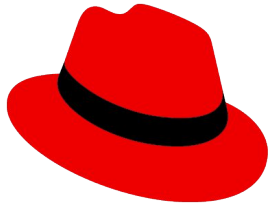
Red Hat Summit

Connect

Harness the power of containers to build, automate and standardise your OS

Red Hat



Red Hat

Martin Skøtt

Senior Solution Architect
Red Hat

Outcomes

What does image mode fix today?

Less risk

Reduce the risk associated with updates with atomic transactions and rollbacks

Better builds

Improve the composability and repeatability of standard builds through layering

Move faster

Increase the speed of experimentation

Streamline process

Simplify end to end management with a single process for OS and applications

Use Cases

Where does image mode fit today?

AI/ML Stacks

Perfectly version app dependencies from kernel, GPU & accelerator drivers, frameworks, runtimes, etc

1:1 App/Host

Manage the OS AND app as a single unit

Edge appliances

Easily manage a fleet of systems with registries and auto-updates

Standalone container hosts

Use common toolchains and pipelines to build containerized applications and the hosting OS

Image mode for RHEL

A container-native workflow for the life cycle of a system

```
FROM rhel9/rhel-bootc:latest

RUN dnf install -y [software]
[dependencies] && dnf clean all

ADD [application]
ADD [configuration files]

RUN [config scripts]
```

Tech Preview

Build

A *bootc* base image & container file is all that's needed to describe a system, applications, and dependencies. Use your existing container tools or pipelines to quickly create and test images.

Deploy

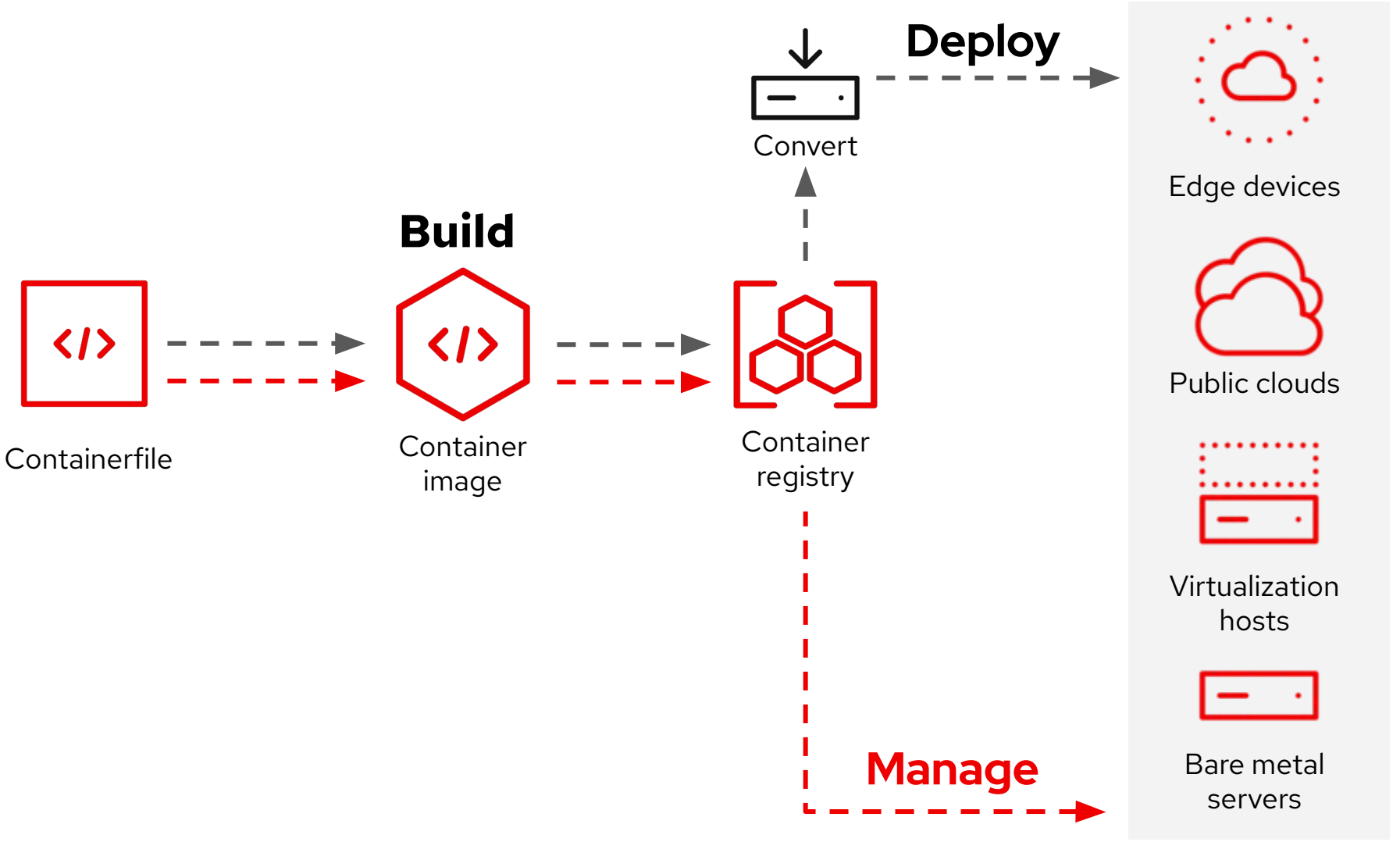
Easily convert to a VM/cloud image or deploy on bare metal using RHEL's installer. The container image includes full hardware drivers, but not cloud agents by default.

Manage

Designed for modern GitOps & CI/CD driven environments. Systems will auto-update from the container registry by default. More advanced control and automation is available via custom rollouts (e.g. Ansible). Intelligence via Insights and on-prem content curation via Satellite are planned for the future.

Image mode for Red Hat Enterprise Linux

Simple. Consistent. Anywhere.



Red Hat

Tech Preview

Image mode for RHEL

Encapsulate differences in a sequence of builds

```
# Derive standard operating environment
FROM rhel9/rhel-bootc:latest

RUN dnf install -y [system agents]
[dependencies] && dnf clean all

COPY [unpackaged application]
COPY [configuration files]

RUN [config scripts]
```

```
# Derive database server from SOE
FROM corp-repo/corp-soe:latest

RUN dnf install -y [database]
[dependencies] && dnf clean all

COPY [configuration files]

RUN [config scripts]
```

Tech Preview

registry.redhat.io/rhel9/rhel-bootc:9.4

The RHEL bootc image is available in technology preview

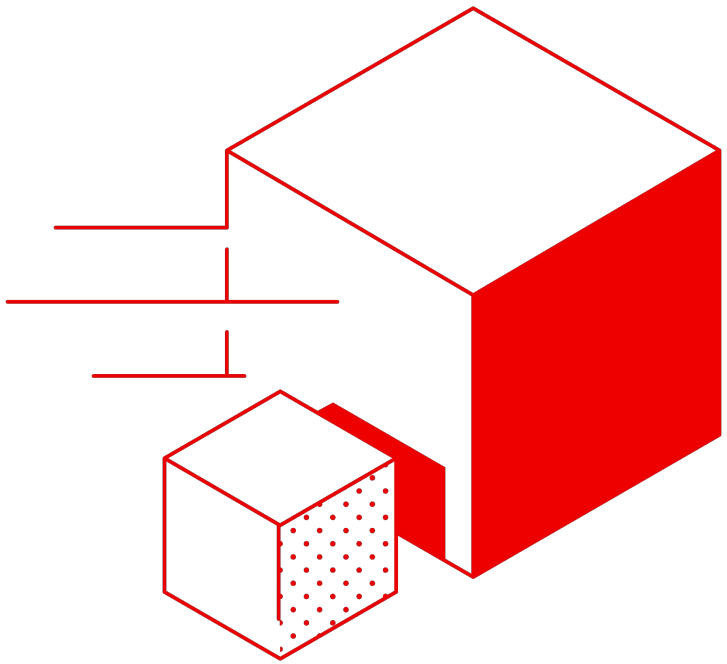


Image Specs:

- 439 rpms
- ~785M compressed
- ~2.2G on disk

Primary contents:

- systemd, kernel, bootc
- rpm-ostree[1]
- linux-firmware
- NetworkManager
- podman
- python
- Misc CLI tools: jq, sos

No cloud-init or virt agents

Documentation

Red Hat

Filesystem Layout

Similar to previous ostree setups - but better!!

Build Time
Everything is writable. e.g. /usr, /etc, /opt, ...

Run Time
All image content is read only

/var - RW, instance persistence. Not updated post install, e.g. `podman volume /var`

/etc - RW, 3-way merge like RHEL CoreOS. Machine local state (hostname, static IP)

Flexible partitioning & disk layout available via Anaconda

Defaults aimed at 80%

- Provides a balance of immutable updates w/ persistent config, logs, & container images.
- composefs backend offers better security

Powerful Configuration for the 20%

- Transient / and /etc
- RW possible for other directories via bindmounts and symlinks to /var

Tech Preview

bootc

A/B booting of container images

`bootc upgrade`

Download and stage an updated container image.

- Automatic updates on by default. Configurable using bootc-fetch-apply-updates.timer

`bootc rollback`

Rollback to the previous state. Staged updates are discarded

`bootc switch`

Change to a different reference image

`bootc install`

Install container image **`to-disk`** or **`to-filesystem`**

- Man page
- https://github.com/containers/bootc
- https://github.com/containers/podman-desktop-extension-bootc

Tech Preview

podman-bootc

Ergonomic edit-compile-debug cycle

Run a bootable container as easy as: **`podman-bootc run <imagename>`**

Automatically injects a user and starts an SSH session

- **`podman-bootc list`**: List running VMs
- **`podman-bootc ssh`**: Connect to a VM
- **`podman-bootc rm`**: Remove a VM

Available on Fedora and MacOS

github.com/containers/podman-bootc

Red Hat

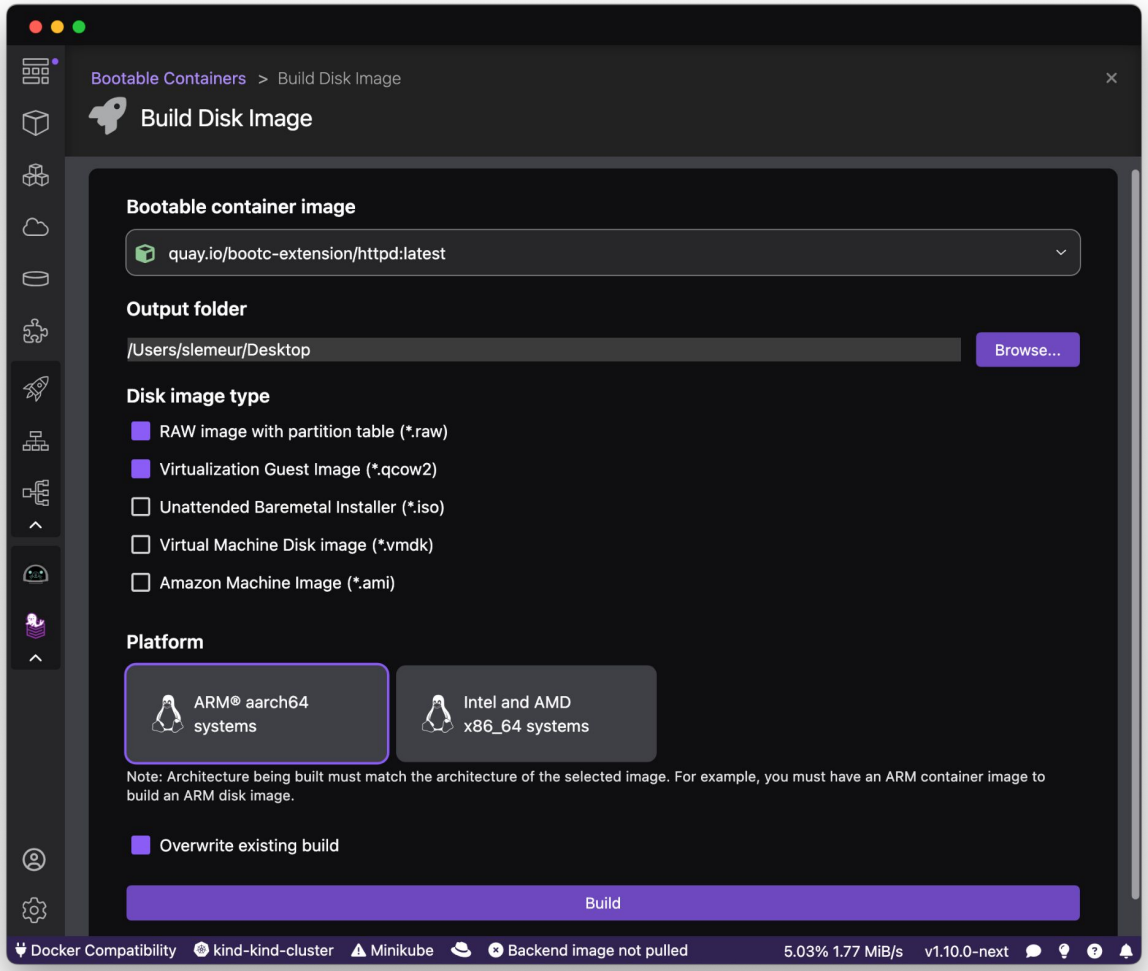
Tech Preview

Bootc image builder

Create **bootable container images** for bare metal to AWS and everywhere in between

`qcow2`	QEMU Disk Images
`ami`	Amazon Machine Images
`raw`	Raw disk image with MBR or GPT partition table
`anaconda-iso`	Unattended installation (USB Sticks / Install-on-boot)
`vmdk`	Virtual Machine Disk Image (vSphere, etc.)

- Designed for and only available as a container image:
  - registry.redhat.io/rhel9/bootc-image-builder
- Available extension for Podman Desktop
  - Build for Intel & Arm architectures



Red Hat

Tech Preview

Install via Kickstart

Deploy container images to bare metal using installation media

```
lang en_US.UTF-8
keyboard us
timezone Etc/UTC --isUtc
text
zerombr
clearpart --all --initlabel
autopart
reboot
user --name=admin-user --groups=wheel
sshkey --username=admin-user "ssh-rsa
AAAAB3Nza……."

ostreecontainer --url quay.io/myimage:latest
```

NEW

Use existing provisioning workflows

- Red Hat Enterprise Linux boot media (isos)
- PXE & HTTP Boot for network based deployments

Kickstart and Anaconda are used for disk layout and select configurations

- `%packages` is ignored
- ostreecontainer will fetch the container image from a registry and write it to disk.

`%pre` and `%post` used for configuration

Red Hat

Join the community!

There is a lot of activity

Upstream projects:

- github.com/containers/bootc
- github.com/containers/podman-bootc
- github.com/osbuild/bootc-image-builder
- github.com/containers/podman-desktop-extension-bootc

Keynote: What if you could boot a container? - DevConf.CZ 2024
(34:55)

Red Hat Summit

Connect

Thank you

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat