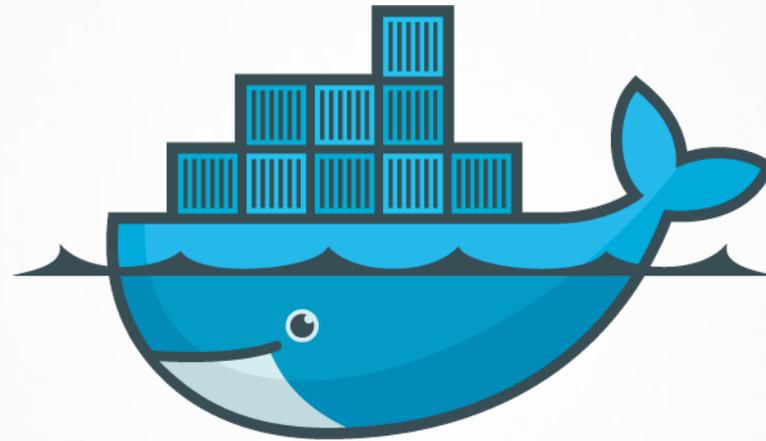
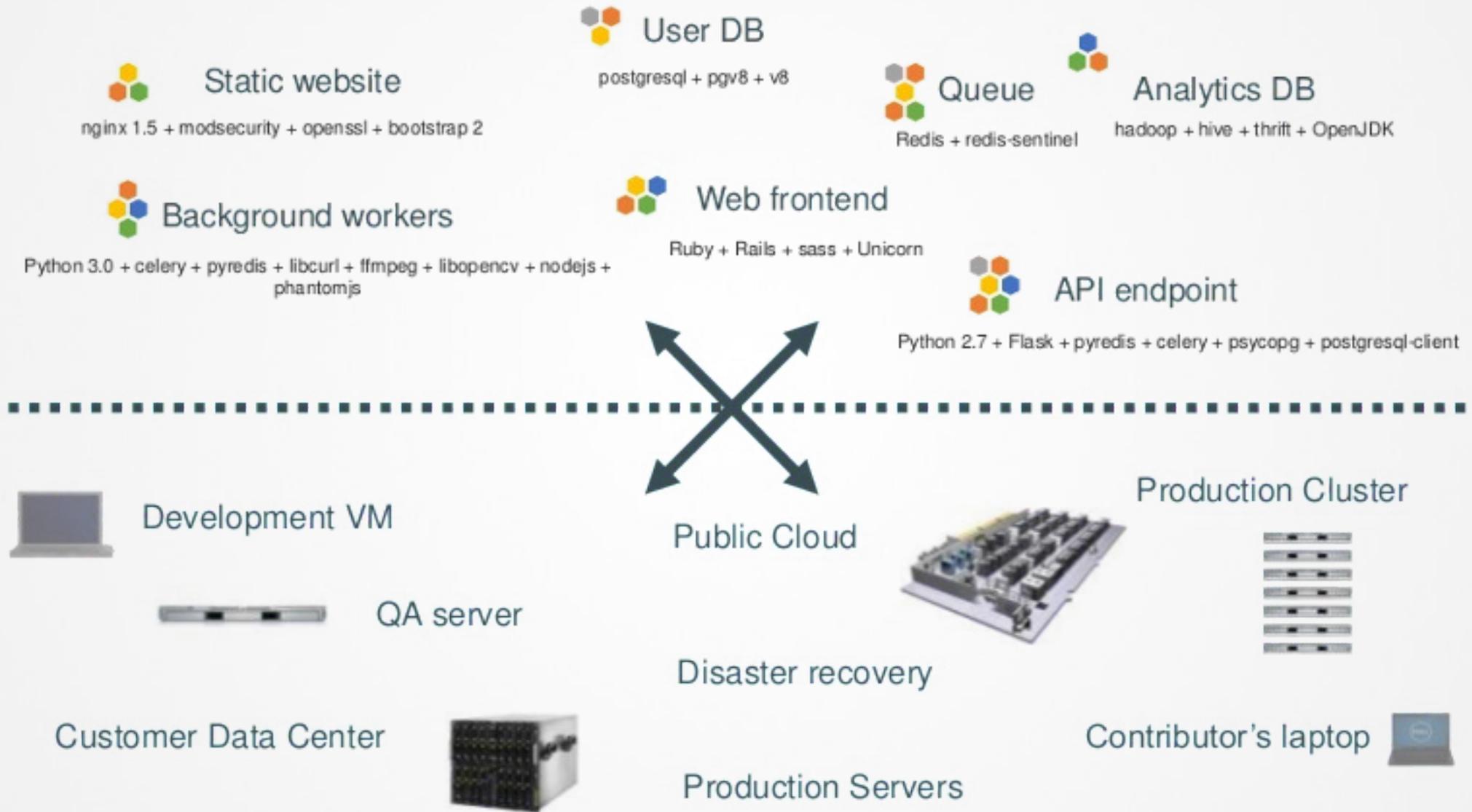


Introducción a



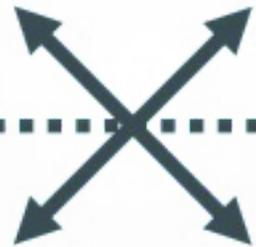
docker

WHY: ¿CUÁL ES EL RETO?



docker

WHY: ¿CUÁL ES EL RETO?



docker

WHY: LA SOLUCIÓN



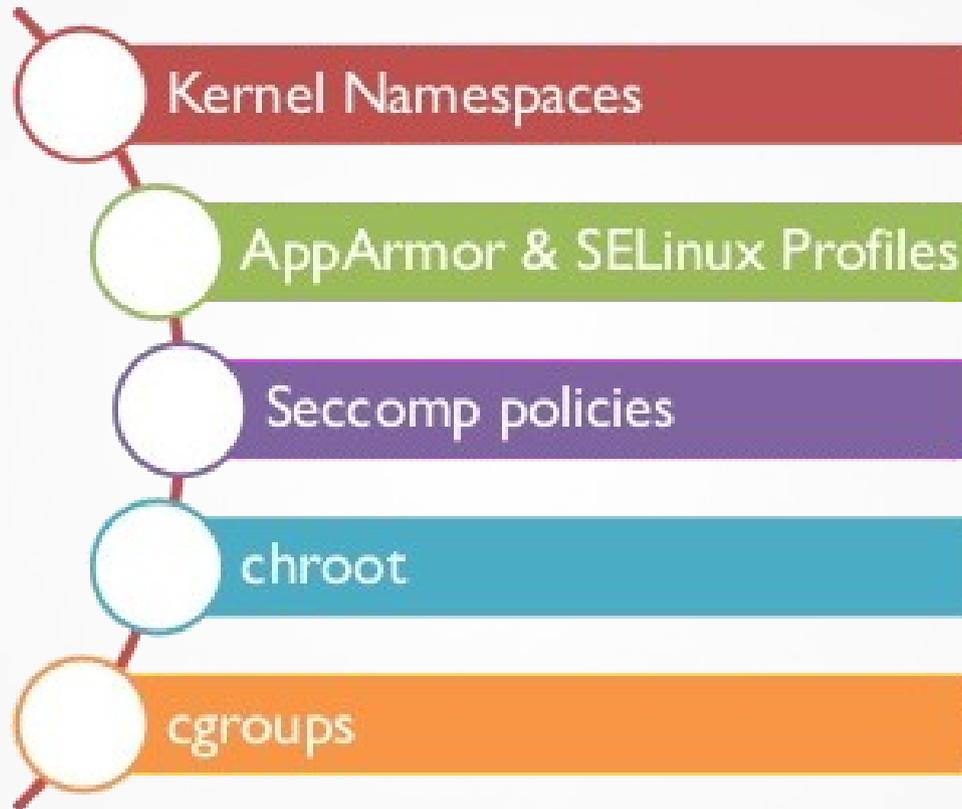
Docker es un motor que permite que encapsular cualquier información y hacerla portátil, autosuficiente y ligera



... de forma que se pueda manipular mediante operaciones estándar y ejecutar de manera consistente en cualquier hardware.

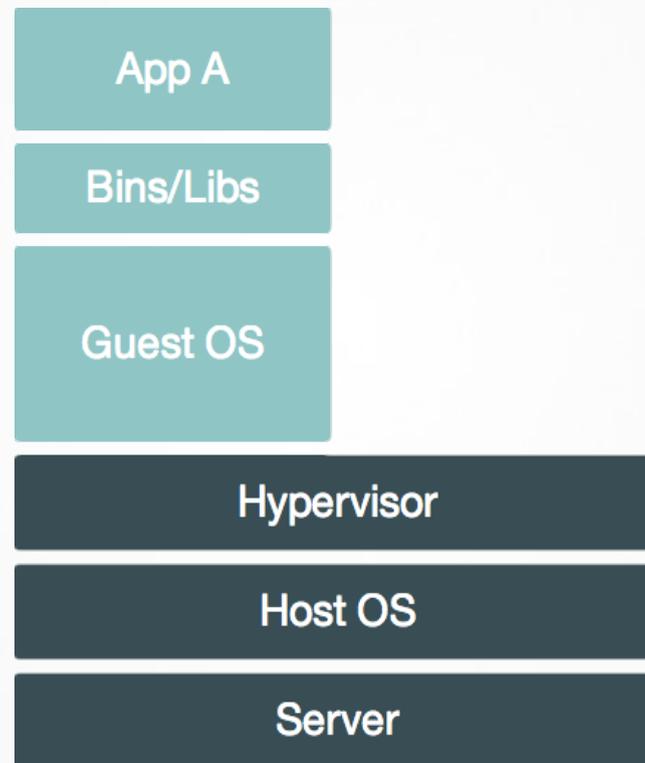
docker

HOW: ¿CÓMO LO HACE?



docker

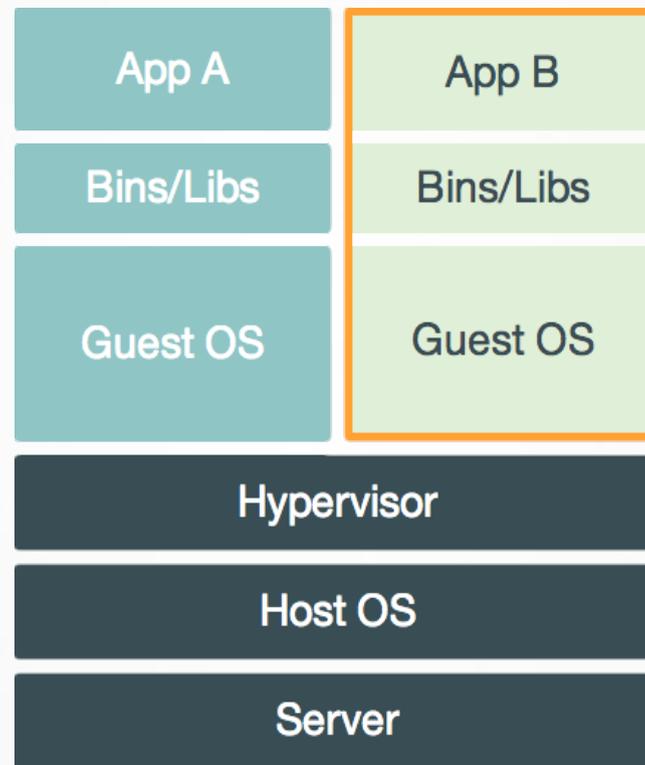
HOW: ARQUITECTURA MÁQUINAS VIRTUALES



Fuente: <http://www.felixcarmona.com/systems/understanding-linux-containers-64>

docker

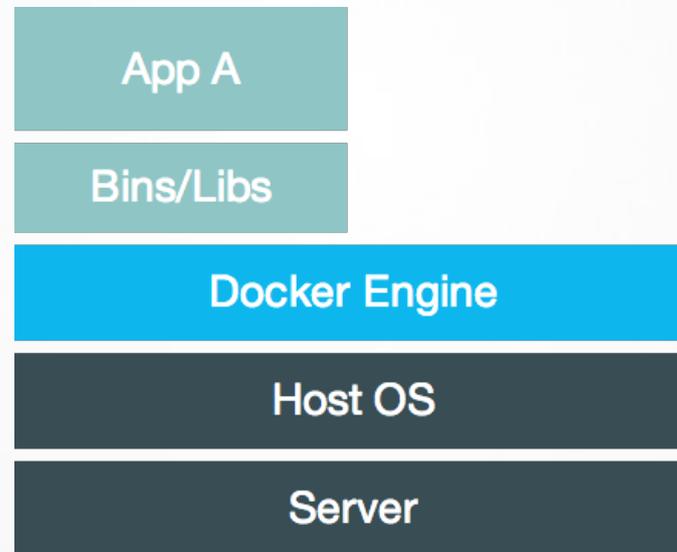
HOW: ARQUITECTURA MÁQUINAS VIRTUALES



Fuente: <http://www.felixcarmona.com/systems/understanding-linux-containers-64>

docker

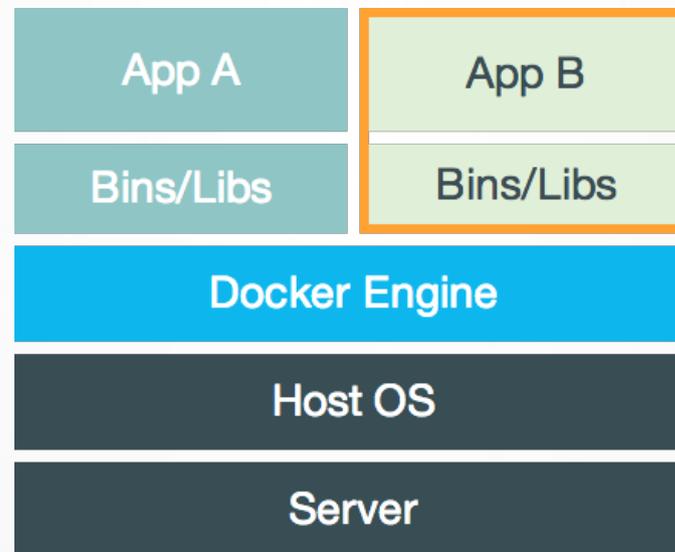
HOW: ARCHITECTURA **docker**



Fuente: <http://www.felixcarmona.com/systems/understanding-linux-containers-64>

docker

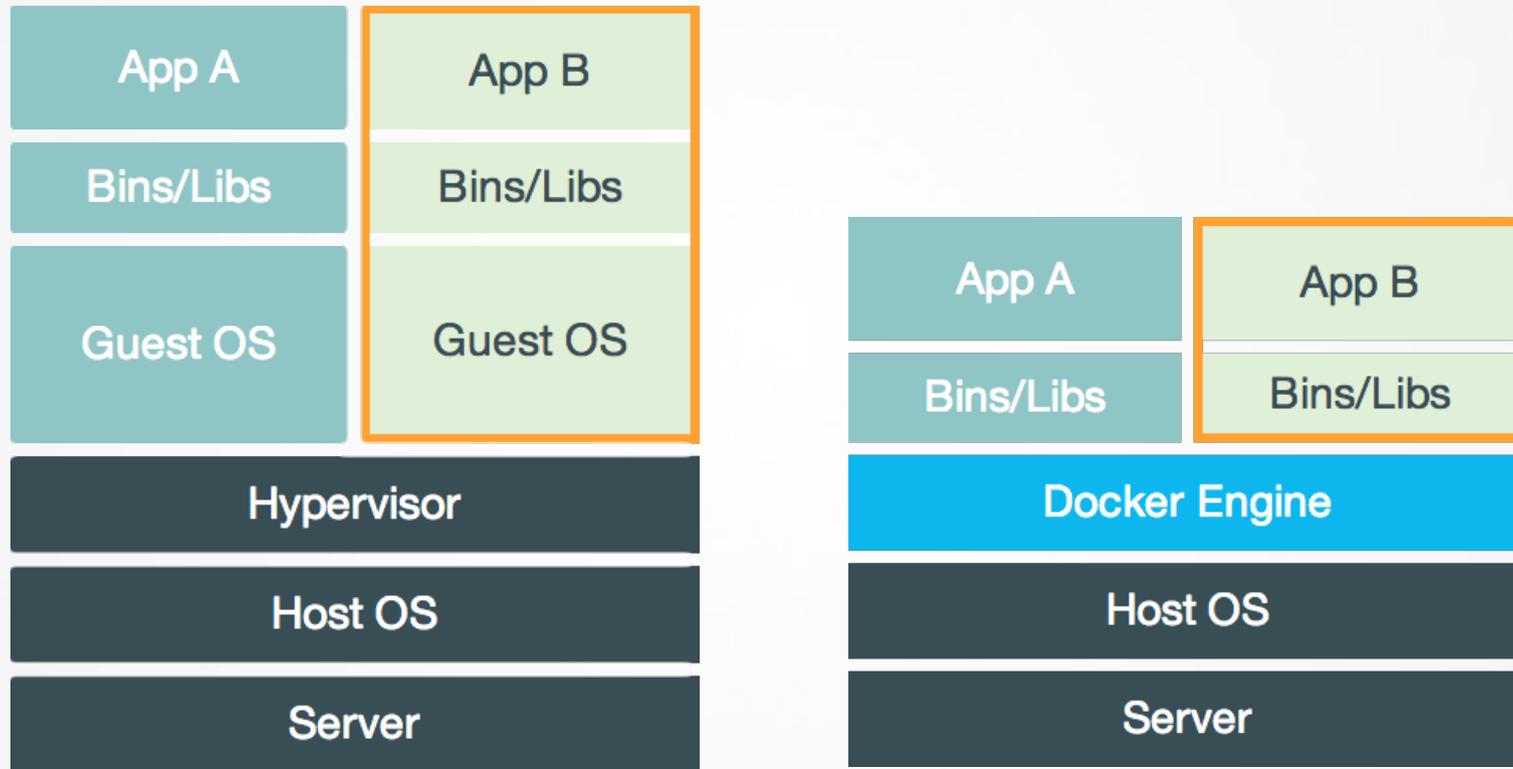
HOW: ARCHITECTURA docker



Fuente: <http://www.felixcarmona.com/systems/understanding-linux-containers-64>

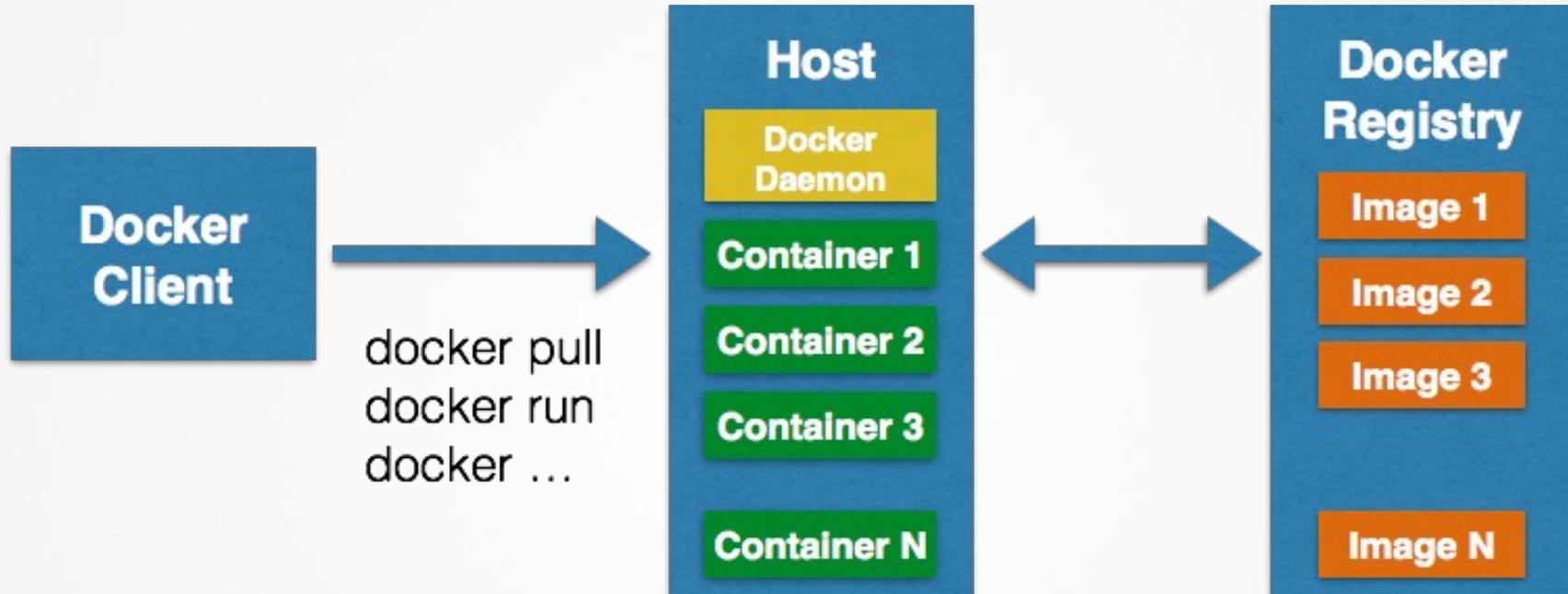
docker

HOW: MÁQUINA VIRTUAL VS docker



Fuente: <http://www.felixcarmona.com/systems/understanding-linux-containers-64>

HOW: COMPONENTES docker



Fuente: <http://blog.arungupta.me/getting-started-with-docker/>

docker

HOW: CONCEPTOS docker



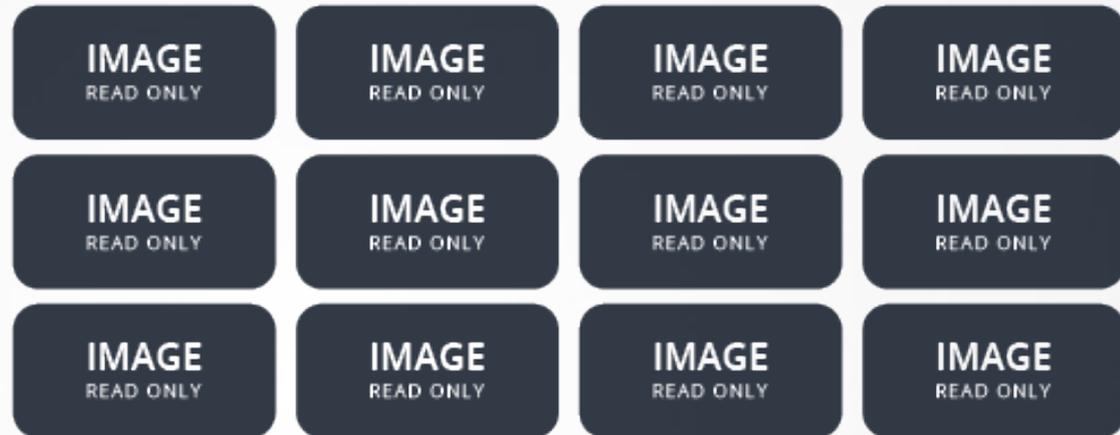
Docker Image

contains basic read-only image that forms the basis of container



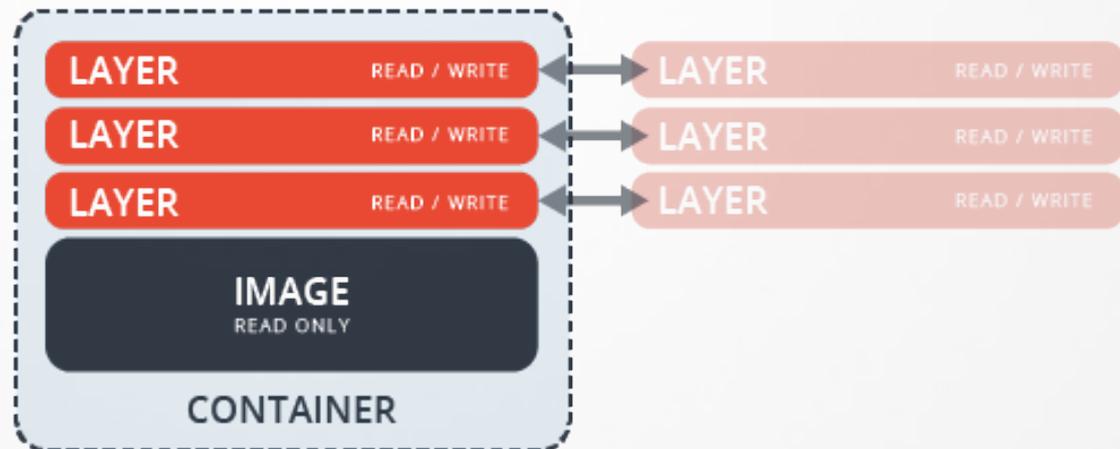
Docker Registry

a repository of images which can be hosted publicly (like Docker Hub) or privately and behind a firewall



Docker Container

is comprised of a base image with layers that can be swapped out so it's not necessary to replace the entire VM when updating an application



Fuente: <http://alandargan.com/using-vagrant-docker/>

docker



DEVOPS VS SYSOPS

WHAT: ¿QUÉ PUEDE HACER **docker** POR TI?



Construye una vez, ejecuta en cualquier lugar

- Entorno limpio, seguro y portable para apps.
- Ejecución de cada app de forma aislada en su propio contenedor.
- No tiene que preocuparse de dependencias o paquetes durante la fase de despliegue.
- Automatización de procesos de testing, integración y empaquetado mediante scripts.

Configura una vez, ejecuta cualquier cosa

- Elimina inconsistencias entre los entornos de desarrollo, testing y producción.
- Mejora significativa de fiabilidad en sistemas de integración continua.
- Debido a que los contenedores son tan ligeros existe una gran mejora de costos, rendimiento, implementación y portabilidad respecto a VMs.

docker

¿PREGUNTAS?





**PRACTICE
TIME**