

Containers and their orchestration

SJK005 – Cloud computing

University Master's Degree in Intelligent Systems

University Jaume I

Competences and learning outcomes

1. To distinguish between virtualization and containerisation.
2. To know what container images and containers are, and its advantages.
3. To know what container orchestration is and its benefits.

Contents

1. Virtualization vs containerisation.
2. Containers with podman.
3. What is container orchestration.

Virtualization vs containerisation

Virtualization

In the realm of computer science, virtualization allows to create virtual version of hardware, OS, storage and other resources.

This is done by means of specialised software: hypervisors.

There are two kinds of hypervisors:

1. Those running directly on the hardware, Ex: Xen, Microsoft Hyper-V.
2. Those running on top of an OS, Ex: VirtualBox or VMware.

Virtualization vs containerisation

Virtualization

We can install guest VMs on top of hypervisors.

Virtualization can be only achieved in "modern" hardware. If you have an old computer, probably it does not support hypervisors of Type-1.

[RedHat video](#)

Virtualization vs containerisation

Containers ([RedHat video](#))

Containers are technologies that allow the packaging and isolation of applications with their entire runtime environment—all the files necessary to run. This makes it easy to move the contained application between environments while retaining full functionality.

You can find lot of information about open containers development at <https://github.com/containers>

Source: RedHat

Virtualization vs containerisation

Container images - Images

Container images are the core of containers technologies.

Container images contains any libraries (dependencies) that they need to run. These libraries can be different (content and even version), from any other image.

This way "dependecy" issues are solve (works in my machine problem).

Container images are a way to define isolate future running groups of processes isolated from any other, all of them running in the same linux machine.

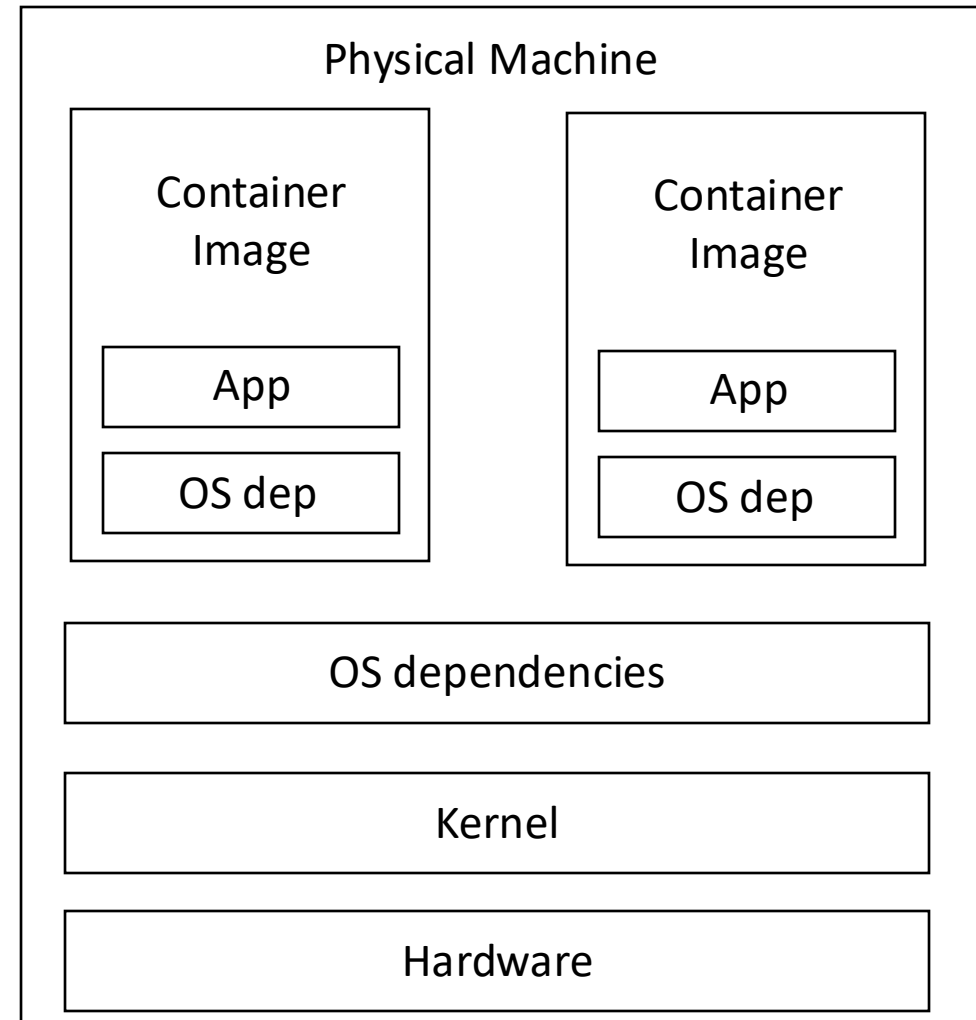
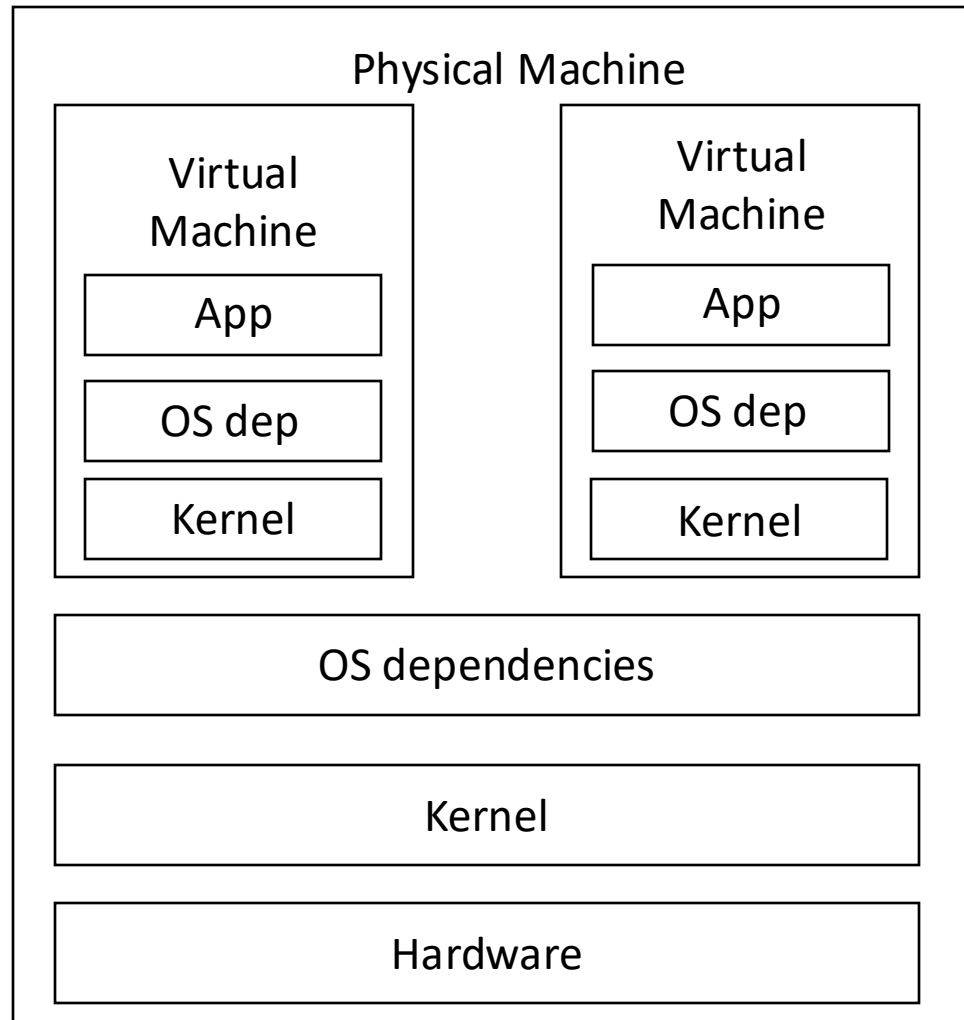
Virtualization vs containerisation

Containers

Containers are groups of processes that run on a Linux system. They are isolated to each other, so no container interfere with any other.

They are instances of container images.

Virtualization vs containerisation



Virtualization vs containerisation

Virtualization is related with infrastructure (IaaS), while **Containerisation** is related with software (SaaS).

If you need to deploy a machine with some specific characteristics, you will use some virtualization software solution (ex: VirtualBox, VMWare).

If you need to deploy some running (isolated) software, you will use some containerisation software solutions (ex: docker, podman).

Containers with podman

What is podman?

Podman is an open-source container, pod, and container image management engine. Podman makes it easy to find, run, build, and share containers.

Podman commands have the same syntax than docker commands, to make smooth transition between both platform.

Source: podman.io

Containers with podman

Why to use podman?

Podman is open-source and free to use.

Podman does not run any daemon (docker does).

Superuser permissions are not required (docker does).

Podman has some advanced characteristics (ex: pods).

Podman share the same command syntax than docker.

Podman is oriented to work with Kubernetes.

Podman understands docker images.

Containers with podman

Install [podman](#)

```
# Ubuntu 20.10 and newer  
sudo apt-get update  
sudo apt-get -y install podman  
podman --version
```

Containers with podman

Podman desktop

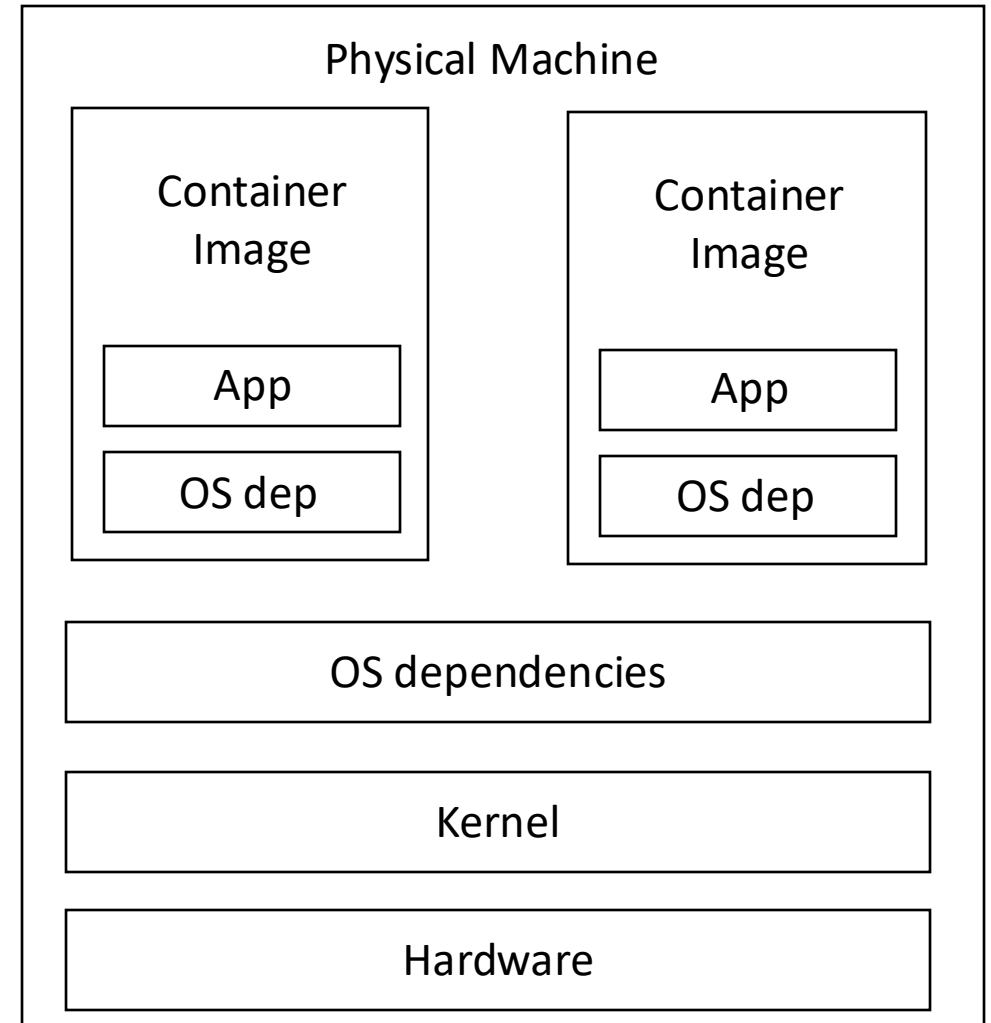
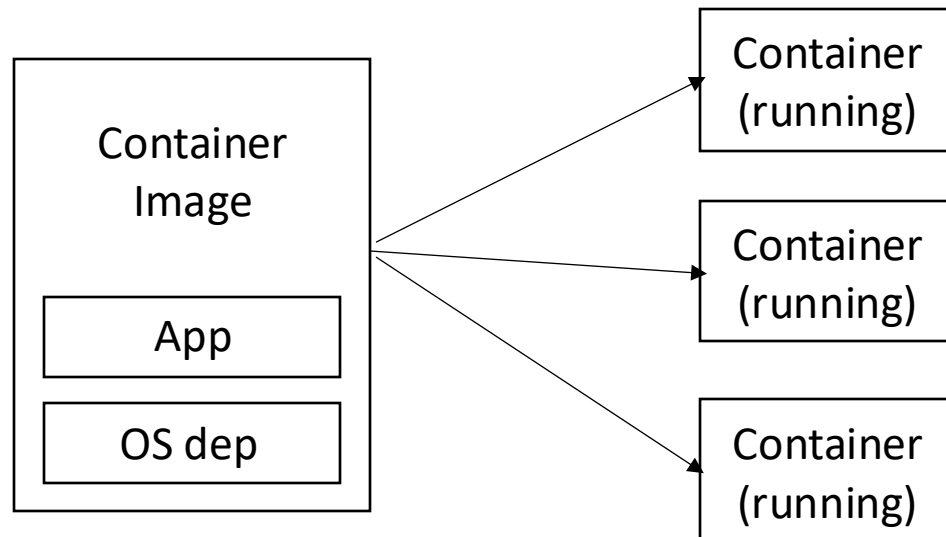
There exist a desktop application with GUI that can be used instead the CLI.



Containers with podman

Containers are instances of container images.

You can have several running containers, instances of the same container image.



Containers with podman

Where to find already built container images

There exists a container hub maintained by docker:

[Docker-hub](#).

You can search there for containerised apps, like servers, databases, message queue brokers, and many more.

[ReadHat](#) maintains an alternative hub but that by docker is most popular and compatible with podman.

Containers with podman

Use existing images

Let's search for "httpd" in the docker-hub web page. This is the Apache web server.

Select "Official image".

```
podman pull httpd
```

Check existing images in your system:

```
podman images
```

Containers with podman

Create a running container from an existing image

For creating a running container:

```
podman run -dit --rm --name servidor -p 8080:80 -v  
"$PWD":/usr/local/apache2/htdocs httpd
```

Open a web browser and navigate to <http://localhost:8080>, you will see the content your \$PWD folder.

Containers with podman

Anatomy of the run command:

```
podman run -dit --rm --name servidor -p 8080:80 -v  
"$PWD":/usr/local/apache2/htdocs httpd
```

-dit --> Deattached mode + interactive.

--rm --> Remove container when exiting.

--name servidor --> Name for the container.

-p 8080:80 --> Maps container port 80 to local port 8080.

-v "\$PWD":/usr/local/apache2/htdocs --> Maps container volume /usr/local/apache2/htdocs to local volume \$PWD (current directory).

Containers with podman

Basic podman commands to work with containers

`podman container ls -->` list all running containers.

`podman ps -->` same as before command.

`podman container ls -- all -->` list all containers available.

`podman container stop name -->` stops a running container.

`podman container start name -->` starts a stopped container.

`podman container rm name -->` removes a stopped container.

`podman ps --size -->` Shows size of the container.

Containers with podman

For command line lovers:

Searching images in hubs:

```
podman search ubuntu
```

Searching for official image:

```
podman search --filter=is-official ubuntu
```

Containers with podman

Exercise:

Search for node-red image.

Pull the image.

Run a node-red image.

Try to use different running options.

Guided exercises:

Let's see how to create new images in several ways (laboratory).

Containers with podman

Other useful podman command:

To list all images: `podman images`

To list all running containers: `podman container ls`

To list all containers: `podman container ls -all`

To stop a running container: `podman stop container's_name`

To attach to a stopped container: `podman attach container's_name`

To remove a container: `podman container rm container's_name`

To remove an image: `podman image rm image's_name`

More commands in the [aulavirtua](#) link.

What is container orchestration

Container orchestration automates the deployment, management, scaling, and networking of containers. Enterprises that need to deploy and manage hundreds or thousands of Linux[®] containers and hosts can benefit from container orchestration.

Container orchestration can be used in any environment where you use containers. It can help you to deploy the same application across different environments without needing to redesign it. And microservices in containers make it easier to orchestrate services, including storage, networking, and security.

Source: RedHat.

What is container orchestration

We can do some basic orchestration with podman using pods.

Pods are the basic unit in container orchestration.

There are specific software tools for orchestrating containers. One of these tools is Kubernetes.

Let's practice it in the lab!!!