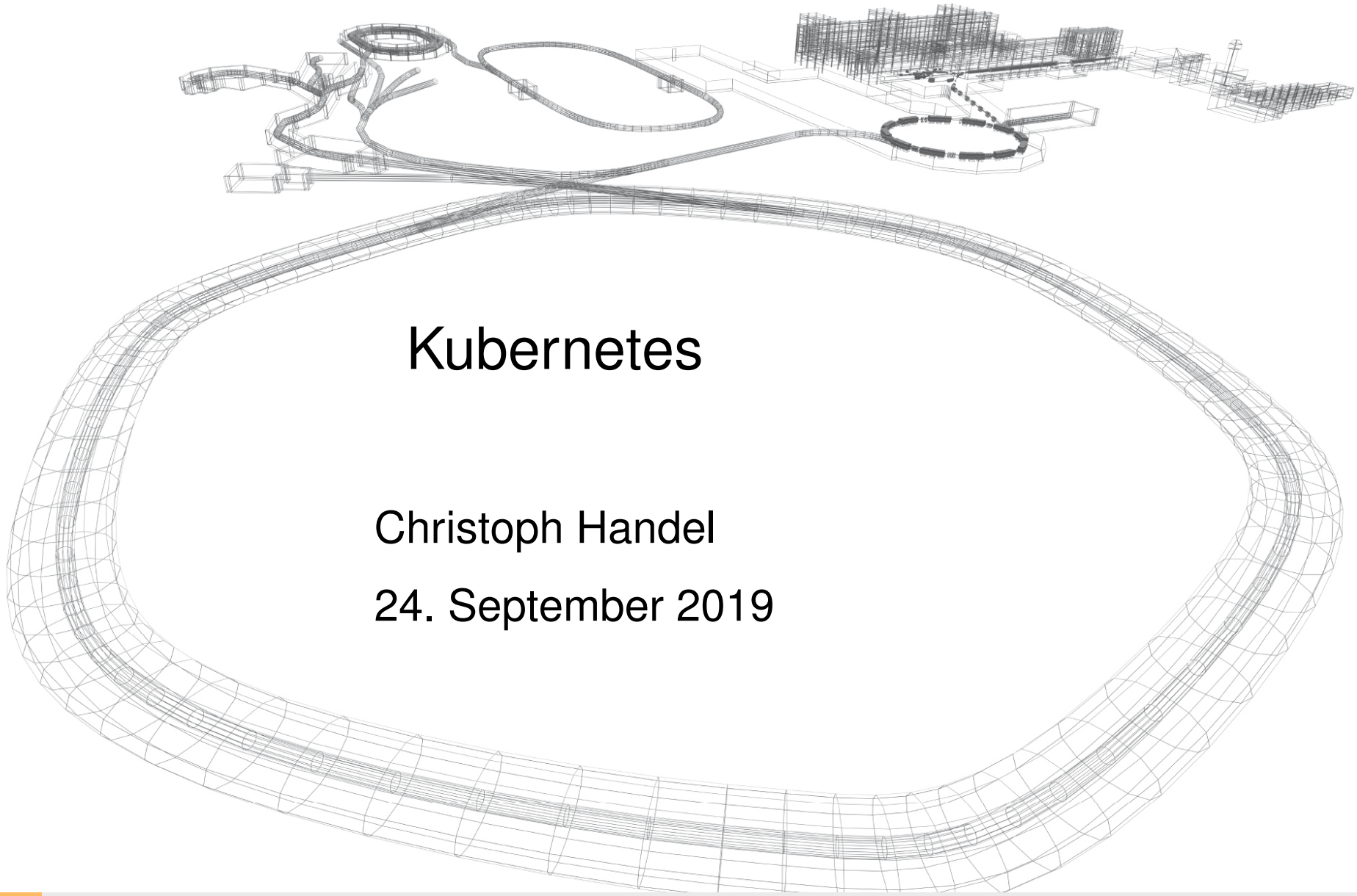


Kubernetes

Christoph Handel

24. September 2019

TOC

Who?

- History
  - created by Google in 2015
  - donated to Cloud Native Foundation
  - which is part of the Linux Foundation
  - apache license
- Used
  - Redhat (Openshift)
  - Oracle Cloud Infrastructure Container Engine
  - Amazon Elastic Kubernetes Service
  - Google Kubernetes Engine

Who?

- History
  - created by Google in 2015
  - donated to Cloud Native Foundation
  - which is part of the Linux Foundation
  - apache license
- Used
  - Redhat (Openshift)
  - Oracle Cloud Infrastructure Container Engine
  - Amazon Elastic Kubernetes Service
  - Google Kubernetes Engine

What?

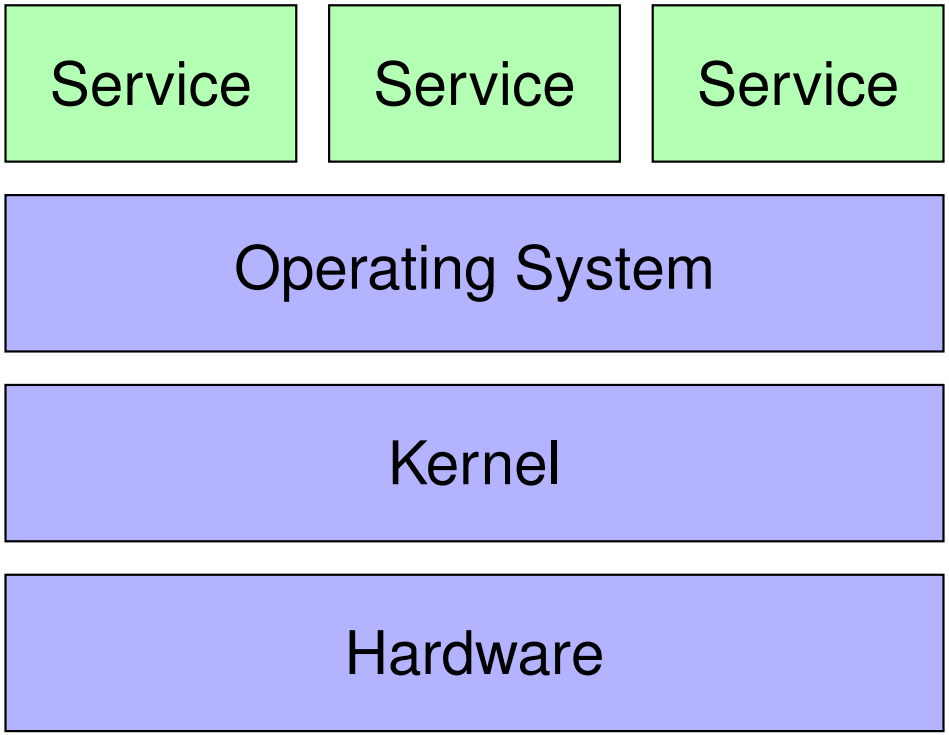
Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation.

What?

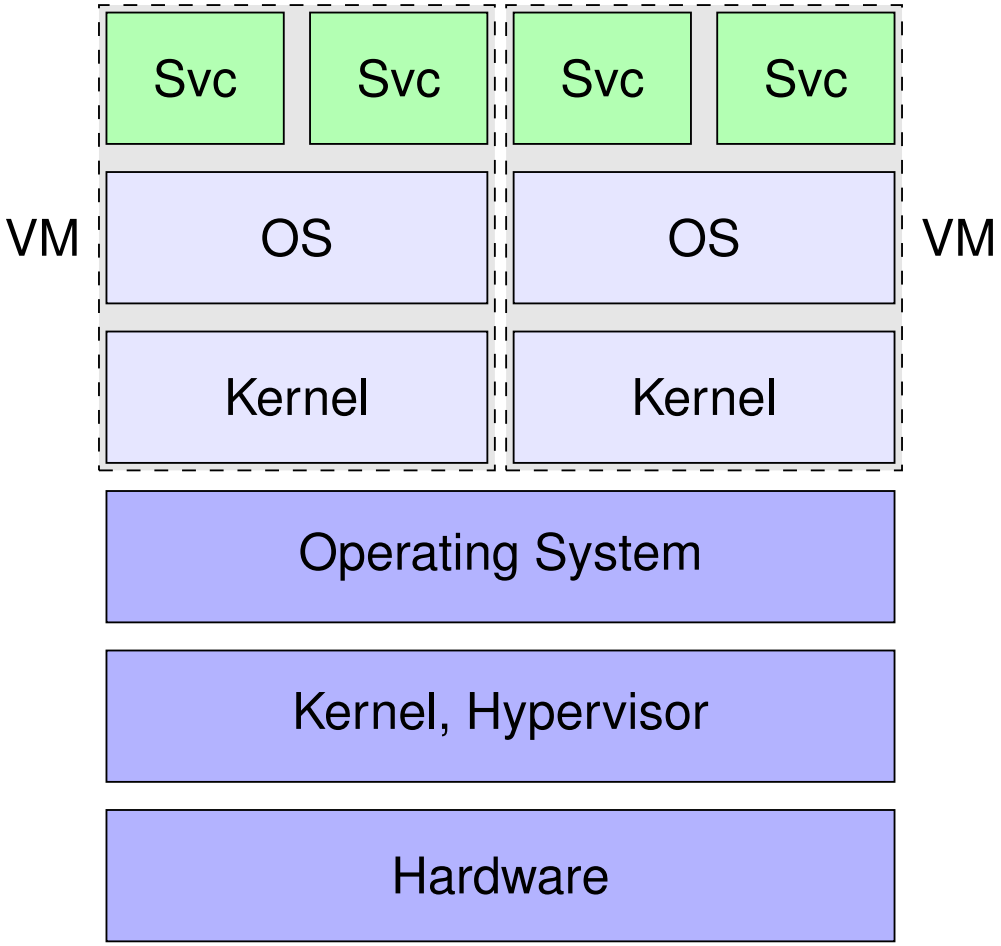
Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation.

TOC

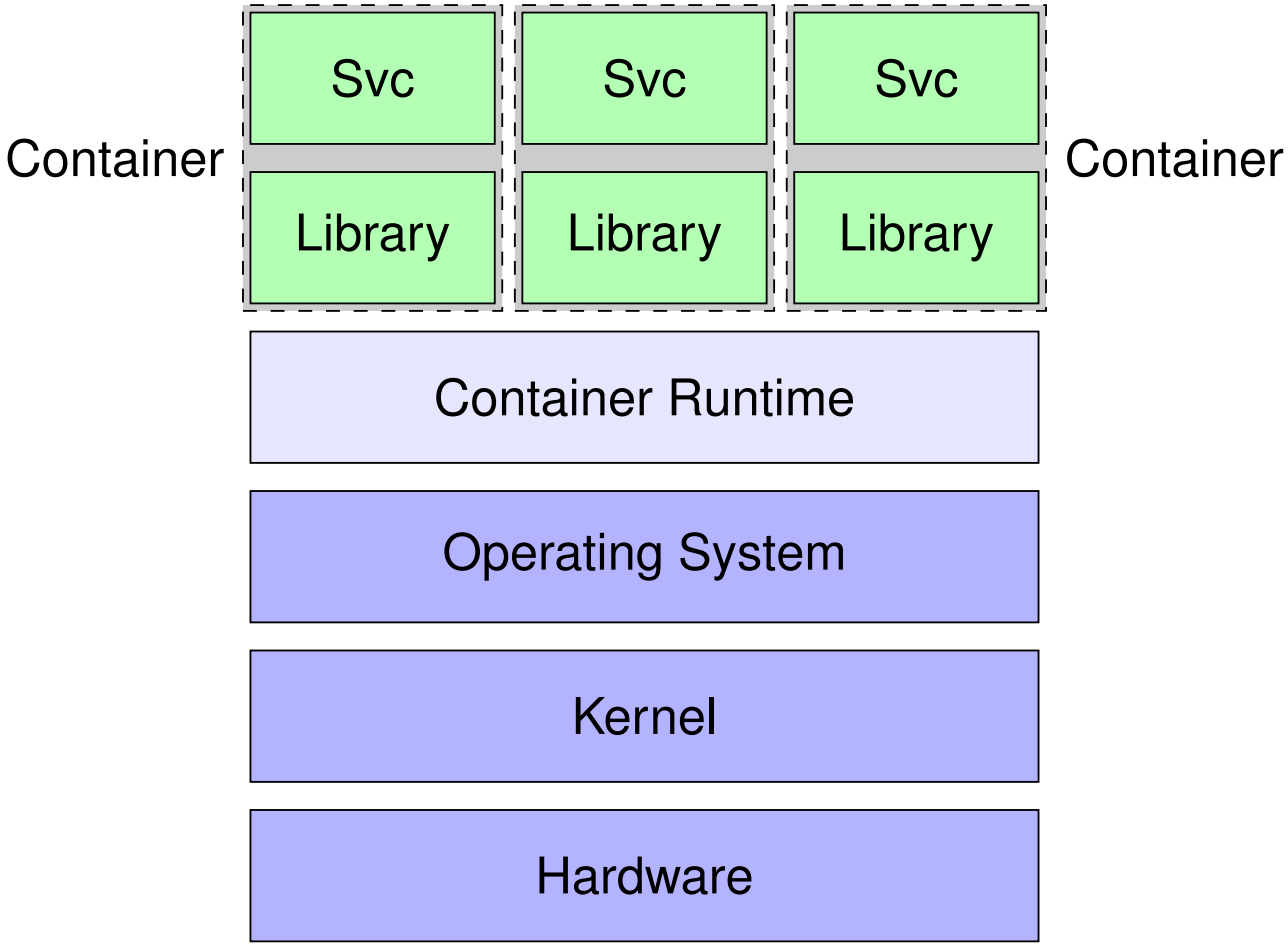
Bare Metal


Service
Service
Service
Operating System
Kernel
Hardware

Virtual Machine



Container

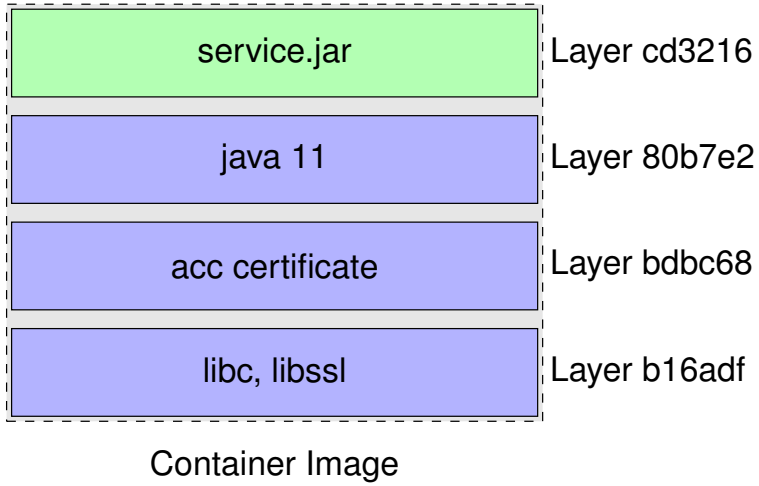

Container
Svc
Svc
Svc
Library
Library
Library
Container
Container Runtime
Operating System
Kernel
Hardware

Container

- Encapsulate a Service and all it's required libraries.
- Is isolated from other Processes. Including CPU, Memory, Network and Filesystem.
- Runs on the same Kernel.
- Becomes a portable Software Package

Container Images and Layers

- A container image constists of multiple layers
- Each layer can add or overwrite files in lower layers
- Layers are checksumed
- Layers can be shared



Building Base Image

```
FROM centos:7.6.1810

# set repositories
RUN rm -f /etc/yum.repos.d/*
ADD repos/* /etc/yum.repos.d/

# install some default packages
RUN yum install -y gsicert unzip which

# cleanup to keep image smaller
RUN yum clean all
```

- `podman build -t aco/base:7.6.1810`
- `podman push aco/base:7.6.1810 registry.acc.gsi.de/aco/base:7.6.1810`

Building Application Images

```
FROM aco/jdk11:7.6.1810

# download and unpack application to /opt
RUN curl -f -L -o /opt/app.zip \
  "https://artifacts.acc.gsi.de/.../app.zip"
RUN unzip /opt/app.zip -d /opt && rm -f /opt/app.zip

# set default command of container
CMD /opt/bin/app.sh
```

Not everything is docker

- Container Runtime
  - docker
  - rkt
  - cri-o
  - …
- Container Build
  - docker
  - podman
  - buildah
  - …

TOC

What?

Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation.

Declare

- Write declaration
- Hand declaration to kubernetes
- Kubernetes downloads images, configures network, creates containers
- Kubernetes maintains desired state, restarting crashed containers, etc.

Deployment

Define a Deployment that runs a container two times.

```
kind: Deployment
metadata:
  name: lsa-server
spec:
  replicas: 2
  # template for creating pods
  template:
    spec:
      containers:
      - name: lsa-server
        image: sample/lsa-server:13.1.0-RC-SR-SNAPSHOT
        # we update the image without incrementing the version
        imagePullPolicy: Always
```

Service

Make the service externally available with loadbalancing.

```
kind: Service
metadata:
  name: lsa-server
spec:
  type: LoadBalancer
  ports:
  - port: 52325
    protocol: TCP
    name: registry
  sessionAffinity: ClientIP
  selector:
    app: lsa-server
```

TOC

Summary

- package software including dependencies
- declare everything: parameter, version, ip port, ...
- deploy identical software multiple times by changing parameters.
- creating test setups is cheap
- high availability
- privilige seperation

the bad

- need to declare everything: parameter, version, ip port, . . .
- update base image library requires rebuild application images
- infrastructure. Kubelet, metallb, etcd, coredns, . . .
- disk space
- complexity. Especially network with lot's of natting
- danger of growing mold. Enforced cleanup policy required.

the unknown

Not yet tested/solved

- storage. Only nfs used, this won't scale well
- permission system
- monitoring, logging
- webinterface, dashboards