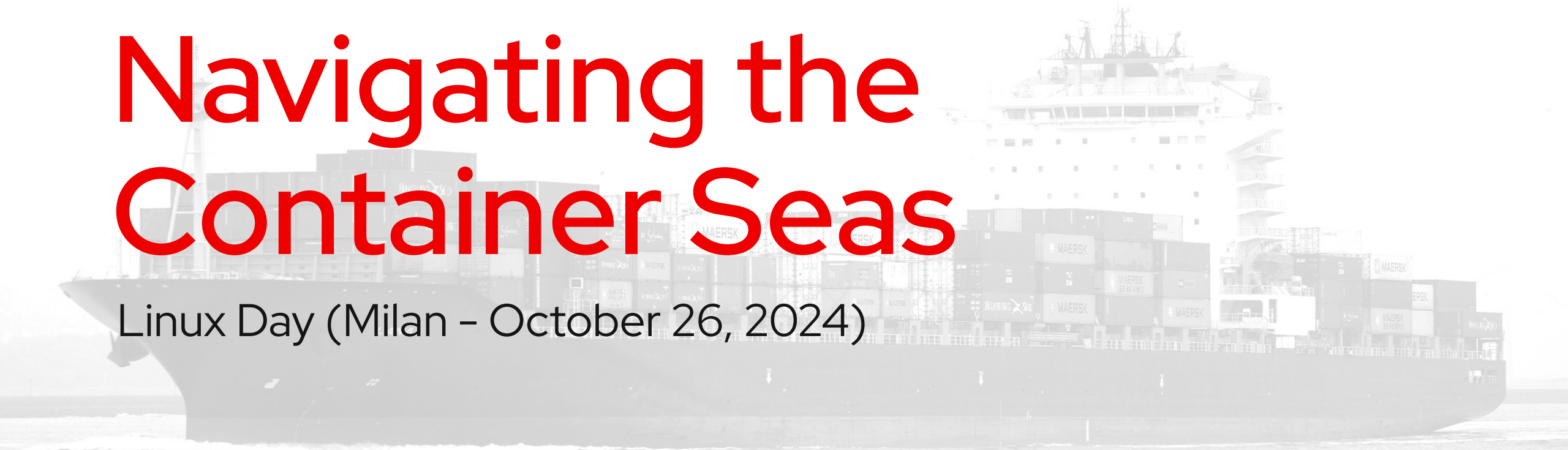
Kubernetes 101: Navigating the Container Seas



Linux Day (Milan - October 26, 2024)

Alfonso Cancellara

Technical Account Manager, OpenShift @ Red Hat

Red Hat

Kubernetes is one of the largest Open Source projects to date

over 88,000 contributors across 44 countries

Source: kubernetes.io/blog/2024/06/06/10-years-of-kubernetes/

Red Hat

Why Red Hat ?

- 2nd largest contributor to Kubernetes
- Part of IBM (6th largest contributor)

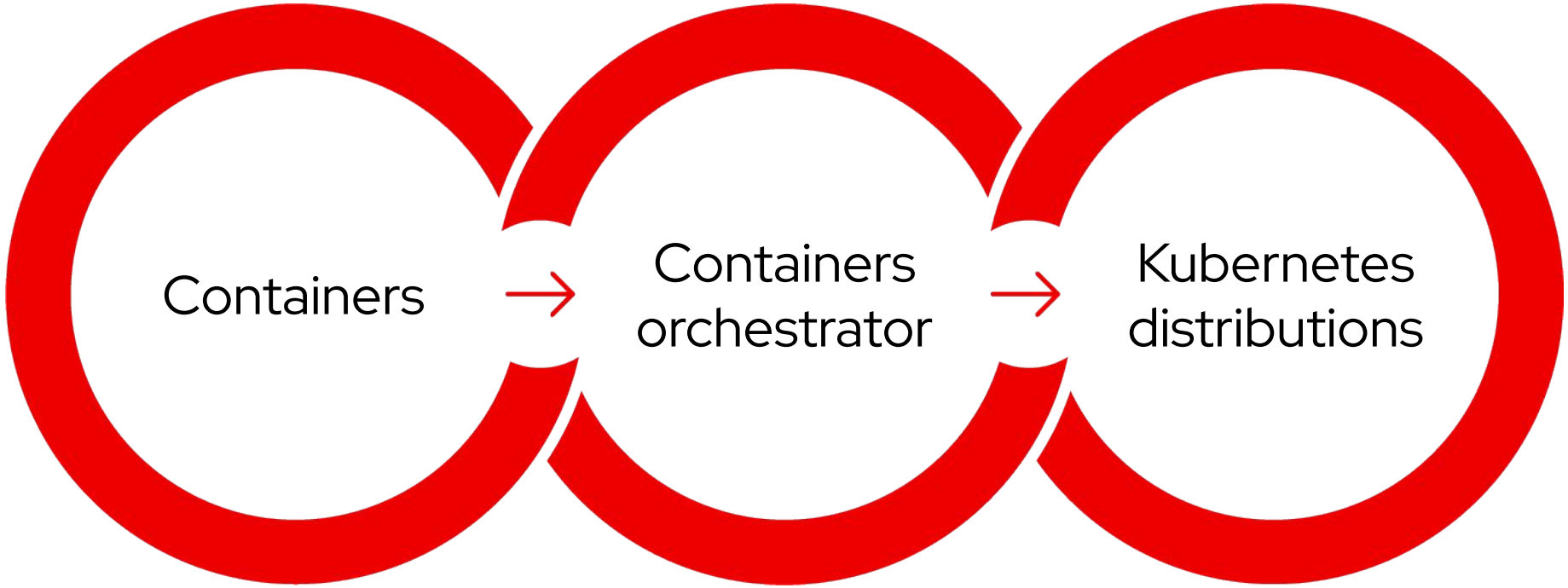
Why me ?

- Working on Kubernetes since 2019
- Currently working on OpenShift in Red Hat

Source: k8s.devstats.cncf.io/d/9/companies-table

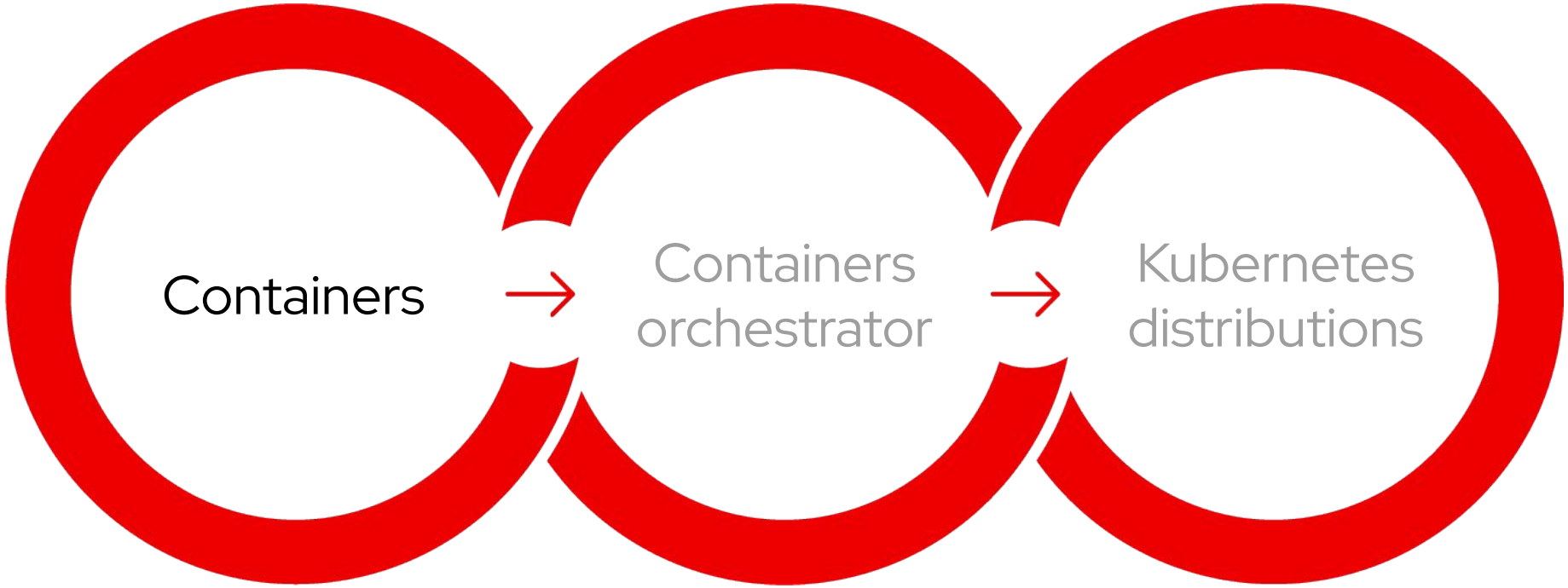
Red Hat

What we'll discuss



Red Hat

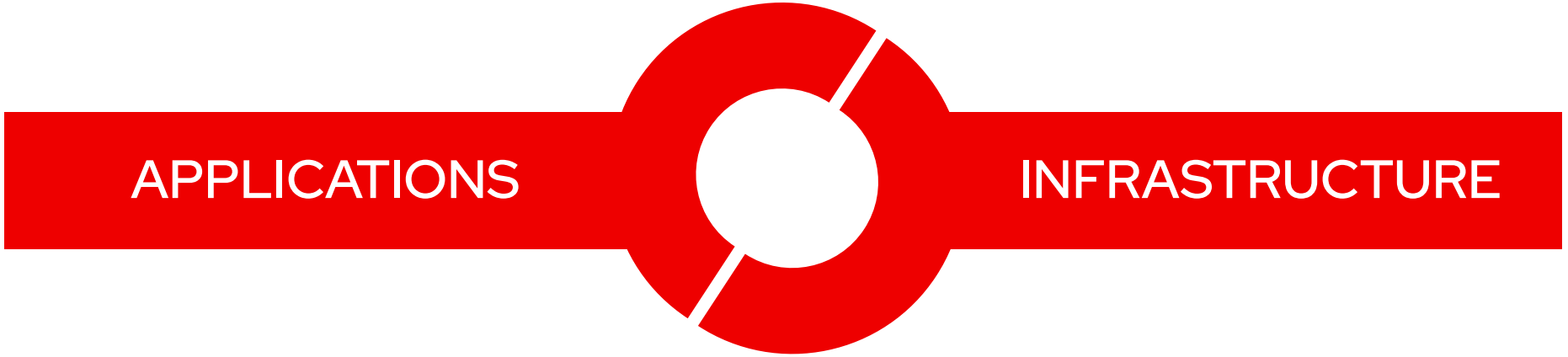
What we'll discuss



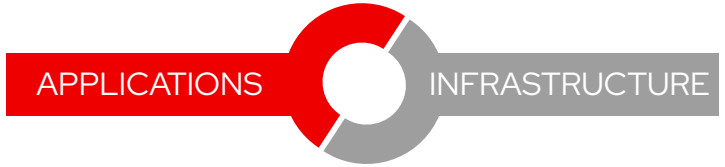
Red Hat

What are containers ?

It depends who you ask



Red Hat



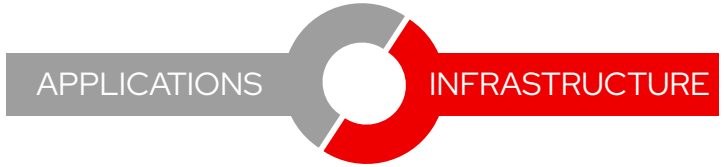
The problem

Applications have different requirements: languages, libraries, and tools

The solution

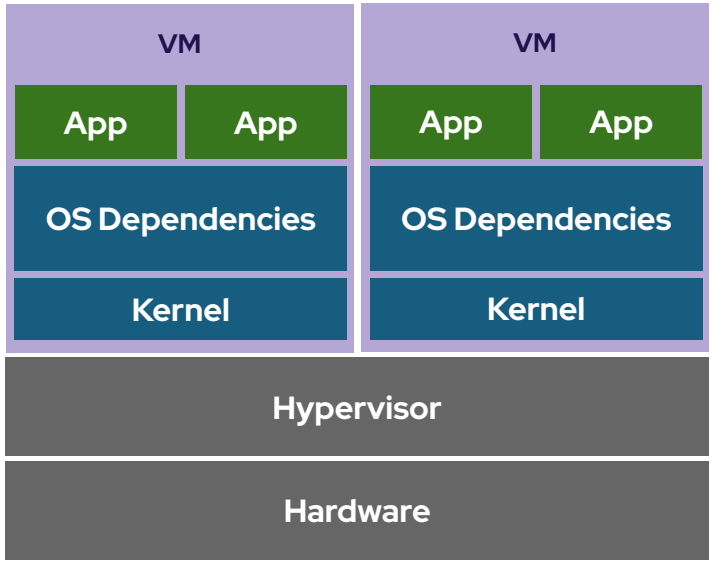
Package applications as units of software that hold together all the needed components

Red Hat



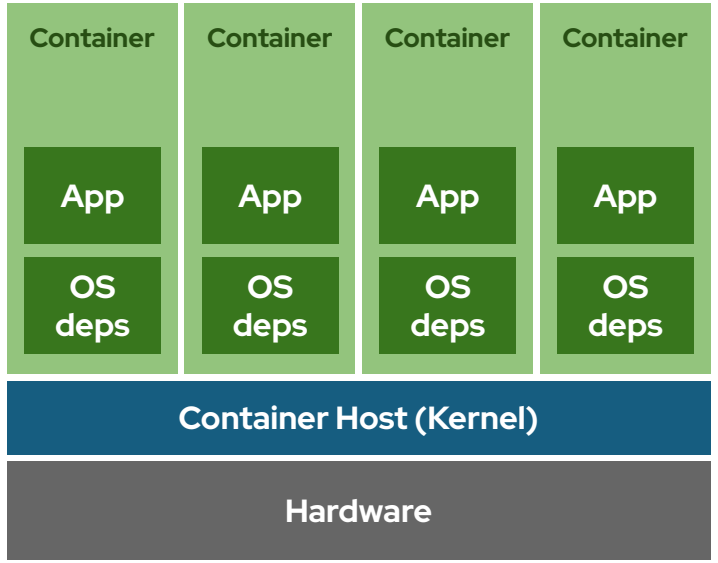
The problem

VMs are “heavy” and usually **not** portable across hypervisors



The solution

Isolated **processes** on a shared kernel (using Linux technologies)



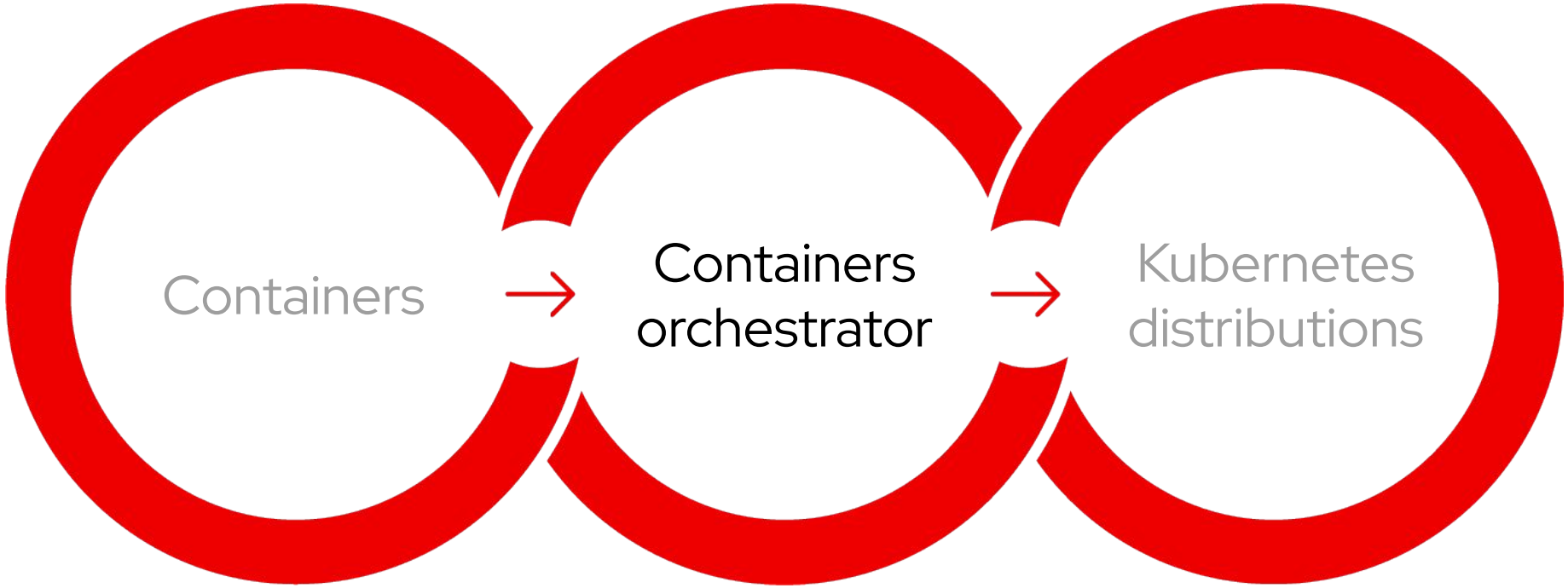
Red Hat

DEMO 1

Create and run a container locally

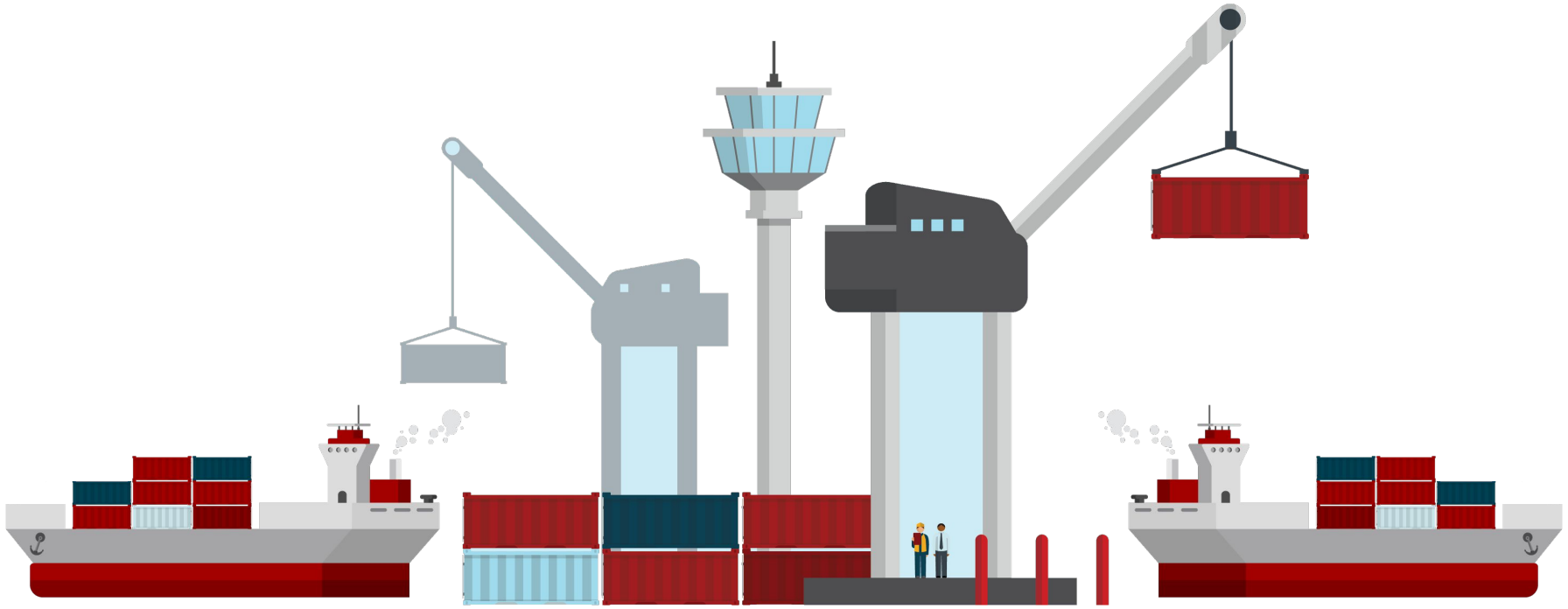
Red Hat

What we'll discuss



Red Hat

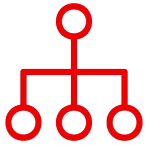
How to manage containers at scale ?



Red Hat

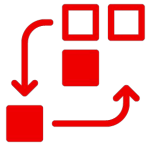
How to manage containers at scale ?

Service discovery



Load balancing

Storage orchestration



Automated rollouts and rollbacks

Self-healing

Secret and configuration management

Red Hat

How to manage containers at scale ?

Use a containers *orchestrator*

kubernetes

Red Hat

Kubernetes objects

Entities representing the state of the orchestrator


Pod : unit of computing (group of one or more containers)


Deployment : set of identical Pods (replicas of the same app)


Service : way to expose Pods over the network


PersistentVolume : unit of storage (“disk” that is usable by a Pod)


ConfigMap : way to set configurations in Pods


Secret : way to store confidential data (ex. connection strings)

Red Hat

Anatomy of a Kubernetes object

Represented as a YAML file

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: demo-pod
  labels:
    app: http-server
spec:
  containers:
    - name: demo-container
      image: quay.io/…/demo-container:latest
       ports:
         - containerPort: 8000
```

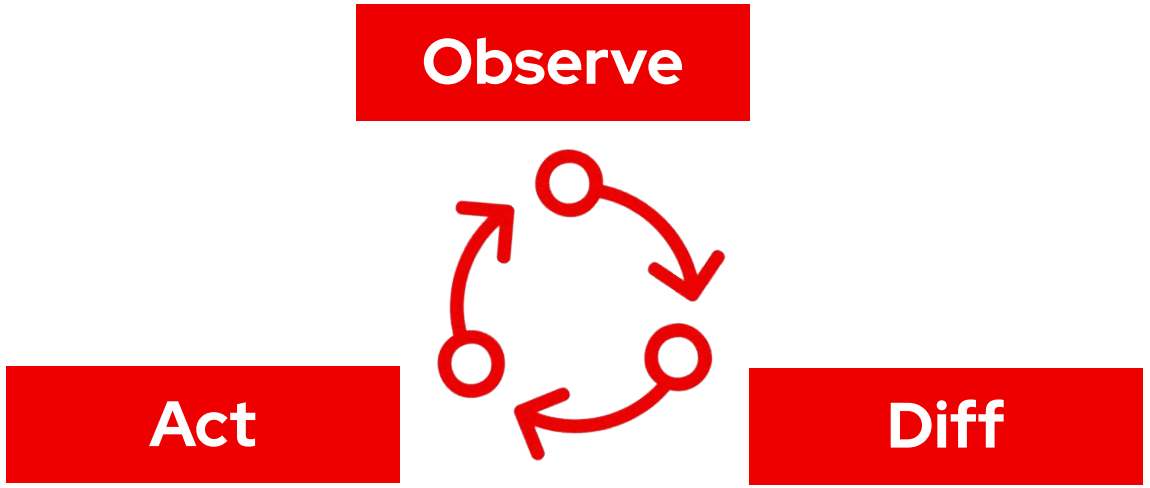

Red Hat

Reconciliation / Control loop

A core Kubernetes concept

Kubernetes is based on the concept of a **declarative specification of the desired state**

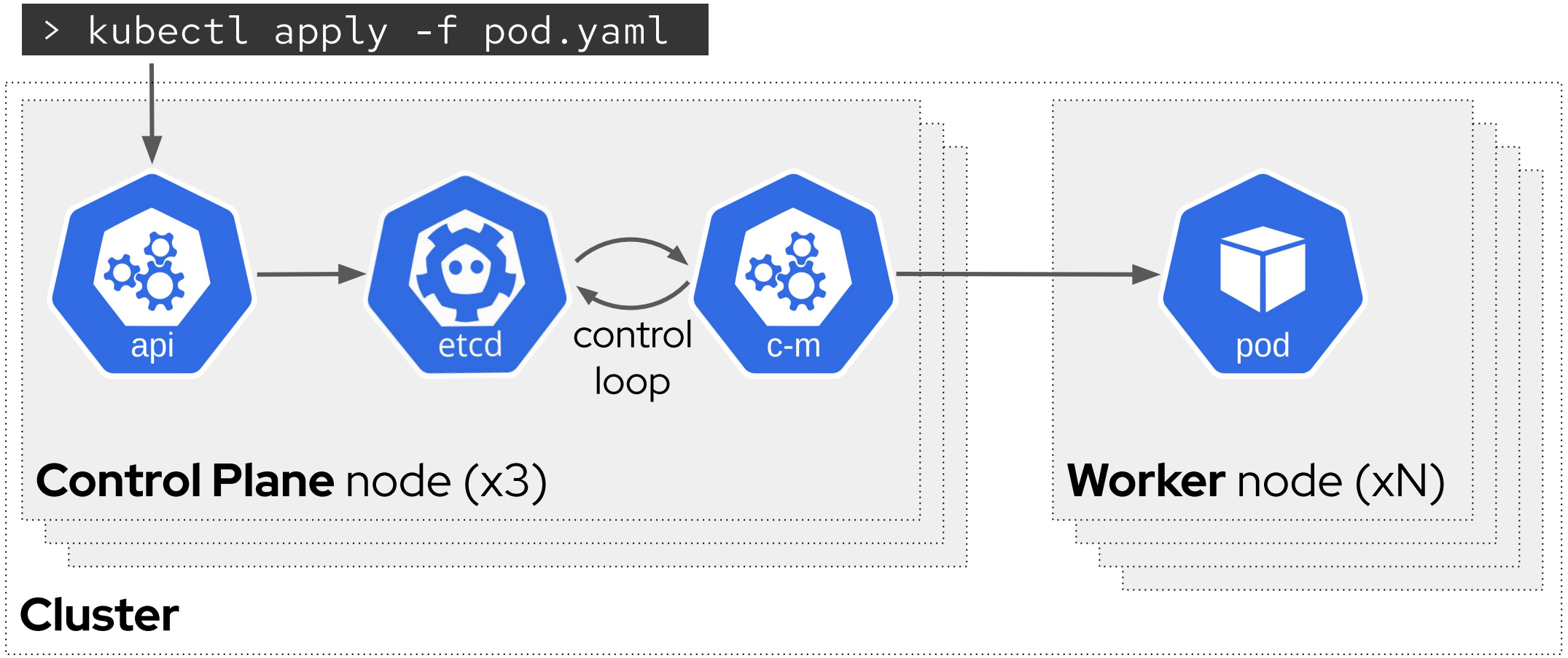
and the **use of reconciliation loops** to **drive the actual state toward the desired state**



Red Hat

Kubernetes architecture

Simplified view



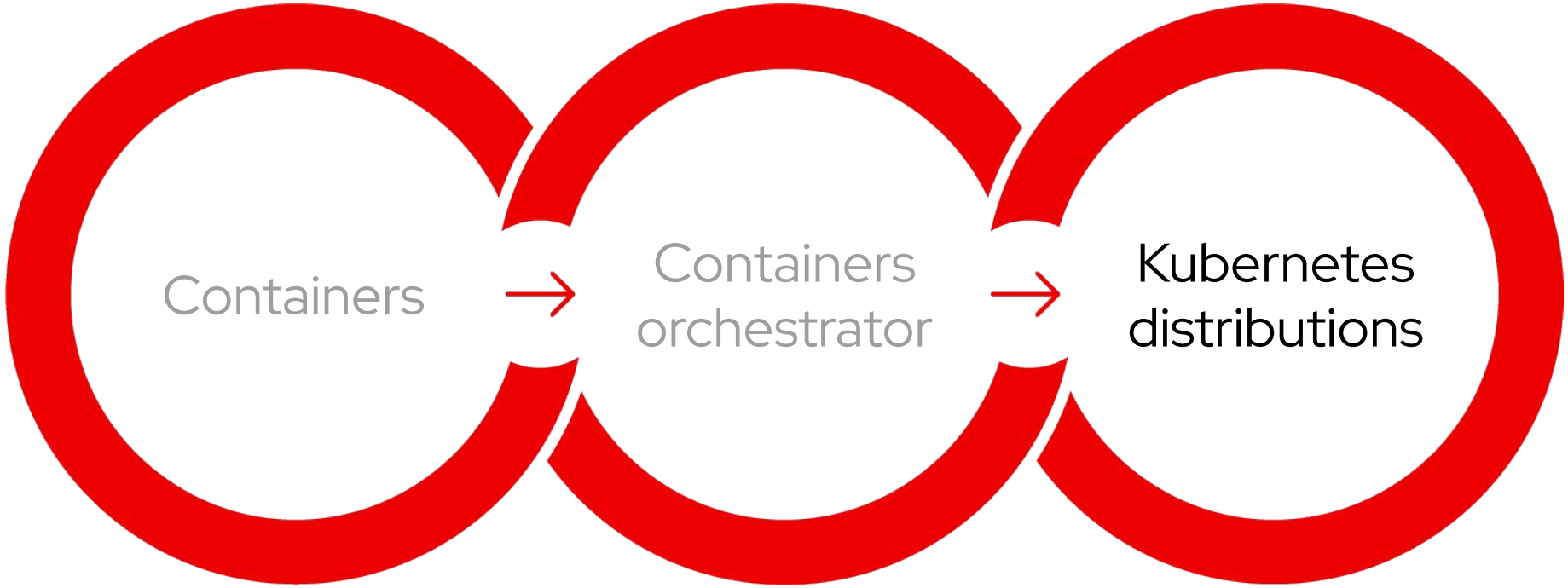
Red Hat

DEMO 2

Create and run a pod on K8s

Red Hat

What we'll discuss



Red Hat

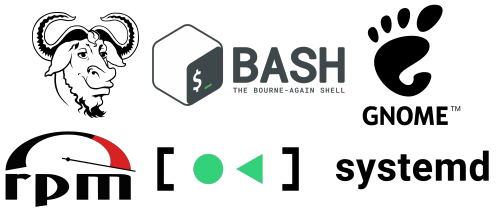
Kubernetes distributions

Are a packaged form of Kubernetes that includes additional tools, features, or services

Linux release

\+

Additional components



...

Production ready

RHEL release

Kubernetes release

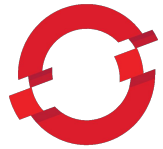
\+

Additional components

...

Production ready

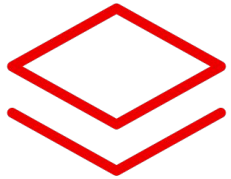
OpenShift release



Red Hat

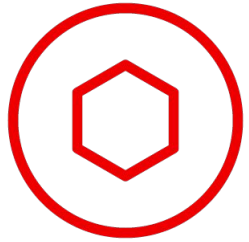
Red Hat Enterprise Linux CoreOS

The default operating system for all OpenShift cluster nodes



Based on **RHEL**

Controlled **immutability**



Container-centric

Red Hat

DEMO 3

Microservices app on OCP

Red Hat

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

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat

Red Hat