



# LATINO**WARE** 2022

19º Congresso Latino-americano de  
Software Livre e Tecnologias Abertas

**02 a 04 de novembro de 2022**  
Hotel Golden Park Internacional Foz do Iguaçu  
Foz do Iguaçu | Paraná | Brasil

Realização:



# Podman Para Iniciantes

*Autor:*

Daniel Lara

# O que é Podman?



podman

É uma ferramenta nativa do Linux sem daemon, de código aberto, projetada para facilitar a localização, execução, criação, compartilhamento e implantação de aplicativos usando contêineres e imagens de contêiner da Open Containers Initiative ( OCI )

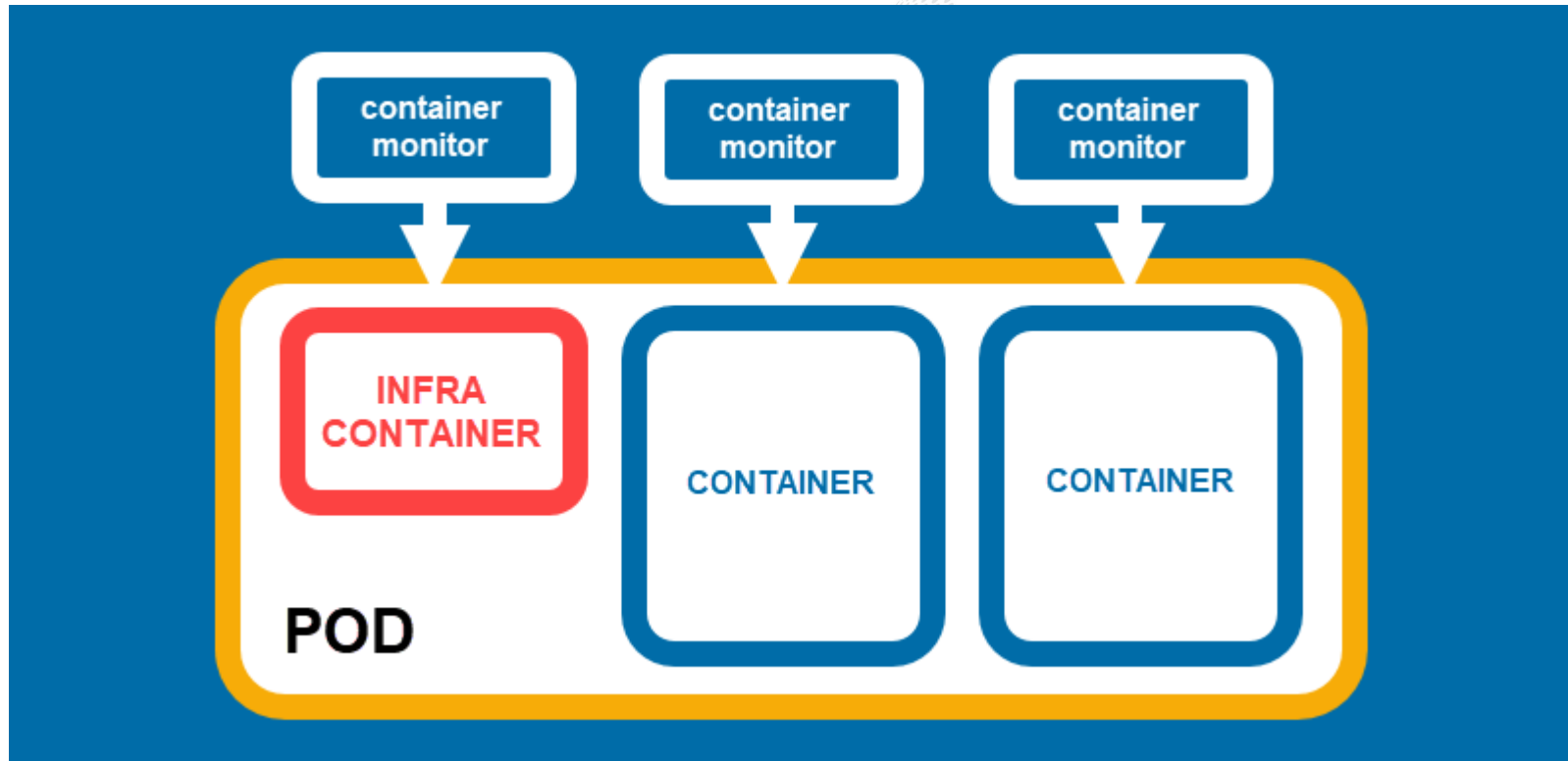
# Vantagens?

- Sem daemons
- Contêineres podem ser executados como root ou não
- Possui Isolamentos entre Pods e Containers

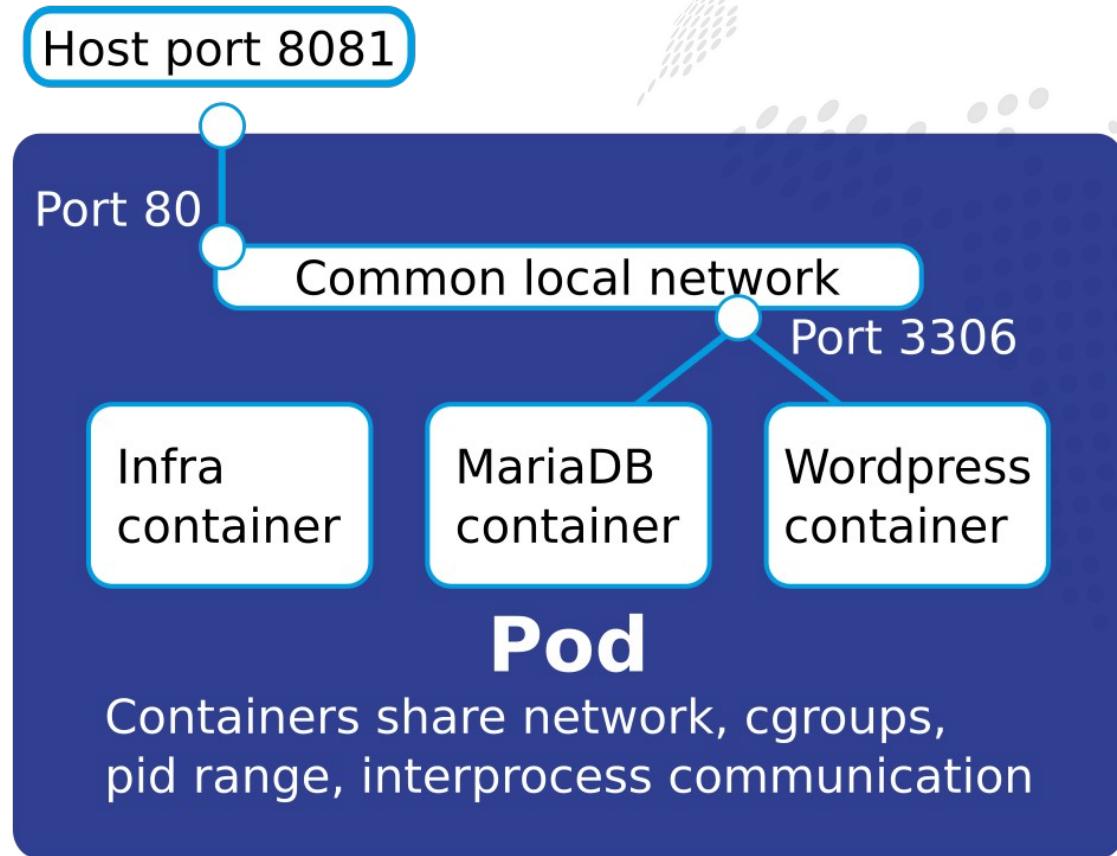
# O que é Pods?

São Semelhantes á definição de Pods do Kubernetes;  
Podemos capturar a descrição YAML de pods e containers  
Locais onde ajude na transição para um ambiente  
orquestrado como o Kuberntes.

# O que é Pods?



# O que é Pods?





# Conhecendo o Podman

```
[daniel@kvm ~]$ podman --help
Manage pods, containers and images

Usage:
  podman [options] [command]

Available Commands:
  attach      Attach to a running container
  auto-update Auto update containers according to their auto-update policy
  build       Build an image using instructions from Containerfiles
  commit      Create new image based on the changed container
  container   Manage containers
  cp          Copy files/folders between a container and the local filesystem
  create      Create but do not start a container
  diff        Display the changes to the object's file system
  events      Show podman events
  exec        Run a process in a running container
  export      Export container's filesystem contents as a tar archive
  generate     Generate structured data based on containers, pods or volumes
  healthcheck Manage health checks on containers
  help        Help about any command
  history     Show history of a specified image
  image       Manage images
  images      List images in local storage
  import      Import a tarball to create a filesystem image
  info        Display podman system information
  init        Initialize one or more containers
  inspect     Display the configuration of object denoted by ID
  kill        Kill one or more running containers with a specific signal
  load        Load image(s) from a tar archive
  login       Login to a container registry
  logout      Logout of a container registry
  logs        Fetch the logs of one or more containers
  machine     Manage a virtual machine
  manifest    Manipulate manifest lists and image indexes
  mount       Mount a working container's root filesystem
  network     Manage networks
```

# Registries

```
[daniel@kvm ~]$  
[daniel@kvm ~]$ podman pull httpd  
? Please select an image:  
  registry.access.redhat.com/httpd:latest  
  ▶ registry.redhat.io/httpd:latest  
  docker.io/library/httpd:latest
```



# Pull

```
[daniel@kvm ~]$  
[daniel@kvm ~]$ podman pull httpd  
✓ docker.io/library/httpd:latest  
Trying to pull docker.io/library/httpd:latest...  
Getting image source signatures  
Copying blob e559e5380898 done  
Copying blob e9995326b091 done  
Copying blob bc66e7ea7efe done  
Copying blob 5d0f831d3c0b done  
Copying blob ee55ccd48c8f done  
Copying config fe8735c23e done  
Writing manifest to image destination  
Storing signatures  
fe8735c23ec5da867dea9f7e69a1d120c12329f32b8a45710ff1a873a1e456ad  
[daniel@kvm ~]$
```

# Run

```
[daniel@kvm ~]$  
[daniel@kvm ~]$ podman run -td -p 8181:80 httpd  
6b5f32ae269ce684faee2d2d68e7fc9816bc16b96f2f17aea332ed55e2247497  
[daniel@kvm ~]$
```

```
[daniel@kvm ~]$ podman ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
6b5f32ae269c	docker.io/library/httpd:latest	httpd-foreground	12 seconds ago	Up 12 seconds ago	0.0.0.0:8181->80/tcp	gallant_mahavira

```
[daniel@kvm ~]$
```

# EXEC

```
[daniel@kvm ~]$  
[daniel@kvm ~]$ podman exec -it 6b5f32ae269c /bin/bash  
root@6b5f32ae269c:/usr/local/apache2#  
root@6b5f32ae269c:/usr/local/apache2# █
```

# Pod

```
[daniel@kvm ~]$ podman pod --help
```

Manage pods

Description:

Pods are a group of one or more containers sharing the same network, pid and ipc namespaces.

Usage:

```
podman pod [command]
```

Available Commands:

clone	Clone an existing pod
create	Create a new empty pod
exists	Check if a pod exists in local storage
inspect	Displays a pod configuration
kill	Send the specified signal or SIGKILL to containers in pod
logs	Fetch logs for pod with one or more containers
pause	Pause one or more pods
prune	Remove all stopped pods and their containers
ps	List pods
restart	Restart one or more pods
rm	Remove one or more pods
start	Start one or more pods
stats	Display a live stream of resource usage statistics for the containers in one or more pods
stop	Stop one or more pods
top	Display the running processes of containers in a pod
unpause	Unpause one or more pods

```
[daniel@kvm ~]$
```

# Pod

```
[daniel@kvm ~]$ podman pod create --name web -p 8181:80 -p 3306:3306  
1d7034eb5e1720c7bcaaaa8ac2bb3902bfec24e5981bf3f48c3156f21dad35b4  
[daniel@kvm ~]$
```

```
[daniel@kvm ~]$ podman pod ps
```

POD ID	NAME	STATUS	CREATED	INFRA ID	# OF CONTAINERS
1d7034eb5e17	web	Created	17 seconds ago	1cbb80af0601	1

```
[daniel@kvm ~]$
```

# Pod

```
[daniel@kvm ~]$ podman run \  
> -d --restart=always --pod=web \  
> -e MYSQL_ROOT_PASSWORD="podman" \  
> -e MYSQL_DATABASE="wp" \  
> -e MYSQL_USER="wordpress" \  
> -e MYSQL_PASSWORD="w0rdpr3ss" \  
> --name=dbwp mariadb  
c1863d37644e03fa3e088ac3e09f1cccd2e47e8cae392482ee978288ea344cf  
[daniel@kvm ~]$  
[daniel@kvm ~]$
```

```
[daniel@kvm ~]$ podman run \  
> -d --restart=always --pod=web \  
> -e WORDPRESS_DB_NAME="wp" \  
> -e WORDPRESS_DB_USER="wordpress" \  
> -e WORDPRESS_DB_PASSWORD="w0rdpr3ss" \  
> -e WORDPRESS_DB_HOST="127.0.0.1" \  
> --name webwp wordpress  
cbe16e50f5847b94e2f82e6546ca86effe48cd68731a7b5f74bcb85e16aeb8e  
[daniel@kvm ~]$  
[daniel@kvm ~]$
```



# Pod

```
[daniel@kvm ~]$  
[daniel@kvm ~]$ podman ps -a --pod
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
1cbb80af0601	localhost/podman-pause:4.2.0-1663766104		10 minutes ago	Up 3 minutes ago	0.0.0.0:3306->3306/tcp, 0.0.0.0:8181->80/tcp	1d7034eb5e17-infra
1d7034eb5e17	web					
c1863d37644e	docker.io/library/mariadb:latest	mariadb	3 minutes ago	Up 3 minutes ago	0.0.0.0:3306->3306/tcp, 0.0.0.0:8181->80/tcp	dbwp
1d7034eb5e17	web					
cbe16e50f584	docker.io/library/wordpress:latest	apache2-foregroun...	About a minute ago	Up About a minute ago	0.0.0.0:3306->3306/tcp, 0.0.0.0:8181->80/tcp	webwp
1d7034eb5e17	web					

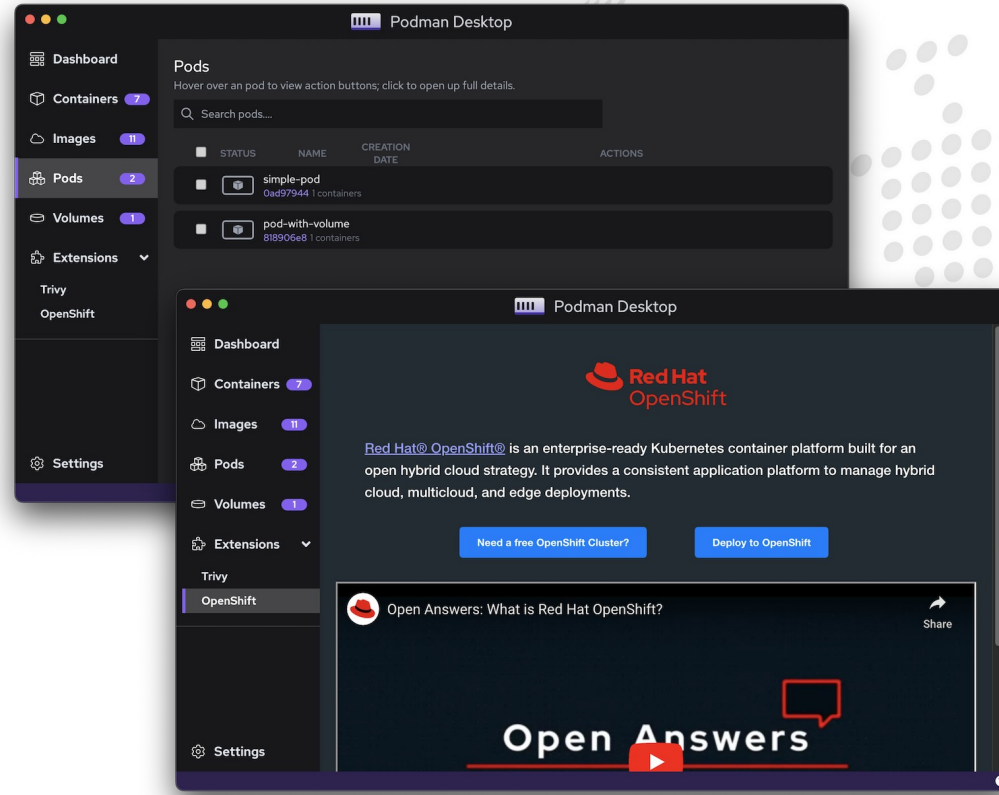
```
[daniel@kvm ~]$  
[daniel@kvm ~]$
```

# Pod

```
[daniel@kvm ~]$  
[daniel@kvm ~]$ podman generate kube web >> web.yaml
```

```
[daniel@kvm ~]$ cat web.yaml  
# Save the output of this file and use kubectl create -f to import  
# it into Kubernetes.  
#  
# Created with podman-4.2.0  
apiVersion: v1  
kind: Pod  
metadata:  
  annotations:  
    io.kubernetes.cri-o.ContainerType/dbwp: container  
    io.kubernetes.cri-o.ContainerType/webwp: container  
    io.kubernetes.cri-o.SandboxID/dbwp: 1cbb80af0601b410ab3684084efdbc18beba75f1be3491ba248b2d7b34c0339  
    io.kubernetes.cri-o.SandboxID/webwp: 1cbb80af0601b410ab3684084efdbc18beba75f1be3491ba248b2d7b34c0339  
    io.kubernetes.cri-o.TTY/dbwp: "false"  
    io.kubernetes.cri-o.TTY/webwp: "false"  
    io.podman.annotations.autoremove/dbwp: "FALSE"  
    io.podman.annotations.autoremove/webwp: "FALSE"  
    io.podman.annotations.init/dbwp: "FALSE"  
    io.podman.annotations.init/webwp: "FALSE"  
    io.podman.annotations.privileged/dbwp: "FALSE"  
    io.podman.annotations.privileged/webwp: "FALSE"  
    io.podman.annotations.publish-all/dbwp: "FALSE"  
    io.podman.annotations.publish-all/webwp: "FALSE"  
  creationTimestamp: "2022-11-01T15:24:37Z"  
  labels:  
    app: web  
    name: web  
spec:  
  containers:  
    - args:  
        - mariadb  
      env:  
        - name: MYSQL_DATABASE  
          value: wp  
        - name: MYSQL_PASSWORD  
          value: webdpr3s  
        - name: MYSQL_ROOT_PASSWORD  
          value: podman  
        - name: MYSQL_USER  
          value: wordpress  
      image: docker.io/library/mariadb:latest  
      name: dbwp  
      ports:  
        - containerPort: 3306  
          hostPort: 3306  
        - containerPort: 80  
          hostPort: 8181  
      resources: {}  
      securityContext:  
        capabilities:  
          drop:  
            - CAP_MKNOD  
            - CAP_AUDIT_WRITE  
      volumeMounts:  
        - mountPath: /var/lib/mysql  
          name: 969aa3f8877f538ecf47d06070487ad170f6f4032d72e915b08b11280f1e7d-pvc
```

# Podman Desktop



# COCKPIT PODMAN

Q Search

Sistema

Visão geral

Logs

Armazenamento

Rede

Podman containers

Virtual machines

Contas

Serviços

Ferramentas

Aplicações

Atualizações de Software

Dump do Kernel

Relatório de diagnóstico

SELinux

Terminal

> Show images

Containers

Show Only running Create container

web pod group

Running Create container in pod

Container	Owner	CPU	Memory	State
dbwp docker.io/library/mariadb:latest mariadb	user: daniel	n/a	n/a	Running

Details Integration Logs Console

```
# ls
bin boot dev docker-entrypoint-initdb.d etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
# ps
  PID TTY          TIME CMD
  138 pts/0    00:00:00 sh
  147 pts/0    00:00:00 ps
# ps ax
  PID TTY          STAT TIME   COMMAND
    1 ?            Ssl   0:00 mariadb
  138 pts/0    Ss     0:00 /bin/sh
  148 pts/0    R+     0:00 ps ax
#
```

> webwp  
docker.io/library/wordpress:latest  
apache2-foreground

user: daniel

n/a

n/a

Running

# INFOS

<https://podman.io/>

<https://cockpit-project.org/>



# Dúvidas?

*Contato:*

[daniellarasouza@yahoo.com.br](mailto:daniellarasouza@yahoo.com.br)