

Breve introducción a Docker

Nahuel Defossé

Indice

Contenedores

Los contenedores permiten a las aplicaciones:

- ▶ Simplificar su desarrollo
 - ▶ Evita el **funciona en mi máquina**. Evita discrepancias entre los entornos.
- ▶ Simplificar su distribución.
- ▶ Aprovechar mejor los recursos
 - ▶ Un contenedor es como una VM **liviana**. Permite aprovechar mejor el CPU, memoria y disco.
- ▶ Facilitan escalamiento horizontal.

Máquinas virtuales. . .

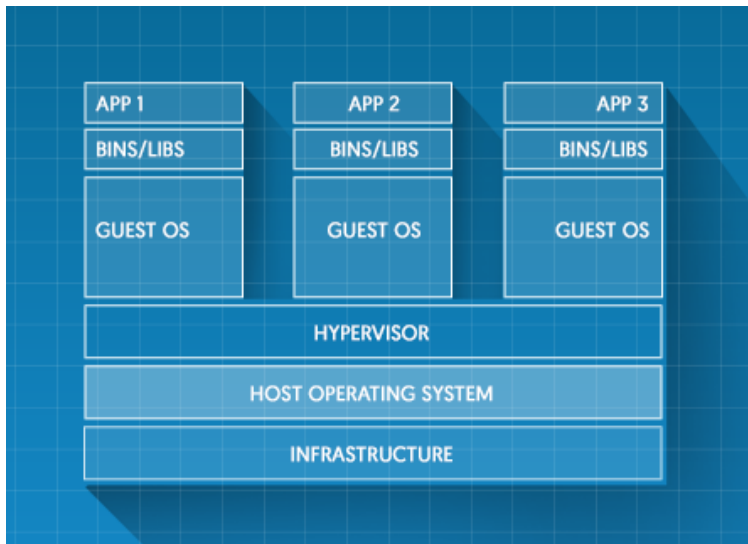


Figura1: Aplicaciones en ejecución con VM

... y contenedores

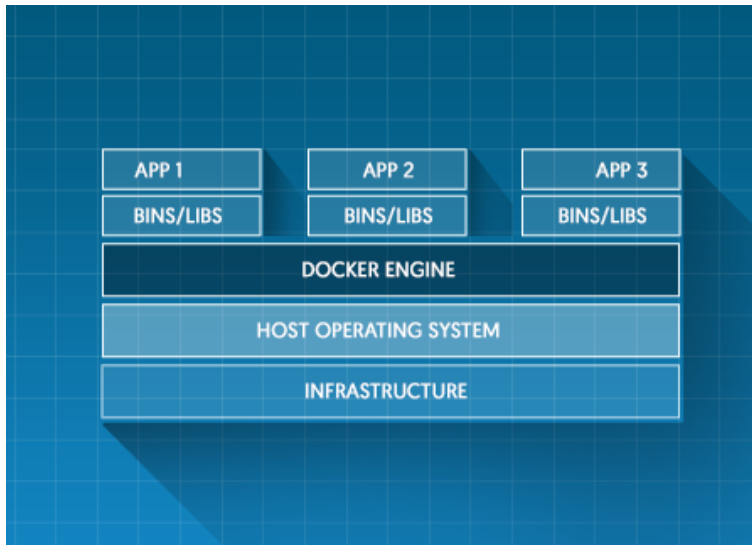


Figura2: Aplicaciones en ejecución con VM

Instalación

Linux

- ▶ Disponible en el sistema de paquetes
- ▶ Guía de Instalación en Ubuntu

Instalación

Windows

Docker Toolbox

- ▶ `docker-machine` + Virtualbox + Kitematic
- ▶ **Windows 10** Utilizando docker nativo

Docker for Windows

- ▶ **Beta** con emulación nativa HyperV (Windows 10)
- ▶ <https://docs.docker.com/docker-for-windows/>

Instalación Windows (2)

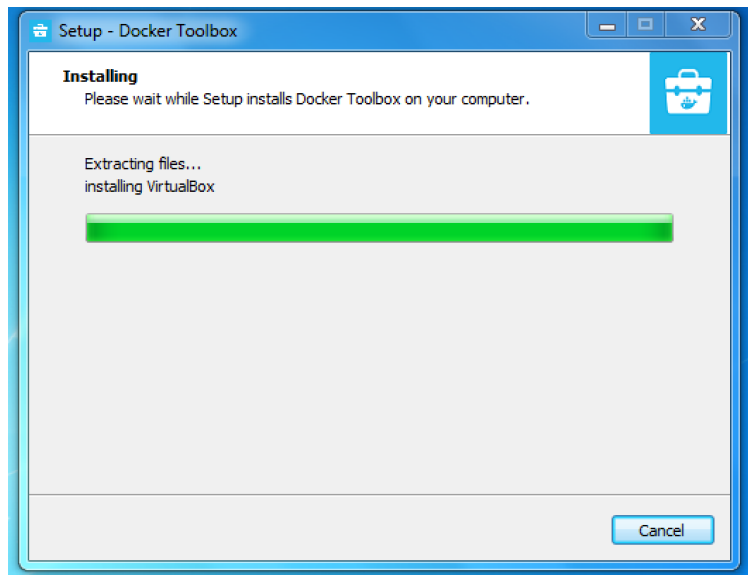


Figura3: Instalación de Toolbox

Instalación

Mac

Docker Toolbox

- ▶ `docker-machine` + Virtualbox + Kitematic

Docker for Mac

- ▶ Beta con emulación nativa de OS X

Windows y Mac

En Linux la ejecución de Docker es nativa, pero en windows, necesitamos un soporte que nos provee docker-machine en formato de máquina virtual.

Creación de una máquina virtual con docker-machine

- ▶ `docker-machine create --driver virtualbox vm`
- ▶ `eval $(docker-machine env vm)` o `docker-machine`

Primer ejecución de un contenedor

Ejecución bash en una imagen **debian** ¹

```
docker run -ti debian bash
```

La salida

```
root@f601df7b7dd9:/# whoami
```

```
root
```

```
root@f601df7b7dd9:/# pwd
```

```
/
```

```
root@f601df7b7dd9:/# ps
```

PID	TTY	TIME	CMD
1	?	00:00:00	bash
8	?	00:00:00	ps

¹-ti tty e interactivo

¿Qué ocurrió?

docker run

Dockerfile

Un dockerfile es la definición

Docker - Contenedores vs Imágenes

Imagen

- ▶ Una *imagen* es la definición del sistema de archivos para un contenedor.
- ▶ Una imagen suele tener un nombre y una etiqueta que identifica su versión.
- ▶ `docker pull busybox ()`
- ▶ `docker images`

Contenedor

- ▶ Es una instancia de ejecución de una imagen.
- ▶ `docker run -ti busybox sh`
- ▶ `docker run -ti ubuntu /bin/bash`

Dockerfile

Ejemplo:

```
FROM ubuntu:16.04
RUN apt-get install python2
RUN useradd foo
COPY miscript.py /home/foo/miscript.py
RUN chown /home/foo/miscript.py
RUN chmod +x /home/foo/miscript.py
CMD ["python2", "/tmp/miscript.py"]
```

Creando la imagen

- ▶ `docker build -t miproyecto .`
- ▶ `docker run`

Filesystem Read Only

Volumes

- ▶ Los volúmenes son la forma de tener persistencia en Docker.

Commit

- ▶ Permite editar los pasos realizados en una imagen doc

Docker Compose

docker-compose permite definir en un archivo JSON, YAML o INI una configuración de uno o mas contenedores.

```
version: '2'
services:
  web:
    build: .
    ports:
      - "5000:5000"
    volumes:
      - ./code
    links:
      - redis
  redis:
    image: redis
```

Demo

Gracias