

embarcadero®

Delphi Academy

Consejos prácticos, trucos y técnicas



Presentando Containers y Docker para Devs

Fernando Rizzato
Lead Software Consultant, Latin America

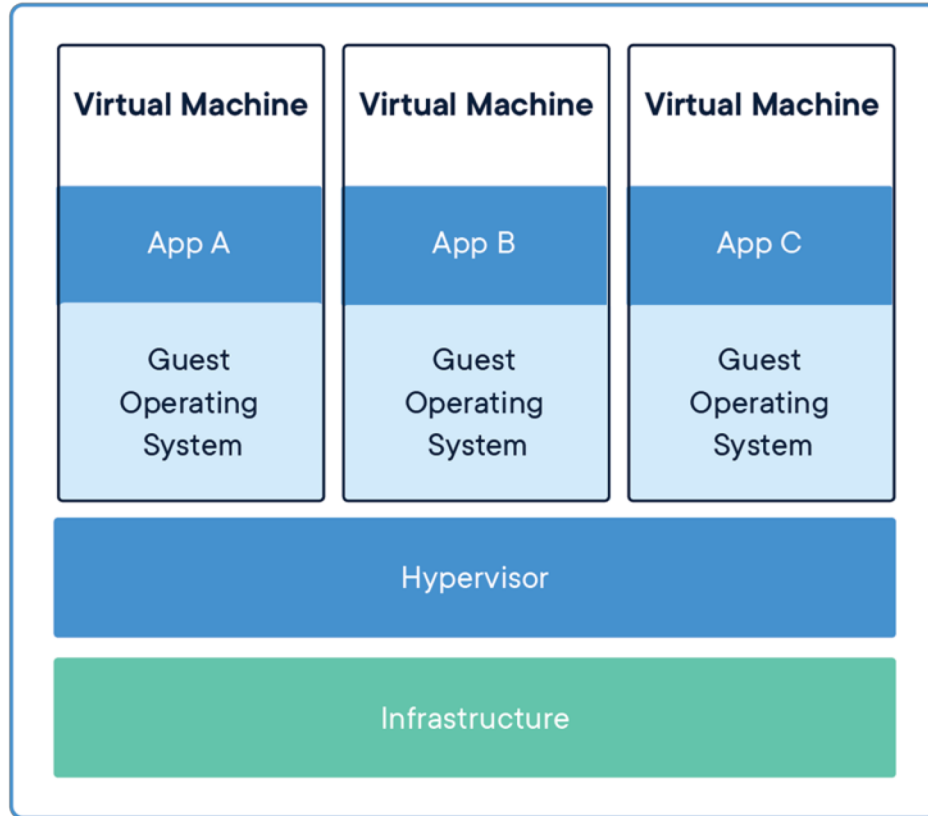
AGENDA

- Máquinas Virtuales vs Containers
- Tipos de Containers
- Docker
 - Conceptos Básicos
 - Instalación y Configuración
 - Ejecución
- Recursos Adicionales

MÁQUINAS VIRTUALES

- Máquinas virtuales (VM) son una abstracción de hardware físico, transformando un hardware en varios
- El hipervisor permite que varias máquinas virtuales se ejecuten en una sola máquina
- Cada VM incluye una copia completa de un sistema operativo, aplicaciones, etc., y son 100% aisladas
- Una VM puede ocupar algunas decenas de GBs
- Las máquinas virtuales también pueden ser lentas para cargar
- *VMWare, VirtualBox, Parallels, Xen, Hyper-V*

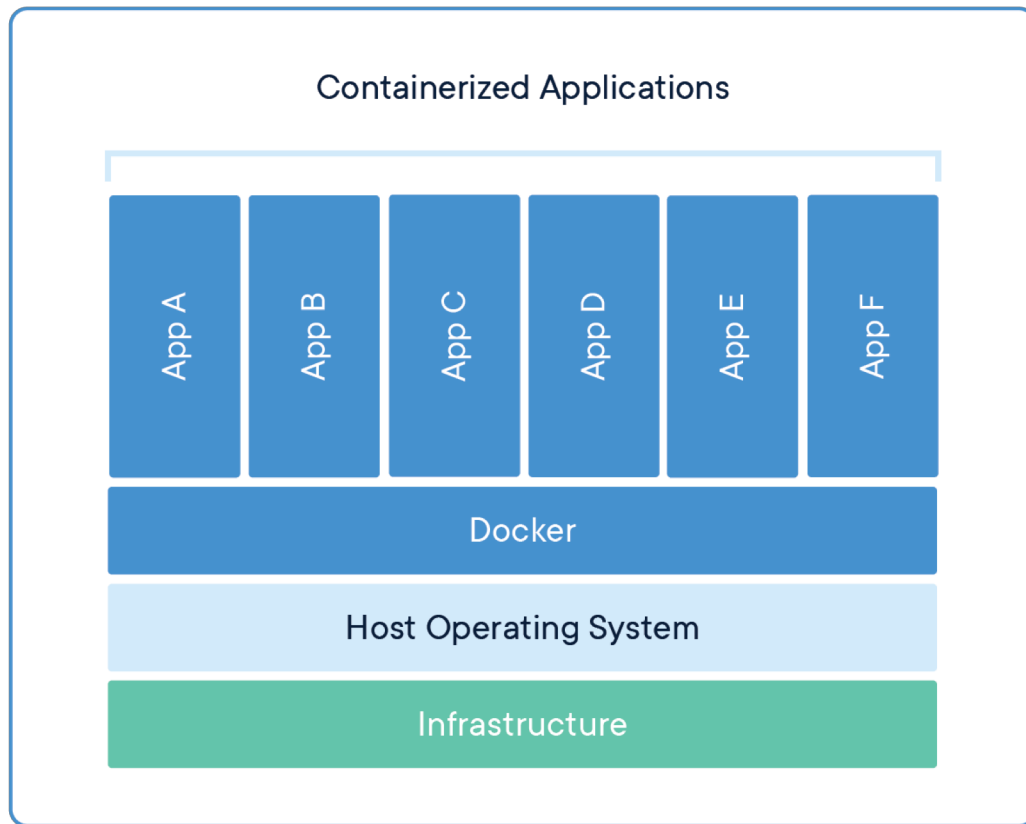
MÁQUINAS VIRTUALES



CONTAINERS

- Containers son una abstracción a nivel de aplicación
- Varios containers se pueden ejecutar en el mismo SO, y compartir el núcleo del mismo
- Cada container se ejecutará como un proceso aislado en el espacio del usuario
- Los containers ocupan menos espacio que las máquinas virtuales (decenas de MB)
- Los containers se pueden ejecutar dentro de las máquinas virtuales
- *Docker, RKT, Mesos, LXC / LXD, OpenVZ*

CONTAINERS



MÁQUINAS VIRTUALES VS CONTAINERS

- Considere un sistema hipotético de 10 GB
- En el modelo de VM, tendremos estos 10 GB multiplicados por el número de instancias necesarias
- En el modelo de contenedores, vamos a compartir una misma imagen con cientos de instancias
- Obviamente hay pros y contras en cada modelo
- Si el requisito es el aislamiento completo, entonces VM parece ser el mejor camino
- Si aislar procesos es suficiente, entonces los contenedores son más escalables y rápidos de configurar (aprovisionamiento)

TIPOS DE CONTAINERS

- Mientras "Docker" se convirtió en sinónimo de containers, eso no significa que es el único tipo disponible.
- Containers de sistema (Linux solamente)
 - LXC, LXD - <https://linuxcontainers.org/>
 - Este tipo de container hace que un único sistema Linux se ejecute como muchas instancias
 - Son más "parecidos" con VMs con relación a la gestión de recursos
- Containers de Aplicación (Linux, Windows, MAC)
 - Docker (<https://www.docker.com>), RKT (<https://coreos.com/rkt>)
 - La idea básica es que cada aplicación se ejecuta en un contenedor

DOCKER – CONCEPTOS BÁSICOS

- Un contenedor se inicia ejecutando una imagen
- Una imagen es un paquete que incluye todo lo necesario para ejecutar una aplicación
- Por lo tanto, un contenedor es una instancia de una imagen
- Los usuarios de Docker almacenan imágenes en repositorios privados o públicos
- Docker-Hub es un registro de Docker para imágenes públicas y privadas (<https://hub.docker.com>)

DOCKER – CONCEPTOS BÁSICOS

- *docker run* – Runs a command in a new container
- *docker start* – Starts one or more stopped containers
- *docker stop* – Stops one or more running containers
- *Docker ps* - Shows running containers
- *docker build* – Builds an image from a Docker file
- *docker pull* – Pulls an image or a repository from a registry
- *docker push* – Pushes an image or a repository to a registry
- *docker export* – Exports a container's filesystem as a tar archive
- *docker exec* – Runs a command in a run-time container
- *docker search* – Searches the Docker Hub for images
- *docker attach* – Attaches to a running container
- *docker commit* – Creates a new image from a container's changes

DOCKER - INSTALACIÓN

- Docker Desktop for Windows
 - <https://docs.docker.com/docker-for-windows/install/>
- Docker Desktop for macOS
 - <https://docs.docker.com/docker-for-mac/install/>
- Docker Server
 - <https://docs.docker.com/install/linux/docker-ce/ubuntu/>
 - <https://docs.docker.com/install/linux/docker-ce/fedora/>

DOCKER - PRIMERA IMAGEN

- *docker run --name my-redis -it ubuntu:latest bash*
- *apt-get update*
- *apt-get install wget*
- *apt-get install build-essential tcl8.5*
- *wget <http://download.redis.io/releases/redis-stable.tar.gz>*
- *tar xzf redis-stable.tar.gz*
- *cd redis-stable, make, make install*
- *./utils/install_server.sh*
- *service redis_6379 start*

DOCKER – EL PRIMER DOCKERFILE

- En una nueva carpeta, cree un archivo denominado **Dockerfile** con el contenido siguiente:

```
FROM ubuntu:latest
RUN ln -fs /usr/share/zoneinfo/America/Sao_Paulo /etc/localtime
RUN apt-get update
RUN apt-get install -y wget
RUN apt-get install -y build-essential tcl8.5
RUN wget http://download.redis.io/releases/redis-stable.tar.gz
RUN tar xzf redis-stable.tar.gz
RUN cd redis-stable && make && make install
RUN ./redis-stable/utils/install_server.sh
EXPOSE 6379
ENTRYPOINT ["redis-server"]
```

- `docker build -t redis .`
- `docker run -d -p 6379:6379 redis`

DEMOS

RECURSOS ADICIONALES

Documentación

- <https://docs.docker.com/get-started/>
- <https://www.docker.com/resources/what-container>

Blogs

- <http://techgenix.com/getting-started-with-containers/>
- http://www.macoratti.net/19/01/intro_docker2.htm
- <http://prof.valiante.info/disciplinas/hardware/maquinas-virtuais-e-containers>
- <http://www.sinestec.com.br/blog/docker-fundamentos-conceito-e-arquitetura/>
- <https://blog.aquasec.com/a-brief-history-of-containers-from-1970s-chroot-to-docker-2016>
- <https://www.comparitech.com/net-admin/docker-vs-virtual-machines/>

GRACIAS!

Preguntas?

Me puedes encontrar en:

@FernandoRizzato

fernando.rizzato@embarcadero.com

Síguenos en

fb.com/EMBTLatAm