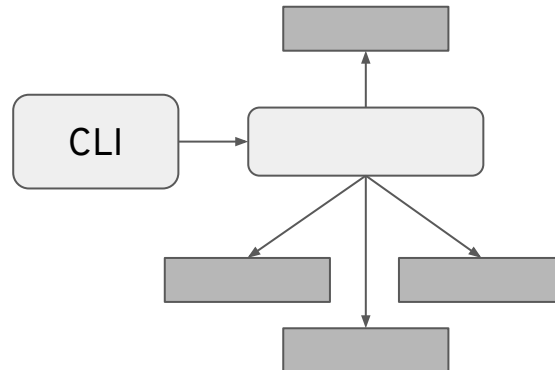


Image Layers



Docker CLI





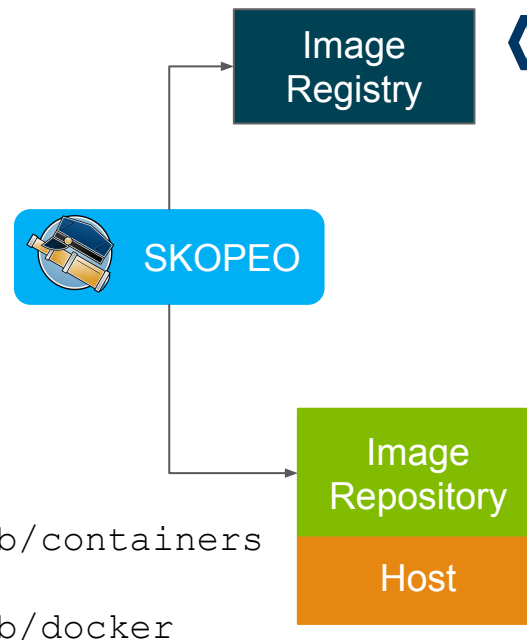
Skopeo.
The first break
away.

IMAGE COPY WITH SKOPEO

- Built for interfacing with Docker registry
- CLI for images and image registries
- Rejected by upstream Docker `_(ツ)_/`
- Allows remote inspection of image metadata
- no downloading
- Can copy from one storage to another

SECURITY FEATURES

Share securely
 No daemon
 Inspect remote images
 No pulling potentially malicious images
 Non-root copy. Bridge between registries.





Podman.
The daemonless
client for
developers and
beyond.

The new container CLI

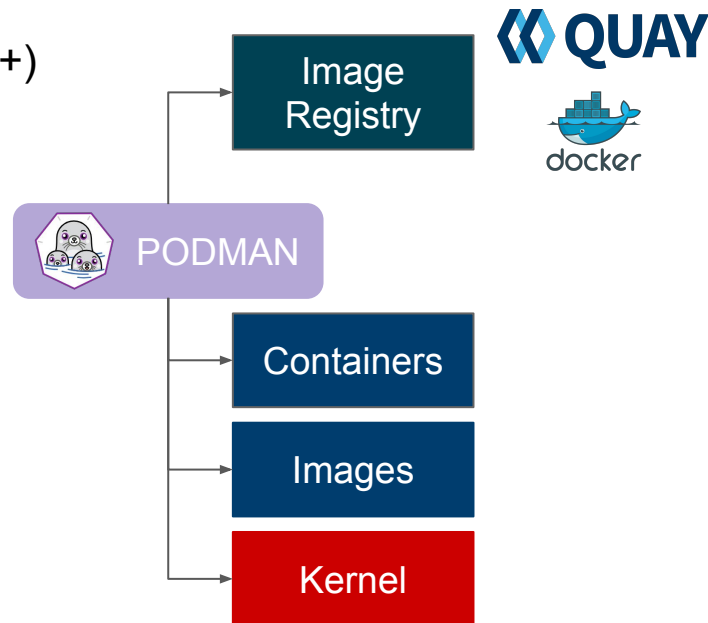


podman

- @ podman.io
- Client only tool, based on the Docker CLI. (same+)
- No daemon!
- Storage for
 - Images - `containers/image`
 - Containers - `containers/storage`
- Runtime - `runc`
- Shares state with CRI-O and with Buildah!

SECURITY FEATURES

Run and develop securely
 No daemon
 Run without root
 Isolate with user namespaces
 Audit who runs what





podman

Developer's experience

- Provides a familiar command line experience compatible with the Docker CLI
- Great for running, building, and sharing containers outside of Kubernetes/OpenShift
- **Build and run containers as non-root (enhanced user namespaces)**
- Can be wired into existing infrastructure where the docker daemon/cli are used today
- Use existing Dockerfiles
- Simple command line interface, **no client-server architecture**
- Docker compatible health checks

Podman Demo



podman

But there's more : podman pod

Pods are a group of one or more containers sharing the same network, pid and ipc namespaces.

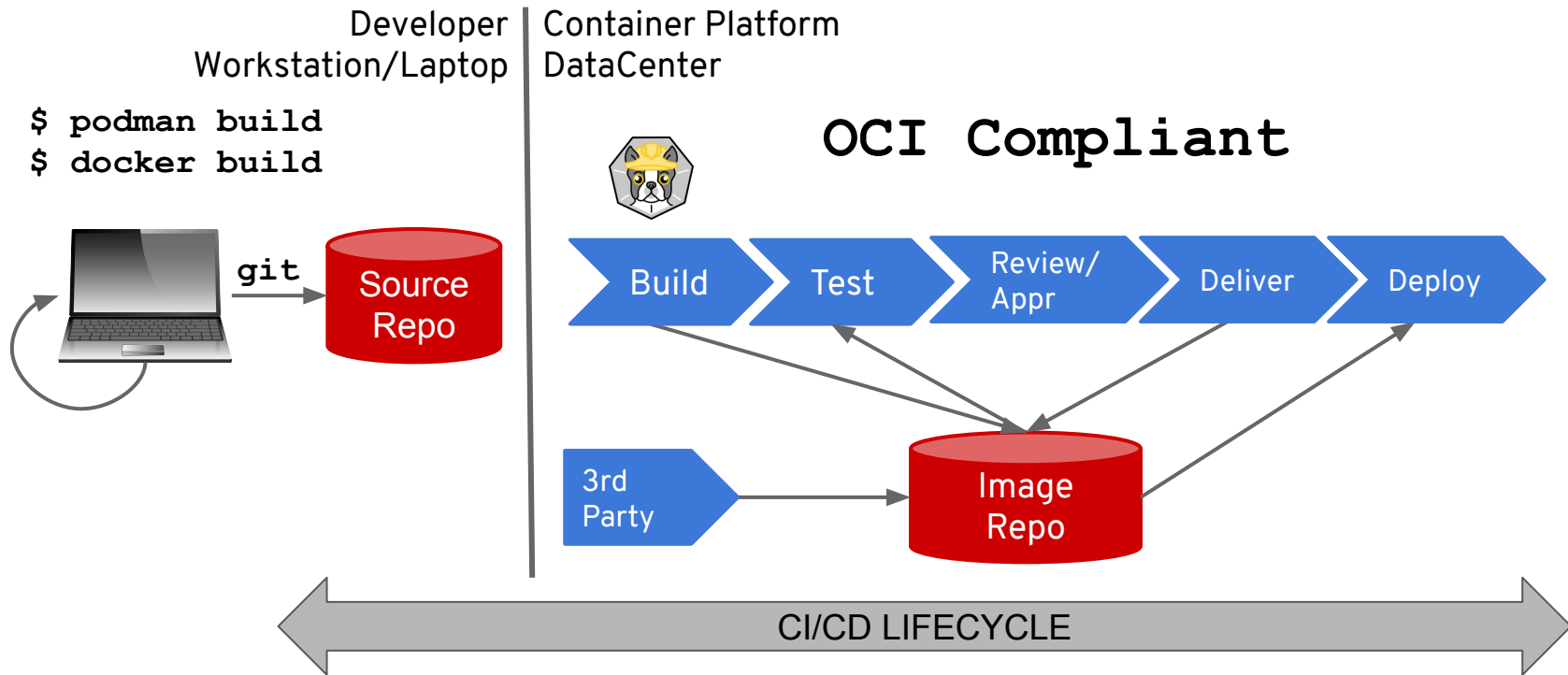
create	Create a new empty pod
exists	Check if a pod exists in local storage
inspect	Displays a pod configuration
kill	Send the specified signal or SIGKILL to containers in pod
pause	Pause one or more pods
ps	List pods
restart	Restart one or more pods
rm	Remove one or more pods

start	Start one or more pods
stats	Display a live stream of resource usage statistics for the containers in one or more pods
stop	Stop one or more pods
top	Display the running processes of containers in a pod
unpause	Unpause one or more pods



Buildah.
The secure
container
builder.

The separation of concerns





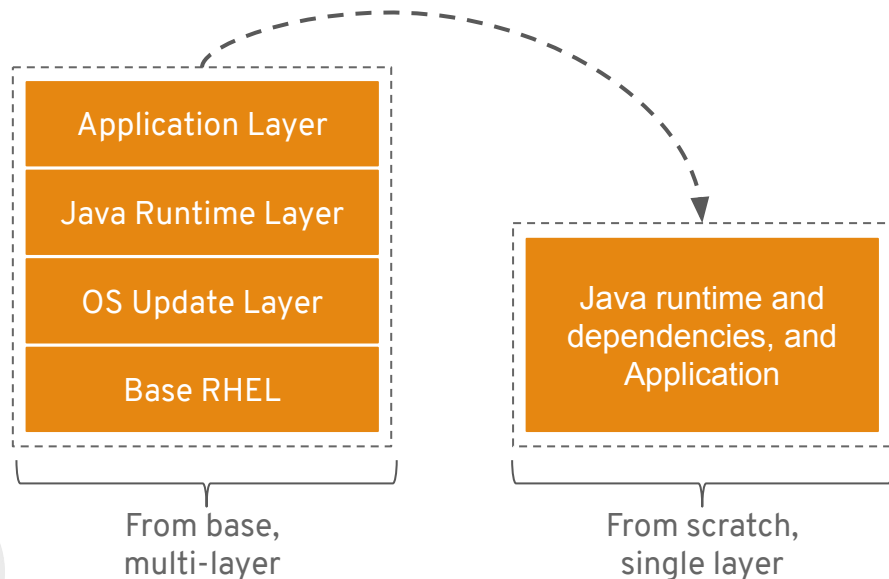
buildah

Why use Buildah?

- Now buildah.io
- Builds OCI compliant images
- No daemon - no “docker socket”
- Does not require a running container
- Can use the host’s user’s secrets.
- Single layer, from scratch images are made easy and it ensures limited manifest.
- If needed you can still maintain Dockerfile based workflow

SECURITY FEATURES

- Build securely
- No daemon
- Shrink the attack surface
- Fine-grained control of the layers
- Run builds isolated
- Better secret management



Buildah Demo



What does Buildah do?

buildah from - Build up a container root filesystem from an image or *scratch*.

buildah config - Adjust defaults in the image's configuration blob.

buildah run - Run a command in the container's filesystem using *runc*.

NOT like docker run. Like Dockerfile RUN.

buildah mount - Mount the container's root filesystem on the host.

buildah commit - Commit container's changes to a new image.



What does Buildah do?

buildah push - Push images to registries (Quay etc.) or a local *dockerd* instance

buildah build-using-dockerfile (a.k.a. *buildah bud*) - Build images using a Dockerfile for instructions

buildah unmount - Oh, it also unmounts container filesystems

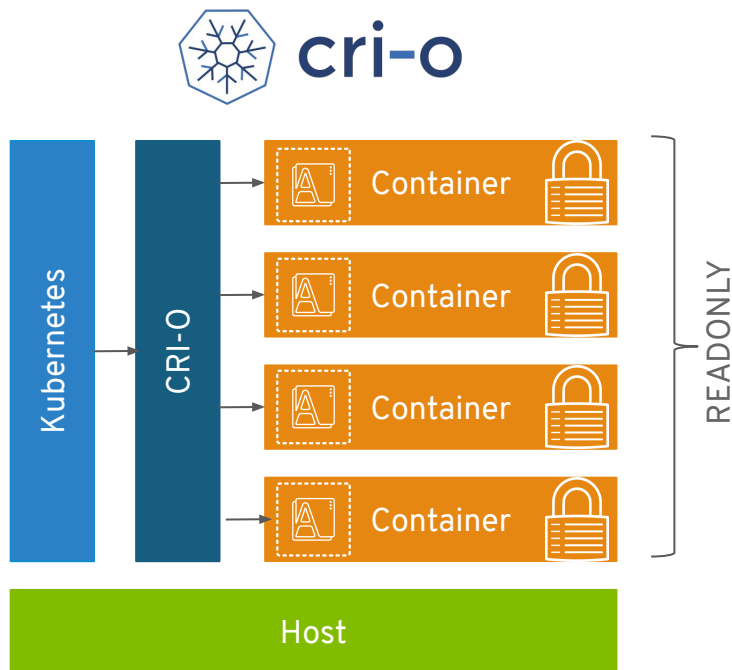
Provide a library API that's used by the CLI

Share libraries and on-disk storage with CRI-O



CRI-O. The OCI runtime abstraction for Kubernetes

OCI AND CRI-O



- A Kubernetes thing
- Now part of CNCF! (April 8th)
- OCI daemon
- Implements Kubelet Container Runtime Interface (CRI)



SECURITY FEATURES

Run securely in a production cluster
 No daemon
 Read-only containers
 Enable fewer capabilities
 User namespaces
 FIPS mode support



Which **app** will
be the best for
the Linux
Containers?



QUARKUS

Supersonic Subatomic Java

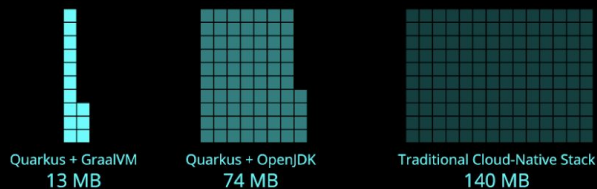
A Kubernetes Native Java stack tailored for GraalVM & OpenJDK HotSpot, crafted from the best of breed Java libraries and standards

Container First

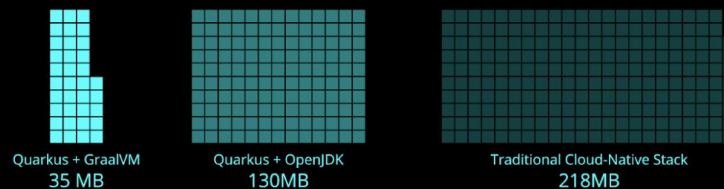
MEMORY AND BOOT + FIRST RESPONSE TIME

Memory (RSS) in Megabytes

REST



REST + JPA



Boot + First Response Time in Seconds

REST



REST + JPA



QUARKUS

A stack to write Java apps



Cloud Native,



Microservices,



Serverless

Quarkus Demo

Benefit No. 1: Developer Joy

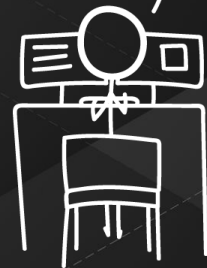
A cohesive platform for optimized developer joy:

- Based on standards, but not limited
- Unified configuration
- Zero config, live reload in the blink of an eye
- Streamlined code for the 80% common usages, flexible for the 20%
- No hassle native executable generation

WAIT.
SO YOU JUST SAVE IT,
AND YOUR CODE IS RUNNING?
AND IT'S JAVA?!



I KNOW, RIGHT?
SUPERSONIC JAVA, FTW!

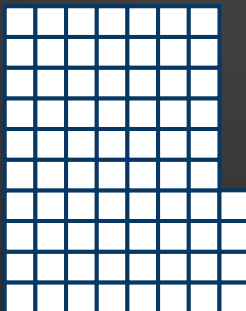


Benefit No. 2: Supersonic Subatomic Java

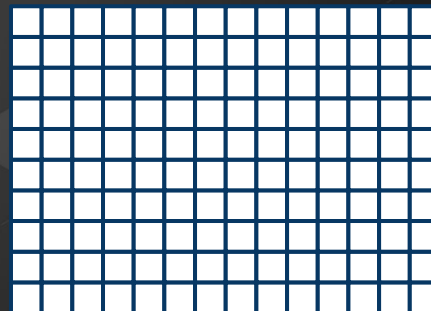
REST



Quarkus + GraalVM
13 MB



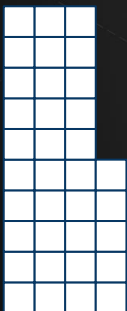
Quarkus + OpenJDK
74 MB



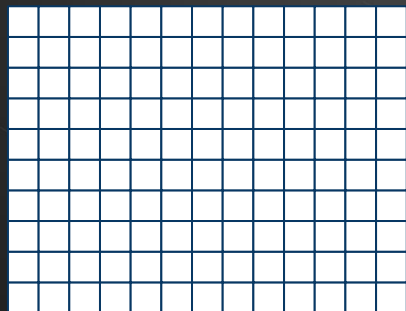
Traditional Cloud-Native Stack
74 MB

Benefit No. 2: Supersonic Subatomic Java

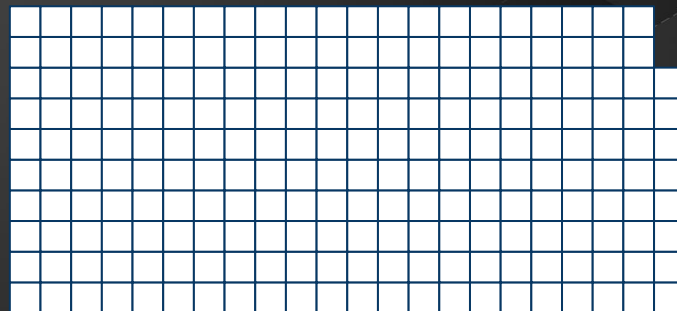
REST + CRUD



Quarkus + GraalVM
35 MB



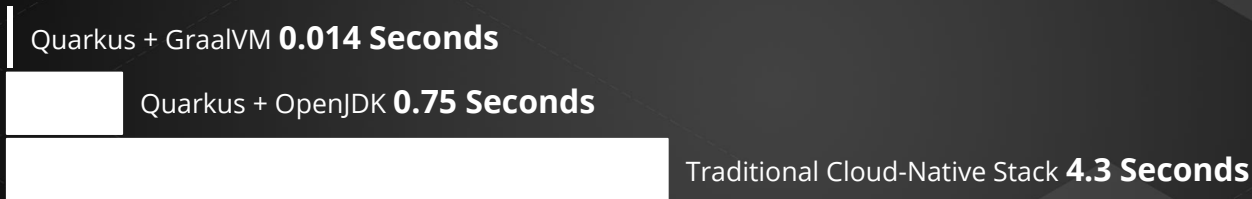
Quarkus + OpenJDK
130 MB



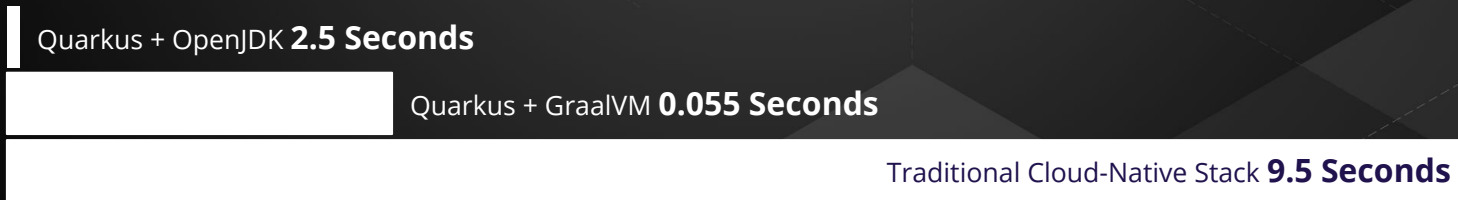
Traditional Cloud-Native Stack
218 MB

Benefit No. 2: Supersonic Subatomic Java

REST



REST + CRUD



Benefit No. 3: Unifies Imperative and Reactive

```
@Inject
SayService say;

@GET
@Produces(MediaType.TEXT_PLAIN)
public String hello() {
    return say.hello();
}
```

```
@Inject @Stream("kafka")
Publisher<String> reactiveSay;

@GET
@Produces(MediaType.SERVER_SENT_EVENTS)
public Publisher<String> stream() {
    return reactiveSay;
}
```

- Combine both Reactive and imperative development in the same application
- Inject the EventBus or the Vertx context
- Use the technology that fits your use-case

Benefit No. 4: Best of Breed Frameworks & Standards



Eclipse Vert.x



Hibernate



RESTEasy



Apache Camel



Eclipse MicroProfile



Netty



Kubernetes



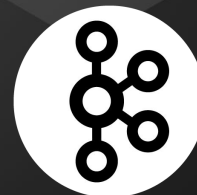
OpenShift



Jaeger



Prometheus



Apache Kafka



Infinispan

Where do I find
these projects
in Red Hat
products?

WHERE CAN I FIND THESE AWESOME PROJECTS IN RED HAT PRODUCTS?



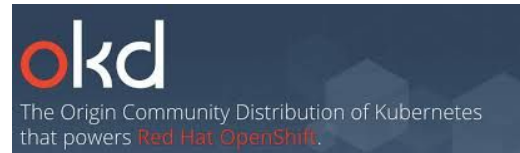
SECURING THE PLATFORMS

Red Hat chose to move away from Docker to Podman, Buildah and CRI-O in order to provide a more secure environment for containers on both OpenShift Container Platform and RHEL



SMART LIGHT CONTAINERS TOOL

The container-tools package installs Podman, Buildah and Skopeo together. Users can use these in root or user namespace.



CONTINUOUS BUILDING, SECURE RUNTIME

Red Hat OCP has been using Skopeo for efficiency for a number of years. OCP 4 moves to using Buildah by default for building container images. CRI-O is the default container runtime. Use Podman for helping to debug pods and containers.



Open source
wins. Questions
inspire
community
innovations.



Why do we have to pull down a container just to inspect it?



Could we decouple kubernetes from the container runtime?



Is it possible to build containers on a cluster without having to install and run a daemon?



How will we design, use, debug containers on the cluster if we don't have client tools because we don't have Docker?



Getting Started

- Download Podman today
 - Package name `podman`
- It won't clash with your existing Docker
- If you feel more adventurous download Buildah too
 - Package name `buildah`
- Or all with `:container-tools`
- Lots of demos and tutorials are available
- Contribute to the projects! (Next page)



podman



buildah

Where can I learn more?



Buildah

<https://github.com/containers/buildah>

<https://buildah.io/>

Podman

<https://github.com/containers/libpod>

<https://podman.io/>

Skopeo

<https://github.com/containers/skopeo>

Other useful links

<https://github.com/opencontainers/runc>

<https://developers.redhat.com/blog/2019/02/21/podman-and-buildah-for-docker-users>

<https://www.katacoda.com/courses/containers-without-docker>



QUARKUS


 <https://quarkus.io>

 <https://quarkusio.zulipchat.com>


 [@QuarkusIO](https://twitter.com/QuarkusIO)

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

 [linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)

 [facebook.com/redhatinc](https://www.facebook.com/redhatinc)

 [youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)

 twitter.com/RedHat

 [linkedin.com/in/williamhenry](https://www.linkedin.com/in/williamhenry)

 twitter.com/ipbabble