

docker



Humberto Duarte Ortiz

Ing. en Sistemas Computacionales

Instituto Tecnológico de Mexicali

- 17 años trabajando en el área de TI
- 15 en CFE.

Desarrollo Aplicaciones Backend/Front End.

Fan de Linux desde 1995

Agenda



- Antecedentes
- Contenedores
- ¿Qué es Docker?
- Alcance
- DEMO



Antecedentes

Movilidad de aplicaciones

Hasta algunos años, el desarrollo de software no tenía serios problemas de “movilidad”.

Pero en la última década, la situación ha cambiado radicalmente:

- Adopción de metodologías ágiles.
- Frameworks de desarrollo.
- Cloud Computing.
- Dispositivos móviles.

Muchos entornos productivos deberán correr sobre varias infraestructuras diferentes, bien por motivos legales, rendimiento o confiabilidad.



Antecedentes

Una Aplicación en un servidor físico



Antecedentes



Una Aplicación en un servidor físico



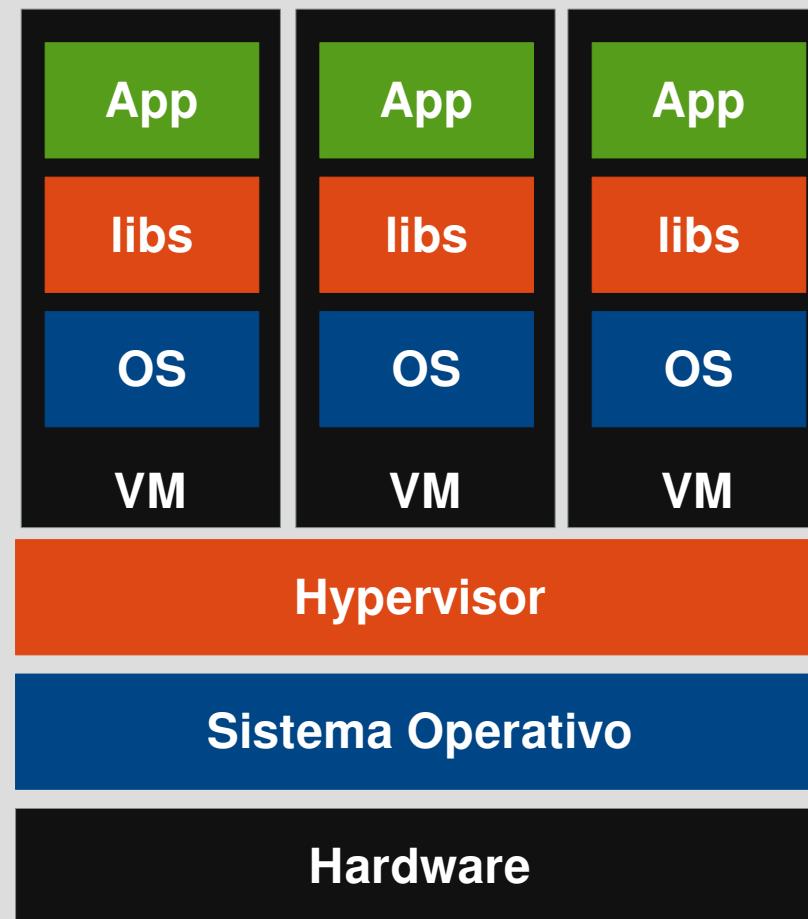
Problemas:

- Tiempo de implementación.
- Costo.
- Desperdicio de Recursos.
- Dificultad para crecer.
- Difícil de migrar.



Antecedentes

Virtualización (Hypervisor-based)





Antecedentes

Virtualización (Hypervisor-based)

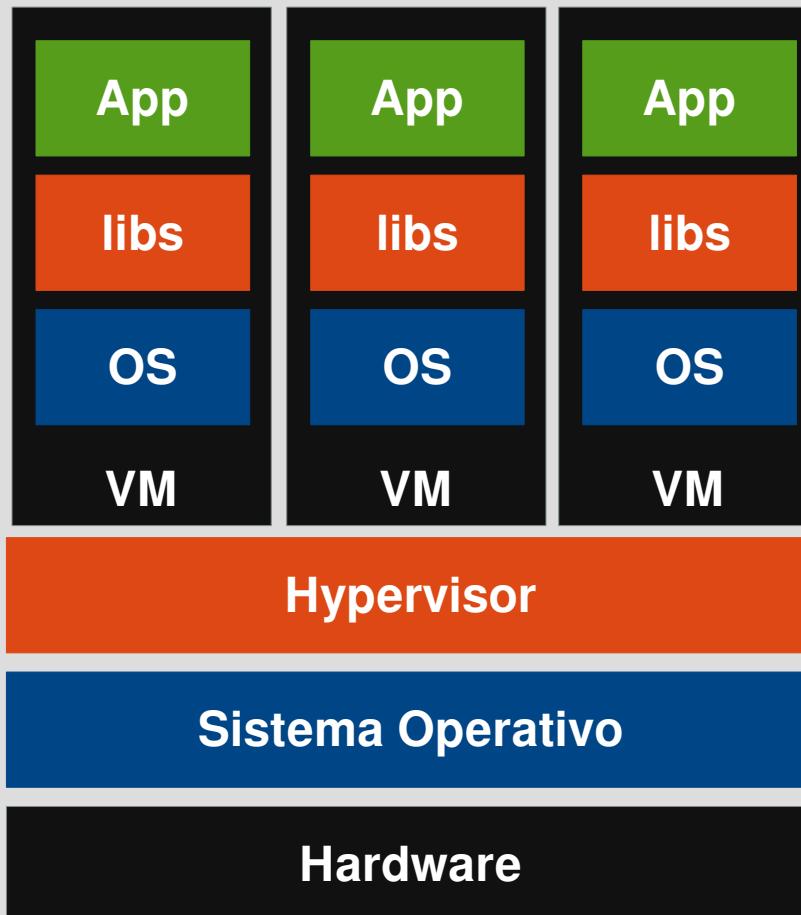


Beneficios:

- Mayor aprovechamiento de recursos.
- Facilidad para crecer.
- Aislamiento de Aplicaciones.
- Reducción de Costos.
- Menor tiempo de Implementación.
- Disponible en la nube.

Antecedentes

Virtualización (Hypervisor-based)



Límites:

- Cada VM aún necesita su propio CPU, RAM, Disco y un OS completo.
- Mientras mas VM, necesitamos más recursos.
- Desperdicio de Recursos.



Contenedores

En la virtualización basada en contenedores, se interactúa directamente con el kernel del Sistema Operativo para crear múltiples Instancias o Entornos Virtuales aislados unos de otros. Cada VE o instancia es llamada **Contenedor** o **Linux Container (LXC)**.

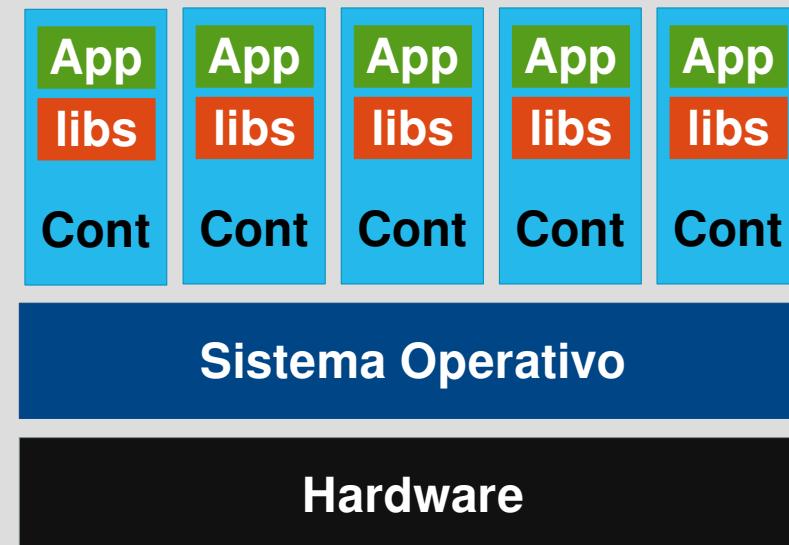
Cada contenedor tiene su propio:

- Sistema de Archivos
- Variables de entorno
- Procesos
- Memoria
- Dispositivos
- Puertos de Red



Contenedores

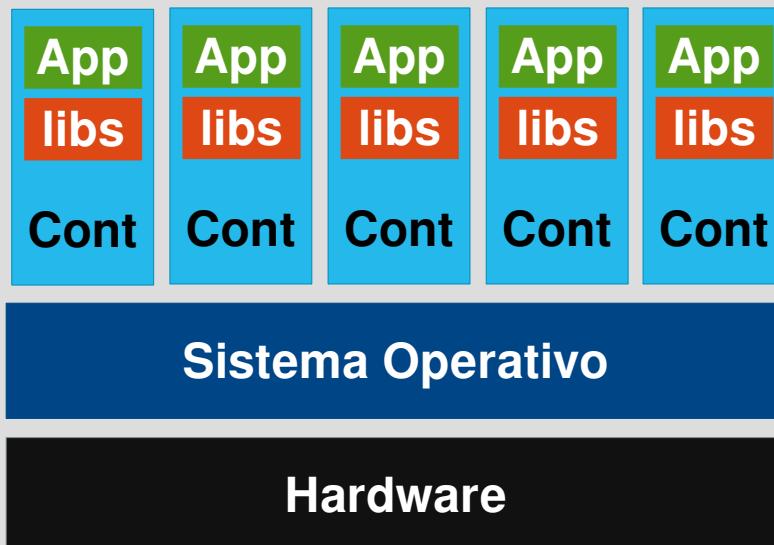
Entornos Virtuales (Contenedores)





Contenedores

Entornos Virtuales (Contenedores)



Beneficios:

- Más ligeros que una VM.
- No se virtualiza Hardware.
- No es necesario Instalar un OS.
- Mejor aprovechamiento de Recursos.
- Más contenedores por máquina.
- Alta portabilidad.
- Mayor rapidéz de implementación.
- Facilitan la movilidad de aplicaciones

Docker



¿Que es Docker?

Es un proyecto open source que permite crear fácilmente contenedores portables, ligeros y autosuficientes.

El objetivo es que el mismo contenedor que un desarrollador construye para escribir y probar una aplicación en su equipo pueda correr en producción a mayor escala, ya sea sobre máquinas virtuales, directamente sobre hardware, en clústeres o clouds públicas

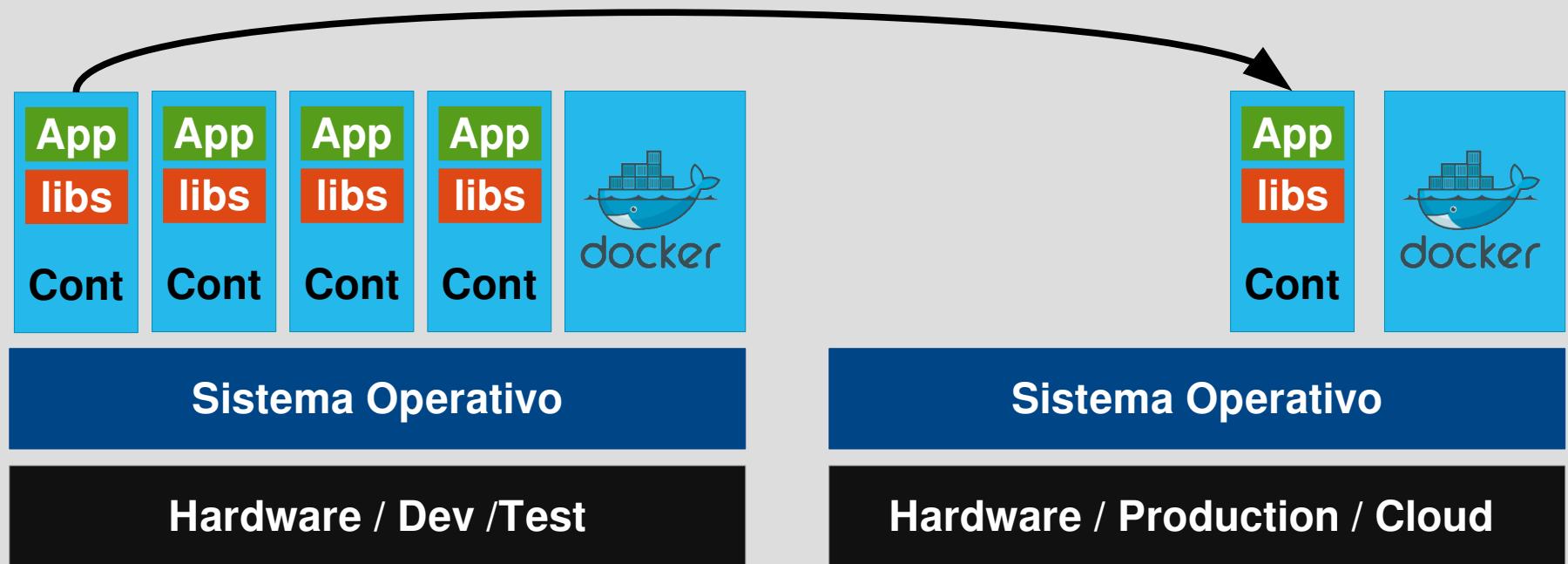
“Build, Ship and Run any App Anywhere”

Docker



DevOps

Developer - Operations



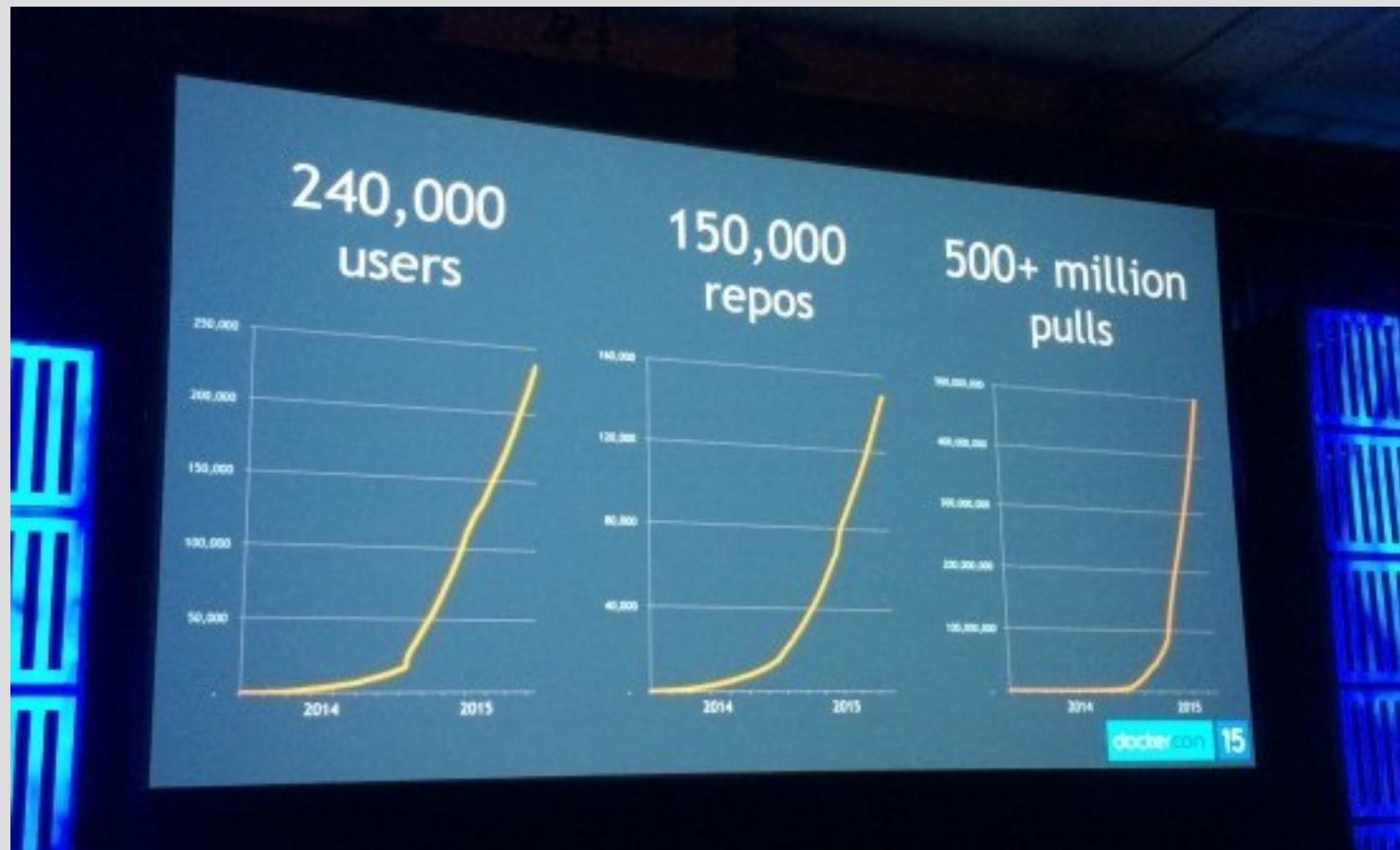
Los desarrolladores se dedican a Desarrollar y se olvidan de problemas de implementación.

Los SysAdmins (Operaciones) implementan con mayor rapidez y se olvidan de instalar dependencias para las aplicaciones.

Docker



Docker



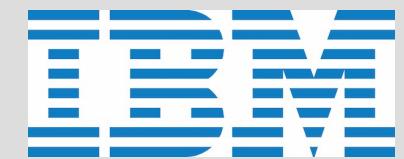
Docker



Docker



CANONICAL



Microsoft Azure

NGINX



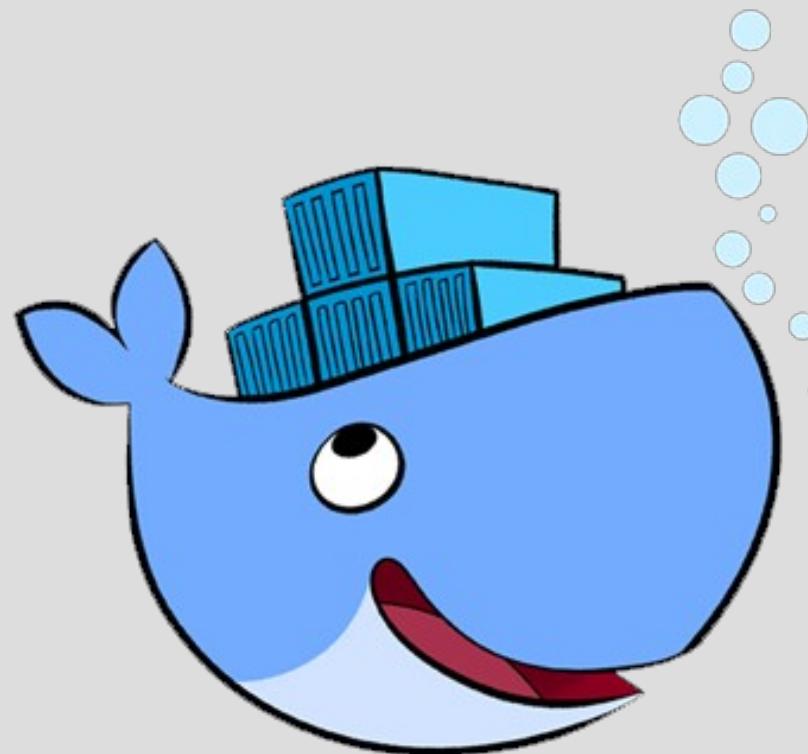
Productos de Docker

- Docker Engine
- Docker Hub
- Docker Machine
- Docker Swarm
- Docker Compose

Docker

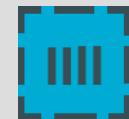


Docker Engine



Docker Engine

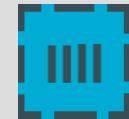
- Es la aplicación que permite crear, mover y ejecutar contenedores.
- Cliente – Servidor
- Interactua directamente con el kernel del OS para crear Entornos Aislados para cada contenedor.



Imagenes/Contenedores



- Las imágenes podemos entenderlas como templates de contenedores.
- Contiene las instrucciones necesarias y configuración para crear el contenedor (programas, archivos, puertos, links, volúmenes, etc...).
- Las podemos modificar y crear nuevas imágenes.
- Se almacenan localmente o en un repositorio remoto como Docker Hub.
- Podemos descargar imágenes creadas por otros usuarios e imágenes oficiales.



Imagenes/Contenedores



- Los contenedores, se crean a partir de una o varias imágenes.
- Un contenedor es una aplicación o serie de aplicaciones, en ejecución aislada.
- Puedo crear varios contenedores a partir de una imagen.
- Contiene todo lo necesario para ejecutar la aplicación (librerías, frameworks, etc..)

Docker Hub

hub.docker.com



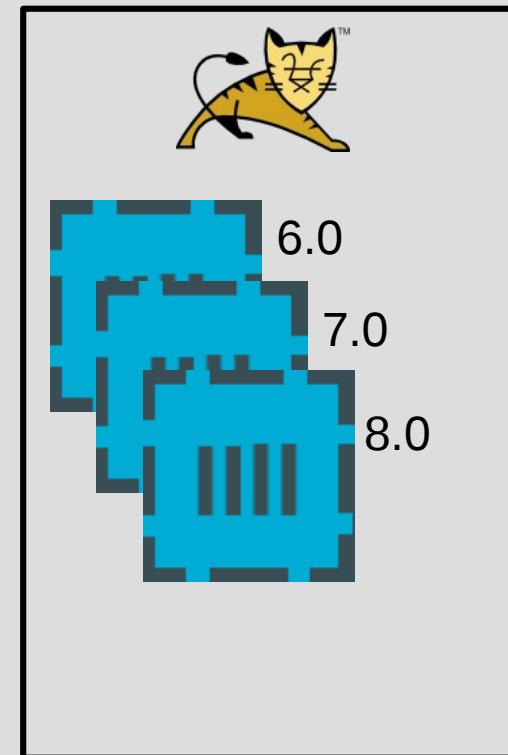
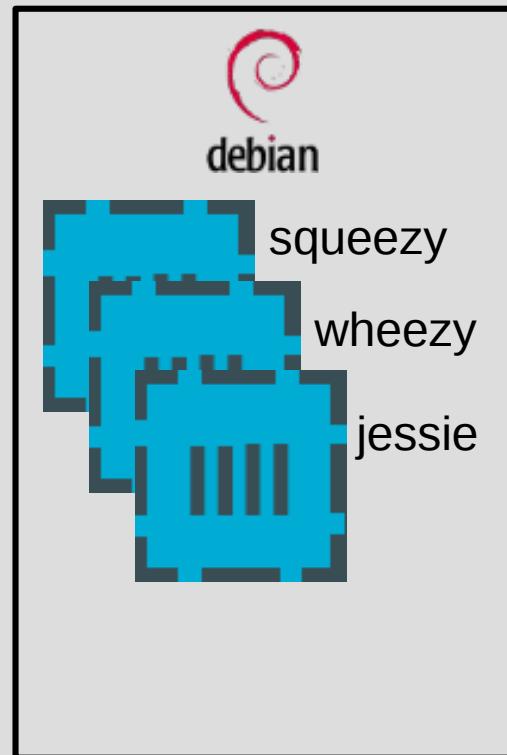
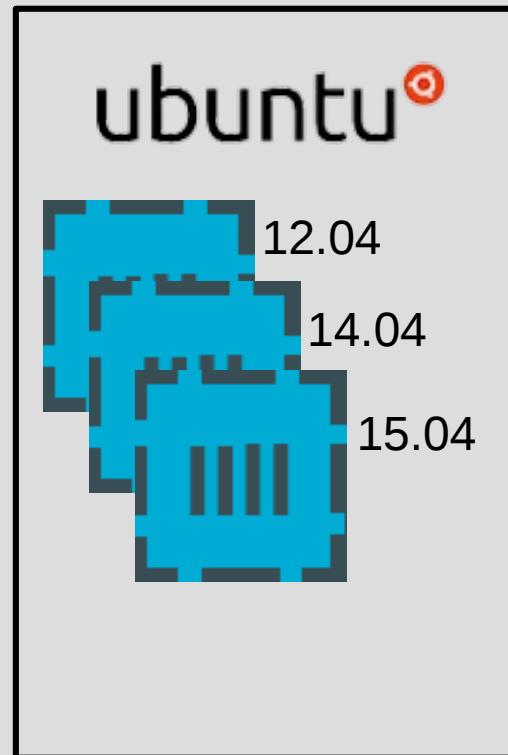
- Es un registro (registry) público que contiene imágenes disponibles para su uso.
- Las imágenes se almacenan en repositorios.
- Cada repositorio contiene una o mas imágenes identificadas con Tags.
- Podemos encontrar imágenes oficiales y no oficiales.

Docker



Docker Hub

hub.docker.com



Orchestration Tools

- Docker Machine



Herramienta que nos permite instalar remotamente docker engine en máquinas virtuales locales, remotas o servicios de nube y convertirlo en un docker host.

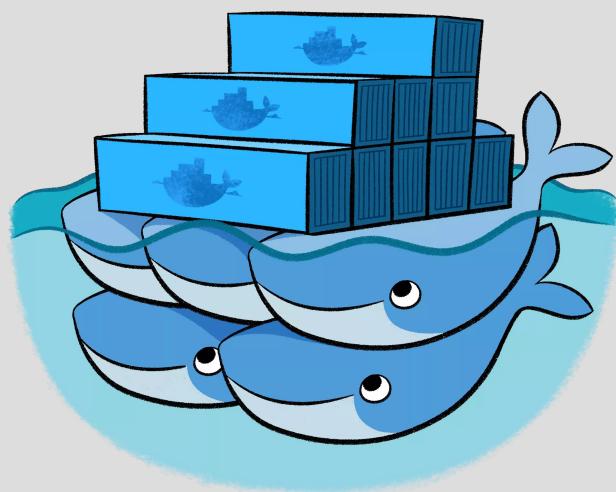
Se pueden crear multiples ambientes aislados con contenedores en el mismo equipo.

Docker



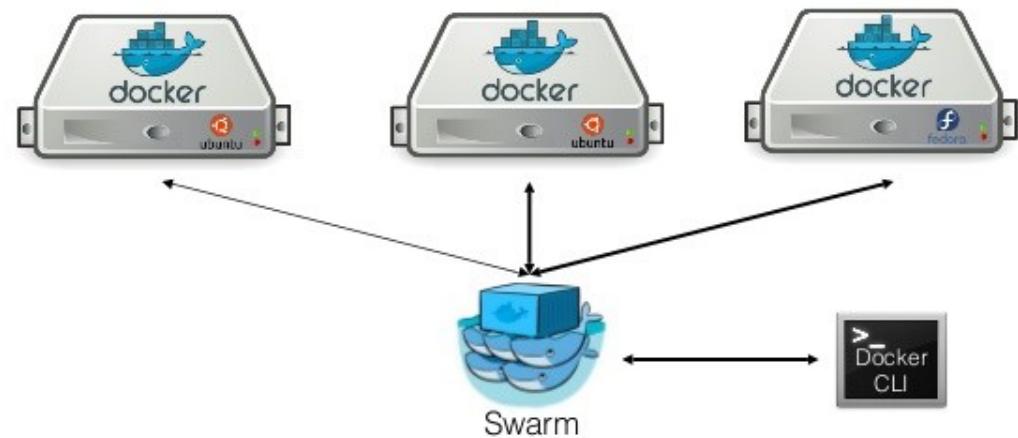
Orchestration Tools

- Docker Swarm



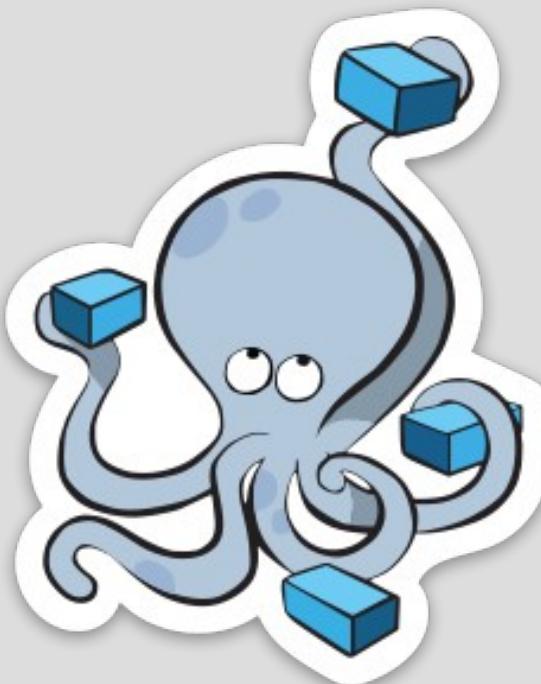
Permite administrar multiples docker hosts creando un cluster de alta disponibilidad.

With Docker Swarm



Orchestration Tools

Docker Compose.

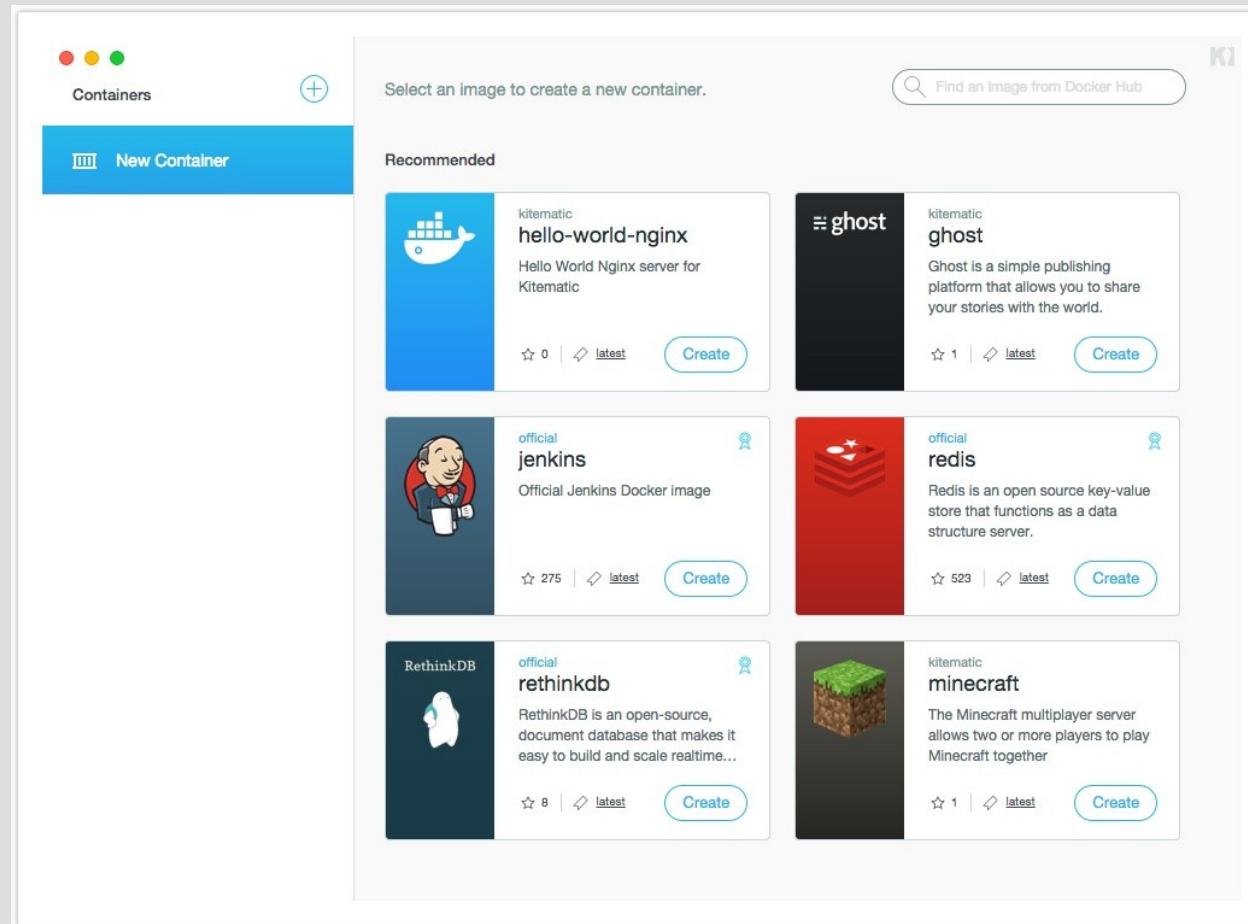


Permite definir y ejecutar aplicaciones multi-contenedores a través de archivos YAML de configuración en donde se definen todos los servicios que la aplicación requiere.

Se utiliza un solo comando junto para arrancar todos los servicios que se definieron en la configuración.

Orchestration Tools

GUI – Clientes Gráficos



Kitematic

- Windows
- Mac OSX

Docker



Orchestration Tools

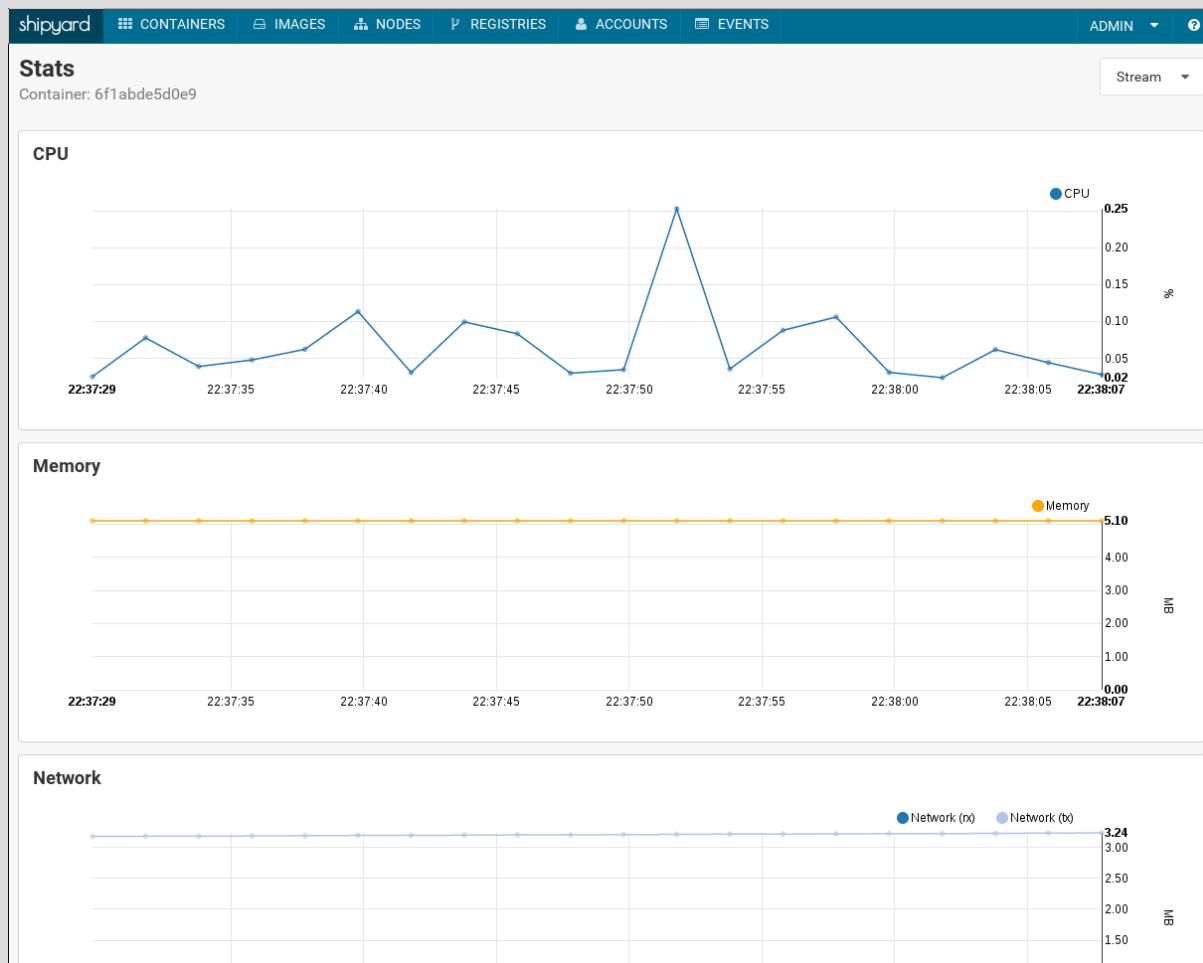
GUI – Clientes Gráficos

Dockerana (Multi-plataforma)



Orchestration Tools

GUI – Clientes Gráficos



Shipyard
• web

Docker

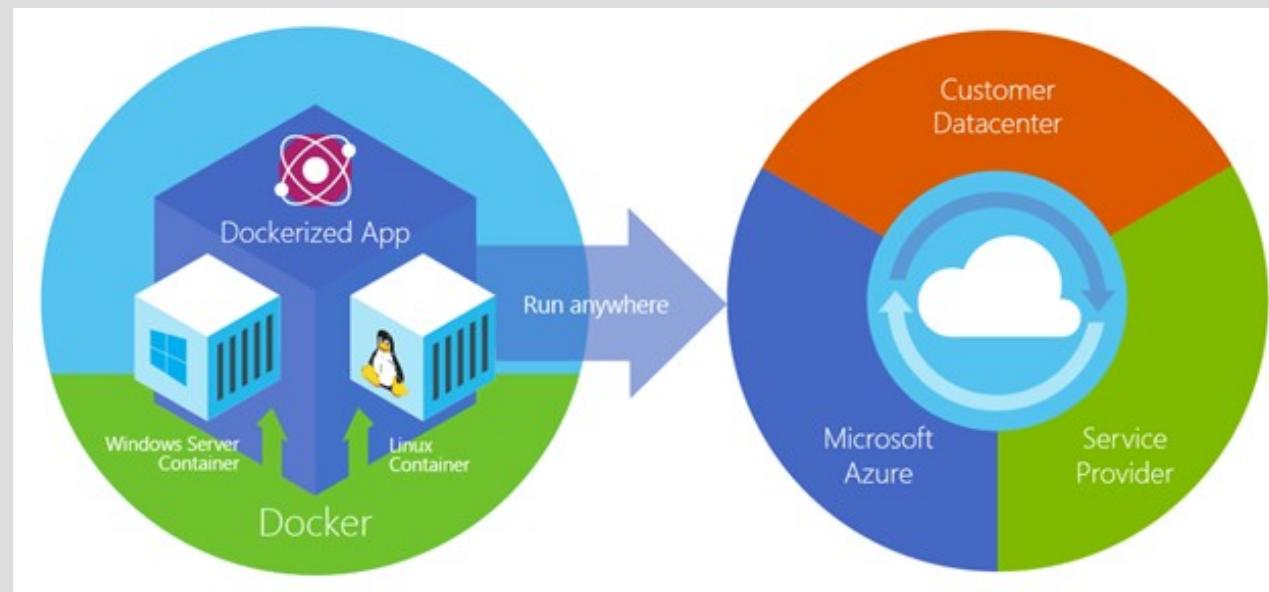


Microsoft?

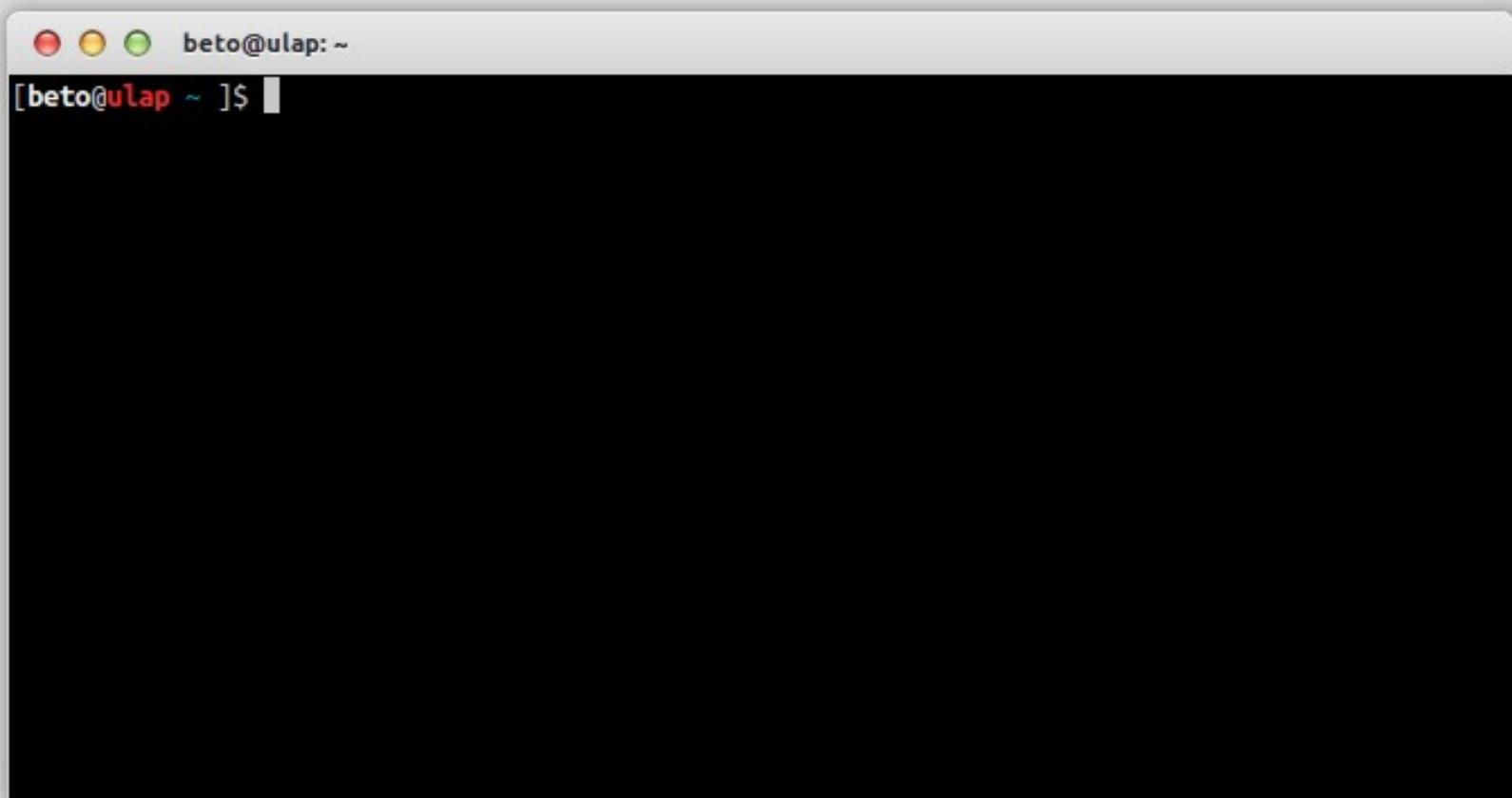
No funciona de forma nativa en Windows. Aún!

Se puede implementar en forma de Máquina Virtual y solo se pueden crear contenedores con aplicaciones que corran en Linux (python, java, nodejs, php, c, c++, go, etc).

Windows Server 2016 contará con soporte para crear Windows Server Containers.



Demo time!

A black rectangular placeholder representing a terminal window, positioned in the center of the slide. It is surrounded by a thin white border and has three colored window control buttons (red, yellow, green) at the top-left corner.

Gracias!



Mayor información en:
www.docker.com

Presentación disponible en:
www.github.com/hdortiz/docker_rocc2015.git

Humberto Duarte Ortiz
humberto.duarte@cfe.gob.mx
hduarte77@gmail.com