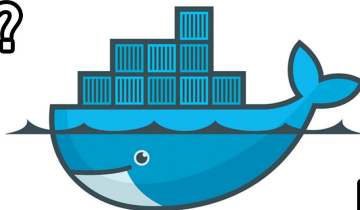


# Docker

Ingeniería de Servidores

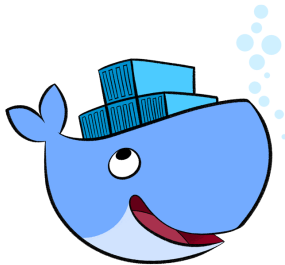
1 de mayo de 2016

# When?



# How?

Docker es un proyecto open-source con la finalidad de “permitir empaquetar una aplicación junto con todas sus dependencias en una unidad estandarizada para el desarrollo software”.



Docker comenzó como un proyecto interno de DotCloud.

2013:

- Es lanzado como código abierto.
- Colabora con RedHat para solucionar incompatibilidades.
- DotCloud Inc. se transforma en Docker Inc.



2014:

- Abandona LXC creando su propia librería, libcontainer, lo que supone el lanzamiento de su primera versión estable.
- Google, Red Hat, Parallels y los ingenieros de Canonical comienzan a colaborar con el proyecto.

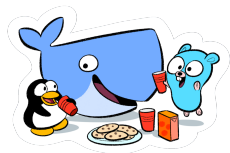
2015:

- “Revolución Docker”



## Contribuciones económicas:

- Red Hat (mayor contribuidor)
- Equipo de Docker
- IBM
- Google
- Cisco Systems
- Amadeus IT Group.



## Docker

Introducción

¿Qué es Docker?

Historia

Arquitectura

Funcionamiento Interno

Toolbox

Ventajas

Desventajas

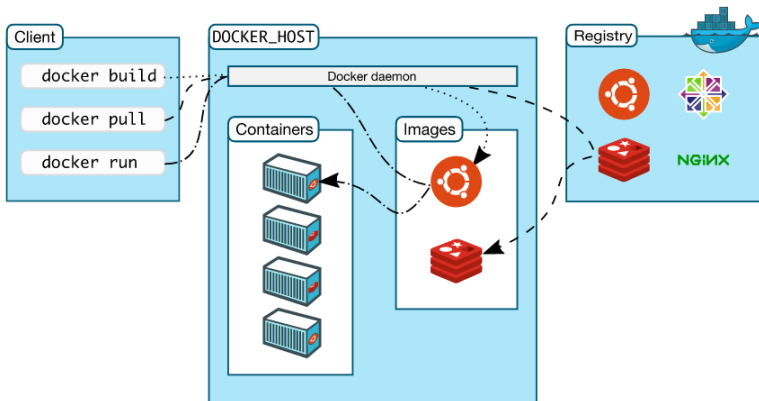
Ideas erróneas

Análisis prestaciones

Interpretación de resultados

Conclusiones

Bibliografía



- **Imágenes:** se utilizan para la *construcción*.
- **Registros:** se utilizan para la *distribución*. Los registros pueden ser públicos o privados y contienen imágenes.
- **Contenedores:** se utilizan para la *ejecución*. Son instancias de una imagen y almacenan todo lo que necesita para que una aplicación se ejecute.



Build



Ship



Run



Docker Toolbox es un paquete proporcionado por Docker que incluye todo lo necesario para instalar Docker en Mac o Windows, donde debe ejecutarse a través de una máquina virtual:



- **Docker Engine:** Es el núcleo de la plataforma Docker. Permite ejecutar los comandos para construir, distribuir y ejecutar contenedores.
- **Kitematic:** Es un proyecto de código abierto construido para simplificar el uso de Docker, proporcionando una interfaz gráfica de usuario muy intuitiva.



- **Docker Machine:** Es una herramienta que permite automatizar las tareas relacionadas con la configuración de esta máquina virtual. '
- **Docker Compose:** “Es una herramienta para la definición y ejecución de aplicaciones multi-contenedor Docker”





- Es un proyecto de código abierto.
- Facilita el desarrollo de aplicaciones (eliminan problemas de dependencias software).
- Los contenedores son ligeros, fáciles de crear y borrar.
- Consume pocos recursos hardware.
- En Docker Hub hay disponibles muchas imágenes que pueden descargarse y modificarse libremente.



- Al ser relativamente nuevo, es posible encontrar errores en algunas versiones.
- Se requiere, como mínimo, Kernel 3.8.
- Para Linux sólo soporta arquitecturas de 64 bits.
- En Windows aún se encuentra en fase de prueba.

- Docker no es una máquina virtual.
- Docker no es independiente del sistema operativo de la máquina anfitriona.
- Docker aumenta la seguridad al ejecutar las aplicaciones en contenedores, pero sigue habiendo vulnerabilidad.
- Docker no tiene overhead.



El hardware y el software utilizado para la evaluación:

- **Equipo 1.** *Procesador Intel Core i3. Frecuencia 1,8 GHz. Memoria 4 GB. Sistema operativo Ubuntu 14.04*
- **Equipo 2.** *Procesador Intel Core i5. Frecuencia 1,3 GHz. Memoria 8 GB. Sistema operativo OS X Yosemite*

Se han de tener en cuenta los siguientes puntos para el estudio:

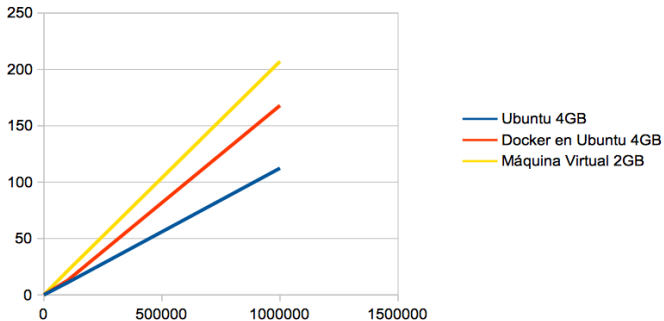
- Medida para la evaluación: tiempo de ejecución (en segundos).
- Máquinas virtuales:
  - Software de virtualización: VirtualBox.
  - La máquina del **equipo 1** tendrá 2 GB de memoria RAM.
  - La máquina del **equipo 2** tendrá 4 GB de memoria RAM.
  - Sistema operativo Ubuntu 14.04.
- En Docker se ejecuta el mismo script desde un contenedor con la imagen de Ubuntu 14.04.



Máximo	Equipo 1	Docker	Máquina virtual 1
100	0,019	0,0244	0,0169
1000	0,139	0,1482	0,2093
10000	1,1362	1,3036	2,1676
100000	10,8797	12,7948	21,2975
1000000	112,3853	167,9897	207,1008

Tabla 6.1: Comparativa en Ubuntu

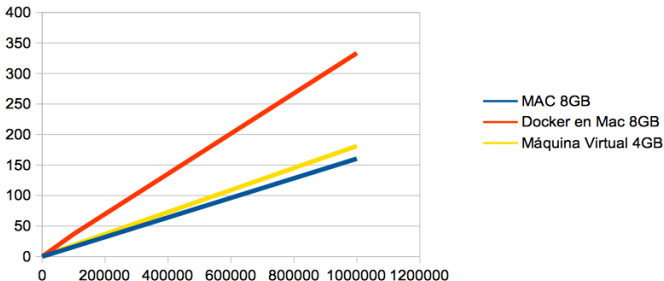
### Comparativa Docker nativo



Máximo	Equipo 2	Docker	Máquina virtual 2
100	0,0162	0,0372	0,0197
1000	0,1531	0,3634	0,1778
10000	1,5818	3,6926	1,9252
100000	15,9377	36,8415	18,7984
1000000	160,4102	333,4249	181,1432

Tabla 6.2: Comparativa en OS X

### Comparativa Docker no nativo



- Docker nativo:
  - 51.89 % más lento que el host.
  - 45.67 % más lento que la máquina virtual.
- Docker no nativo:
  - 33.101 % más lento que el host.
  - 23.289 % más rápido que la máquina virtual

En cuanto al tiempo de ejecución:

- Docker nativo: ventajoso frente a las máquinas virtuales.
- Docker no nativo: no supone ventajas.

La diferencia del tiempo de ejecución de Docker con respecto al host puede justificarse por su portabilidad.



<https://soundcloud.com/thenewstackmakers/solomon-hykes-creator-of-the-docker-open-so>  
Audio de entrevista a Solomon Hykes.



<https://www.docker.com/what-docker> Web  
oficial de Docker.



DOCKER CONTAINERS Cristopher Negus, 2015.



<https://docs.docker.com/engine/understanding-docker> Documentación oficial de  
Docker, arquitectura.



DOCKER: UP & RUNNING Sean P. Kane, Karl Matthias,  
2015.



<http://koldohernandez.com/docker-que-es-y-caracteristicas-principales>  
**Koldo Hernández.**



<http://vagrantanddocker.blogspot.com.es/2015/06/cuales-son-sus-ventajas-y-desventajas.html>  
**Blog, vagrant y docker.**



<http://www.javiergarzas.com/2015/07/que-es-docker-sencillo.html> **Blog Javier Garzas, introducción a Docker**



[http://coast.pink/docker-software\\_1283959.html](http://coast.pink/docker-software_1283959.html)  
**Enciclopedia online COAST.PINK**



<https://blog.docker.com/2013/10/dotcloud-is-becoming-docker-inc/> **Blog de Docker.**



[https://blog.docker.com/2014/03/docker-0-9-introducing-execution-drivers-an](https://blog.docker.com/2014/03/docker-0-9-introducing-execution-drivers-an/) **Blog de Docker.**



[http://www.zdnet.com/article/docker-libcontainer-unifies-linux-container](http://www.zdnet.com/article/docker-libcontainer-unifies-linux-container/) **Artículo en zdnet.**



[http://www.infoworld.com/article/2925484/application-virtualization/look-whos-helping-build-docker-besides-dock.html?utm\\_content=bufferee5d7&utm\\_medium=](http://www.infoworld.com/article/2925484/application-virtualization/look-whos-helping-build-docker-besides-dock.html?utm_content=bufferee5d7&utm_medium=)

social&utm\_source=linkedin.com&utm\_campaign=buffer **Revista digital InfoWorld**



<https://www.docker.com/company> **Web oficial de Docker, acerca de Docker.**



<https://www.docker.com/products/docker-toolbox>  
**Documentación oficial de Docker, Toolbox**



<https://www.docker.com/products/docker-engine>  
**Documentación oficial de Docker, Docker Engine.**



<https://docs.docker.com/compose/overview/>  
**Documentación oficial de Docker, Docker Compose**



<https://docs.docker.com/machine/>  
**Documentación oficial de Docker, Docker Machine**





<https://docs.docker.com/kitematic/userguide/>  
**Documentación oficial de Docker, Kitematic.**



<http://www.gonzalonazareno.org/cloud/material/IntroVirtualizacion.pdf> **Blog IES Gonzalo Zareno**



<https://docs.docker.com/engine/userguide/>  
**Documentación oficial de Docker, guía de usuario.**



<https://docs.docker.com/engine/reference/commandline/ps/> **Documentación oficial de Docker, comando ps.**



<https://docs.docker.com/engine/reference/builder/> **Documentación oficial de Docker, comando build.**



<http://www.javiergarzas.com/2015/11/para-los-que-empiezan-crear-y-ejecutar-una-html> Blog Javier Garzas, tutorial Docker