



Containers

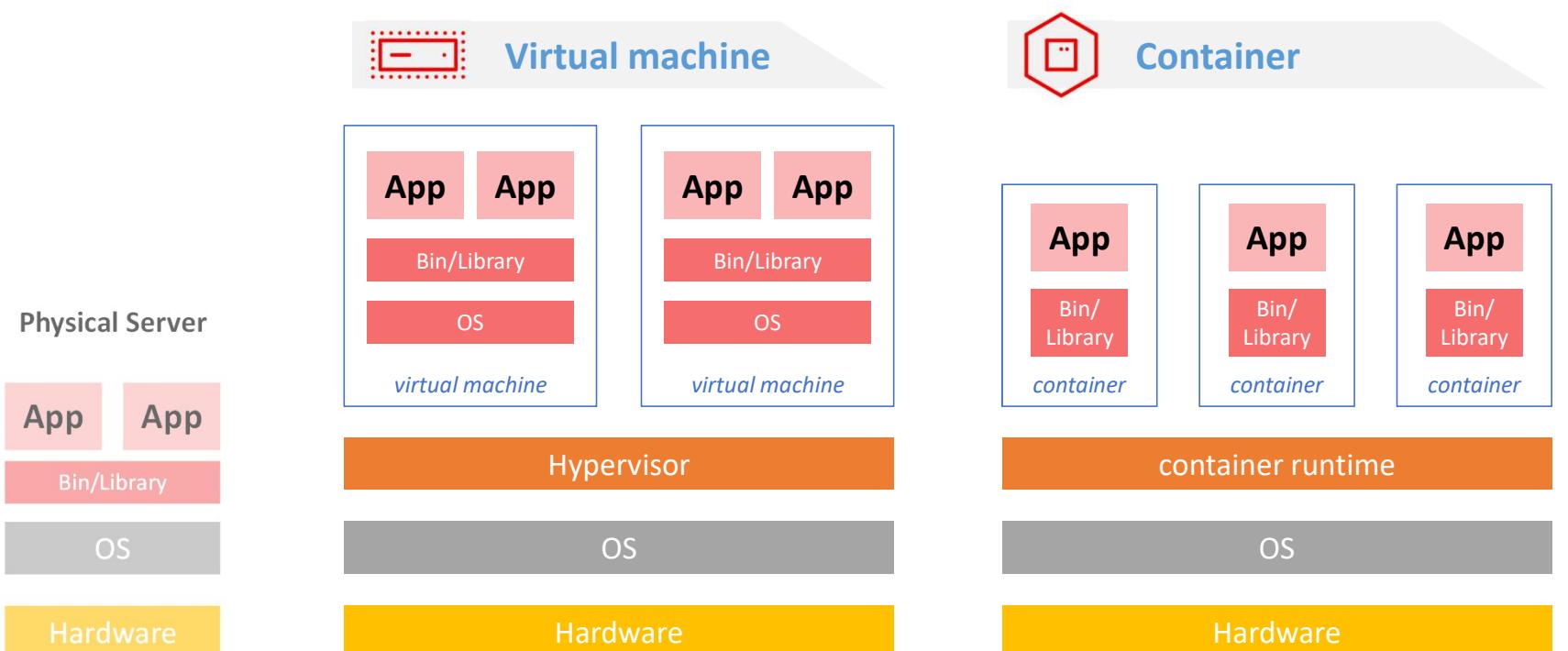


What is a container in a nutshell?

A container is a standard **unit of software** that **packages minimum necessary elements** such as **the application itself, libraries and dependencies required for application execution.**

Differences between containers and virtual machines

A container does not have an OS.



Differences between containers and virtual machines

A container does not have an OS.



virtual machine

||

private room



container

||

spaces separated by dividers



Container Characteristics (1 of 3)

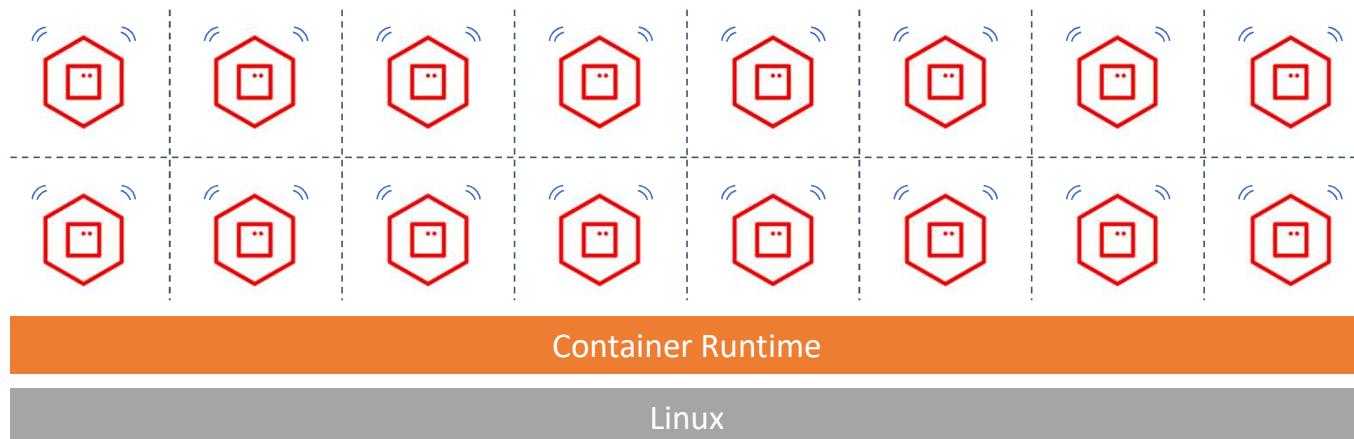
Containers have a number of characteristics.

- ▶ **Runs on Linux**

- Use the features of the Linux kernel.
- Multiple containers can run simultaneously on a single Linux host.

- ▶ **Isolation**

- It shares the kernel of the Linux host, but the containers are isolated and do not conflict with each other.
 - Containers can still communicate each other.

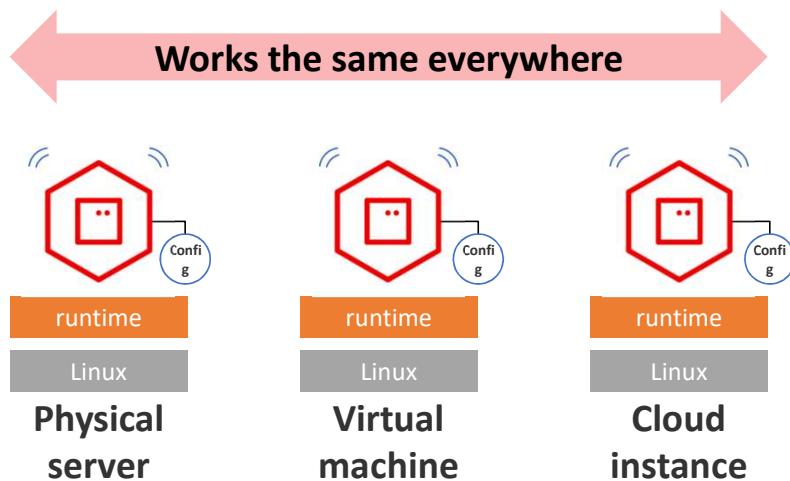


Container Characteristics (2 of 3)

Containers have a number of characteristics.

► Portability

- Works equally well in any environment.
 - The configuration information that depends on the environment is held separately from the container.



► Lightweight

- It has no OS and has only the minimum necessary elements.

► Fast boot

- OS startup time can be omitted.

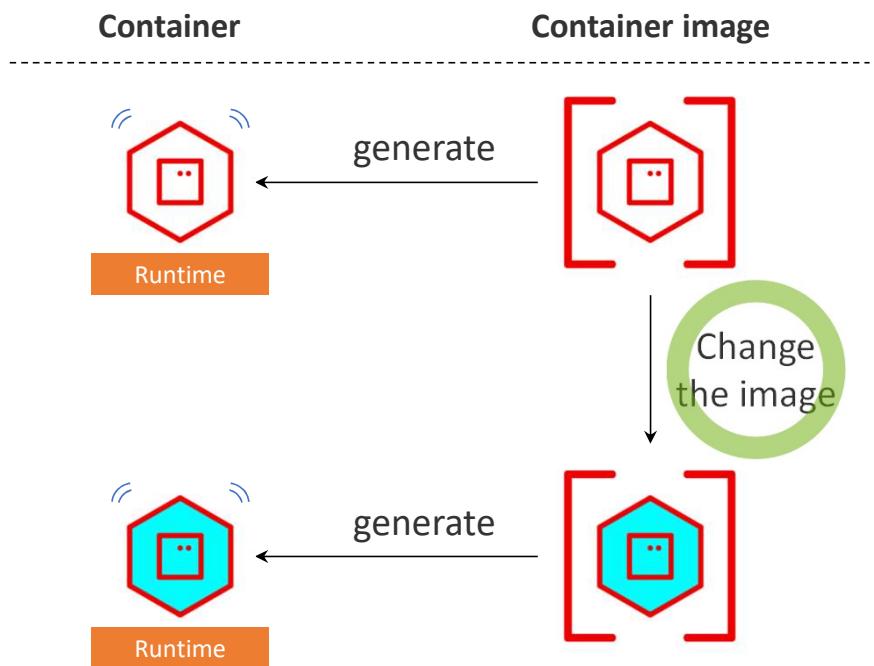
	Virtual machine	Container
Size	1 to 2 digits GB	2 digits MB to 1 digit GB
Start-up time	In minutes	In seconds



Container Characteristics (3 of 3)

Containers have a number of characteristics.

- ▶ **Generate from image**
 - Created by cloning from a container image.
- ▶ **Immutable**
 - A container started from the same container image will **always be the same**.
- ▶ **Ephemeral**
 - Changes made to the container are lost when the container stops.
 - The container itself has no persistence.
 - If you want to change the container, **change the container image and start a new container.**
 - Discard the old container.



Containers Improve Application Development and Operations

The **benefits of using containers** are **more for applications** than for infrastructure.

Application benefits

- **Improve development productivity**
 - Automated test execution to delivery
 - Preparing multiple environments on a single base
- **Operational efficiency**
 - Automatic response when trouble occurs
 - Reduction of manual work
- **Faster release cycles**
 - Frequent releases of fixes and updates
- **Portability**
 - Application regardless of execution base

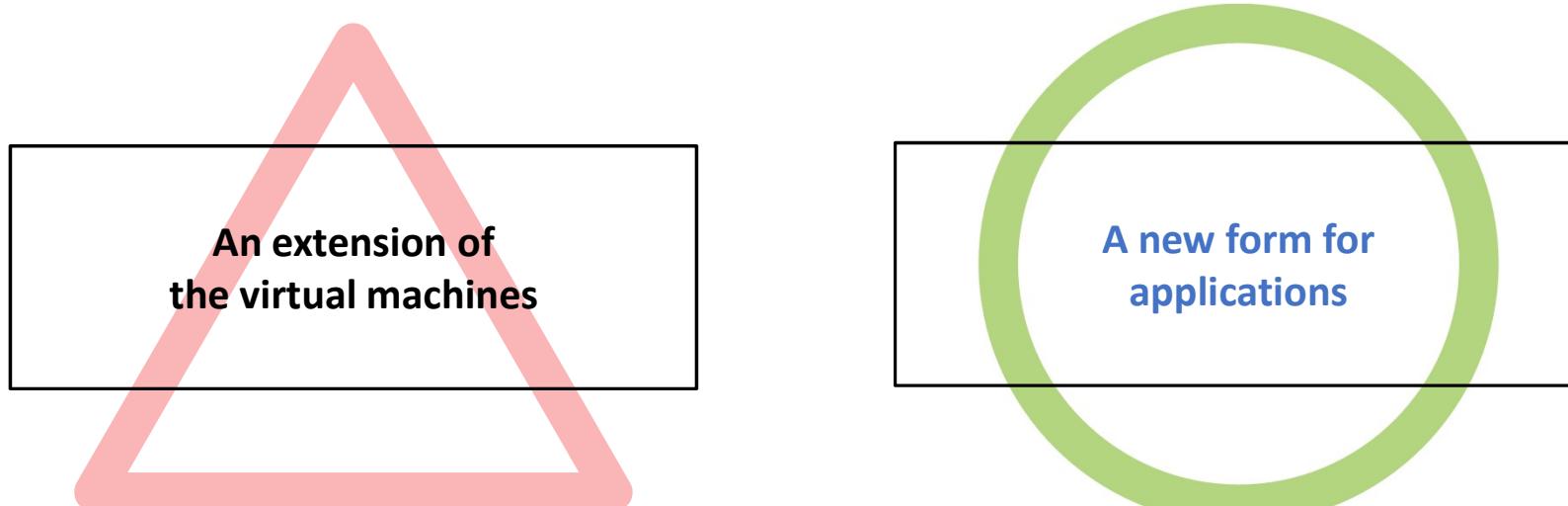
Infrastructure benefits

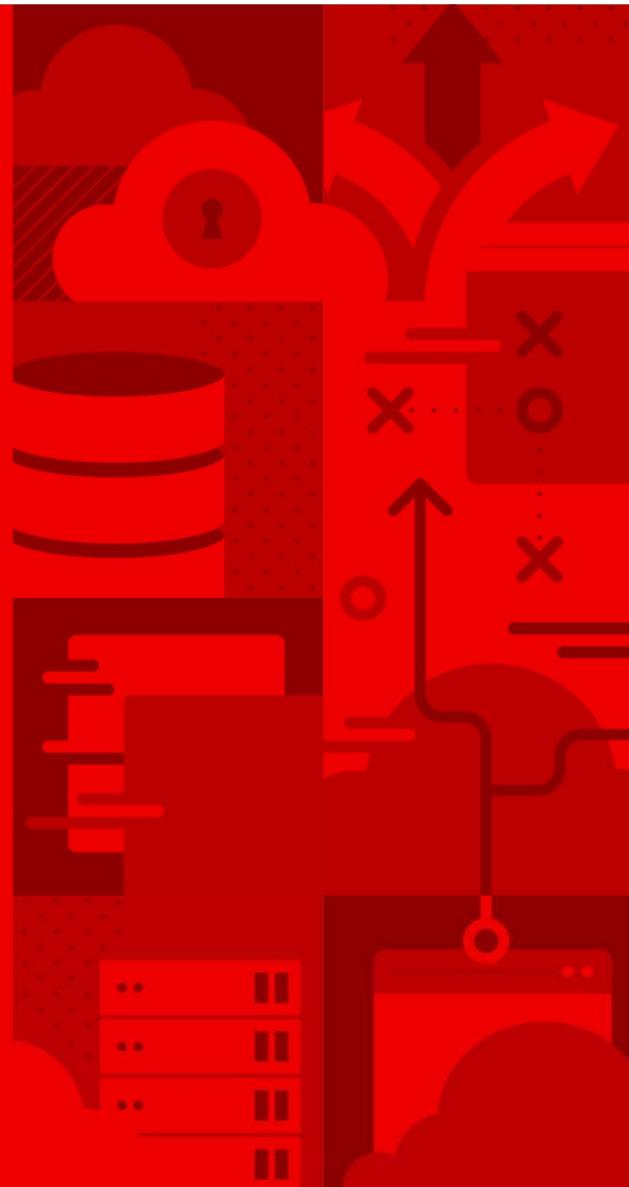
- **Improved consolidation ratio**
 - Improved infrastructure resource utilization, reduced costs
- **Operational efficiency**
 - Automation of infrastructure resource provisioning
 - Standardizing the work

Containers are the new form of applications

It's best to think of containers in terms of applications.

- When we think of containers as an extension of virtual machines, it's easy to focus on the infrastructure benefits and neglect the application benefits.
- Think of containers as the new form of your applications and realize greater benefits.





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

