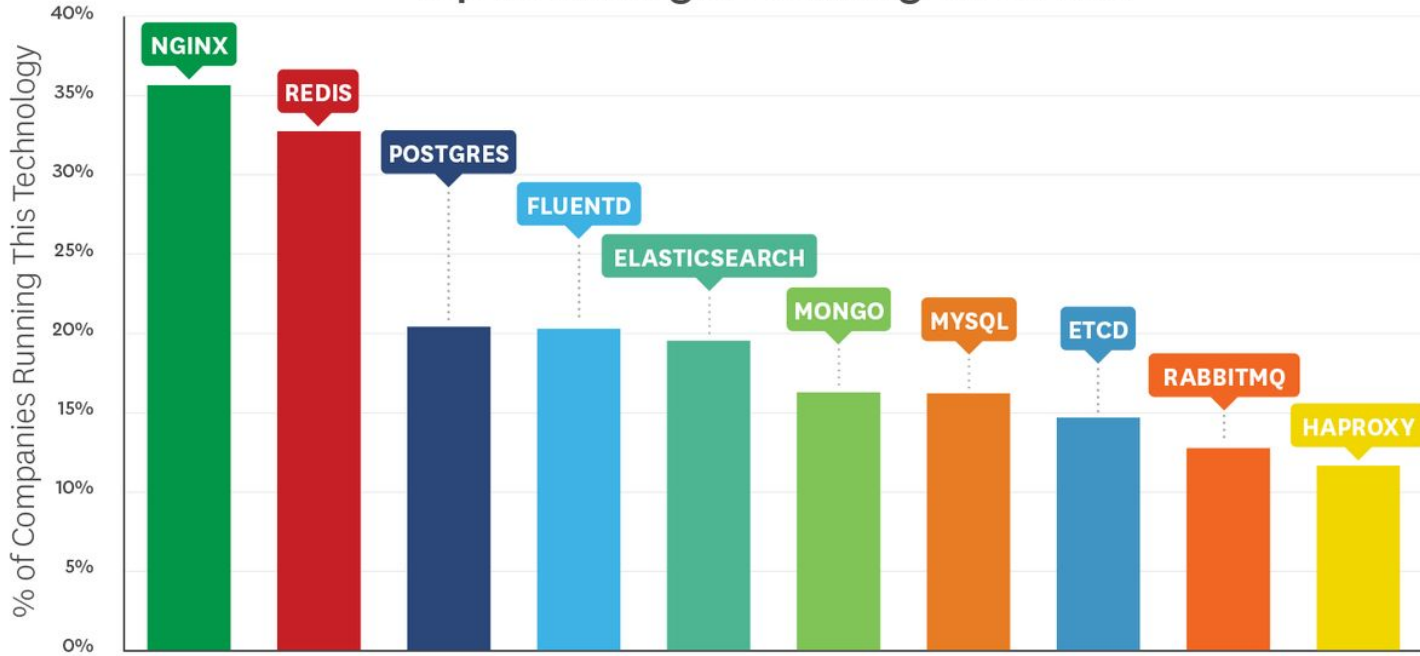




Imagen: DotCloud, Inc.

Gustavo Magallanes Guijón, febrero del 2024.

Top Technologies Running on Docker



Source: Datadog

FreeBSD Jails expand on Unix chroot to isolate files



Jails

2000

Linux-VServer ports kernel isolation, but requires recompilation

VServer



2001

Solaris Zones bring the concept of snapshots



Zones

2004



cgroups

2006

Google introduces Process Containers, merged as cgroups



Namespaces

2008

Red Hat adds user namespaces, limiting root access in containers

2008



LXC

IBM creates LXC, providing user tools for cgroups and namespaces

2013

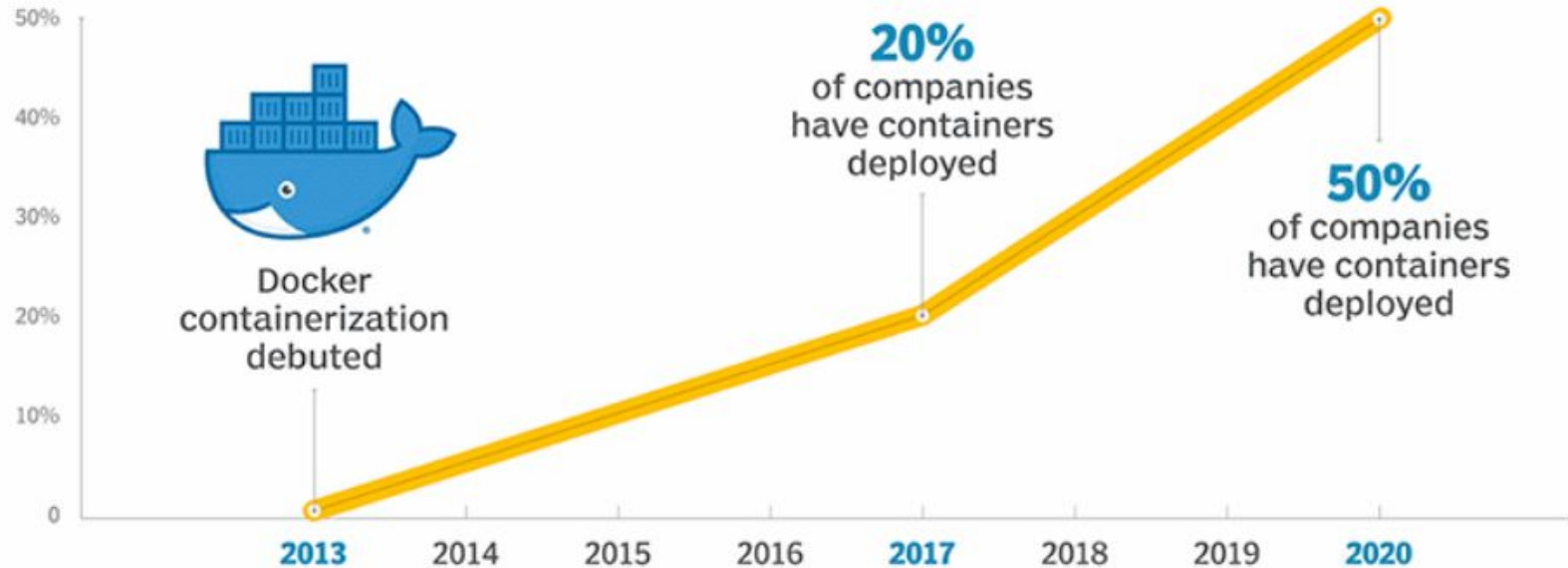


Docker

Docker provides simple user tools and images. Containers go mainstream

<https://borosan.gitbook.io/docker-handbook/containerization-history>

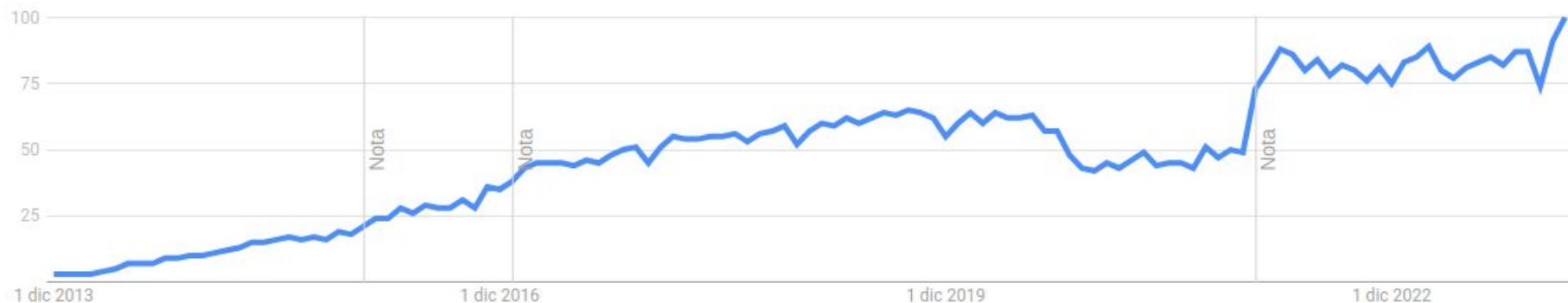
Containerization timeline



Source: Gartner

Tendencias Docker en Google Trends

Interés a lo largo del tiempo ?



¿Qué es Docker?



- Es una herramienta de código abierto que simplifica el proceso de despliegue de aplicaciones al utilizar contenedores de software.
- Los contenedores son entornos ligeros y portátiles que contienen todo lo necesario para que una aplicación se ejecute, incluidas las bibliotecas y las dependencias.
- Automatiza el empaquetado de aplicaciones en contenedores, lo que facilita su distribución y ejecución en diferentes entornos de manera consistente y sin problemas.

¿Qué es Docker?

- Esto significa que se pueden crear, probar y distribuir aplicaciones de manera más rápida y eficiente, sin preocuparse por las diferencias entre los sistemas operativos y las configuraciones de hardware.
- En resumen, Docker simplifica el desarrollo y despliegue de aplicaciones al proporcionar un entorno consistente y confiable para ejecutarlas.



Ventajas

- Docker proporciona un entorno de desarrollo consistente y aislado. Esto garantiza que todos los desarrolladores estén trabajando en el mismo contexto, lo que facilita la colaboración y reduce los errores causados por discrepancias en los entornos de desarrollo.
- Al estandarizar el entorno de desarrollo a través de contenedores, Docker asegura que todos los miembros del equipo estén utilizando las mismas versiones de software, bibliotecas y dependencias. Simplifica la configuración inicial para nuevos desarrolladores que se unen al equipo.



Ventajas

3. Reduce los conflictos entre aplicaciones y mejora la seguridad al limitar el impacto de posibles vulnerabilidades.
4. Docker facilita el empaquetado de aplicaciones y su despliegue en diferentes entornos, lo que permite implementar rápidamente cambios en producción.
5. Al encapsular una aplicación y todas sus dependencias en un contenedor, los despliegues se vuelven más consistentes y menos propensos a errores, lo que acelera el ciclo de desarrollo y mejora la capacidad de respuesta.



THE DOCKER CONTAINER

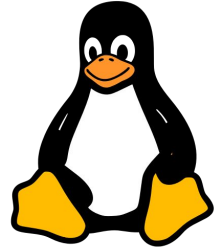


MY HOST MACHINE

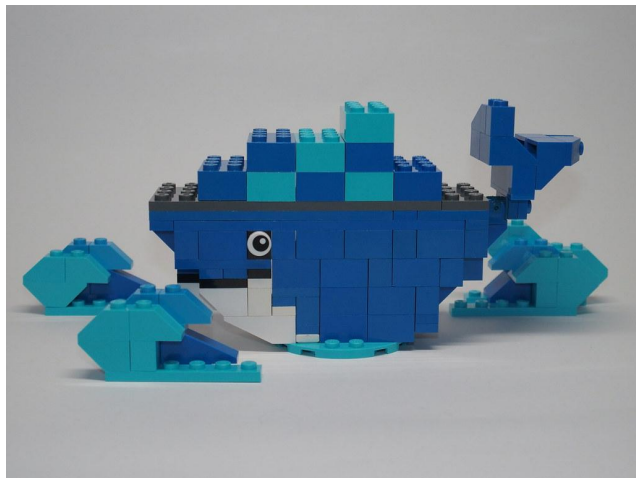
imgflip.com

Portabilidad y la compatibilidad que ofrece Docker

- Docker es compatible con sistemas operativos Linux, lo que significa que puede ejecutarse en cualquier sistema que ejecute Linux, ya sea un servidor físico, una máquina virtual, un entorno de nube o incluso en un sistema personal.
- Distribuciones de Linux: Ubuntu, CentOS, Debian, entre otras. Esto significa que las aplicaciones empaquetadas en contenedores Docker pueden ejecutarse de manera consistente en diferentes distribuciones de Linux sin necesidad de modificaciones adicionales.



Portabilidad y la compatibilidad que ofrece Docker

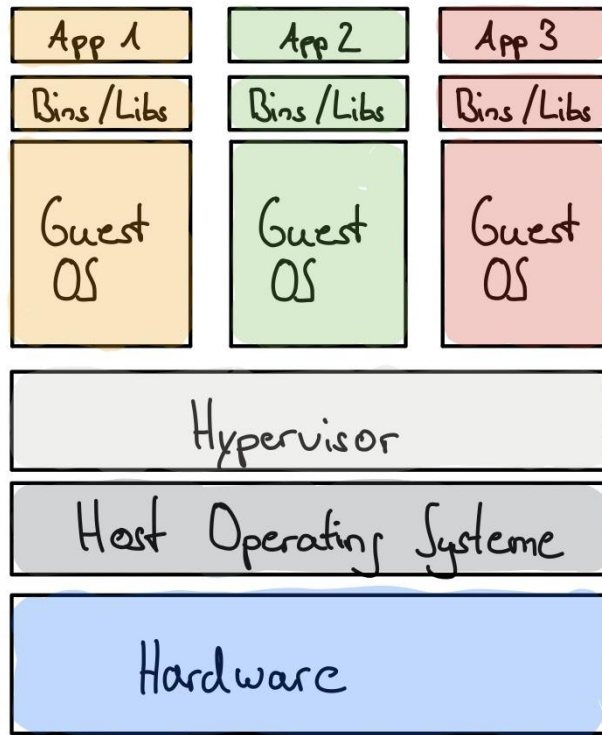


- Para asegurar la compatibilidad y la portabilidad, es importante que la arquitectura del contenedor Docker coincida con la arquitectura del sistema anfitrión. Por ejemplo, un contenedor Docker diseñado para ejecutarse en una arquitectura x86 (Intel) no funcionará en un sistema con arquitectura ARM (Acorn RISC Machines).
- Esto significa que si una aplicación se ejecuta correctamente en el sistema anfitrión, también se ejecutará de manera similar en un contenedor Docker en ese mismo sistema.

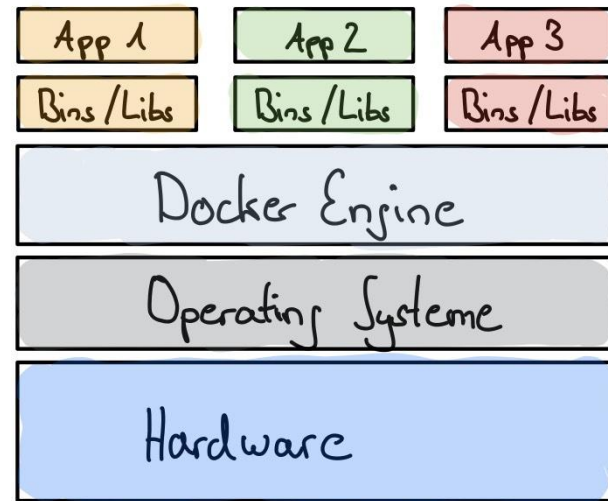
Resumen:

- Ofrece entornos de desarrollo limpios, seguros y portátiles.
- Facilita despliegues reproducibles sin perder dependencias.
- Garantiza el aislamiento de aplicaciones para evitar interferencias.
- Simplifica pruebas, integración y empaquetado.
- Reduce problemas de compatibilidad entre diferentes entornos.
- Agiliza despliegues rápidos y económicos de aplicaciones.
- Proporciona las ventajas de una máquina virtual sin la sobrecarga asociada de una MV completa.





Virtual Machines



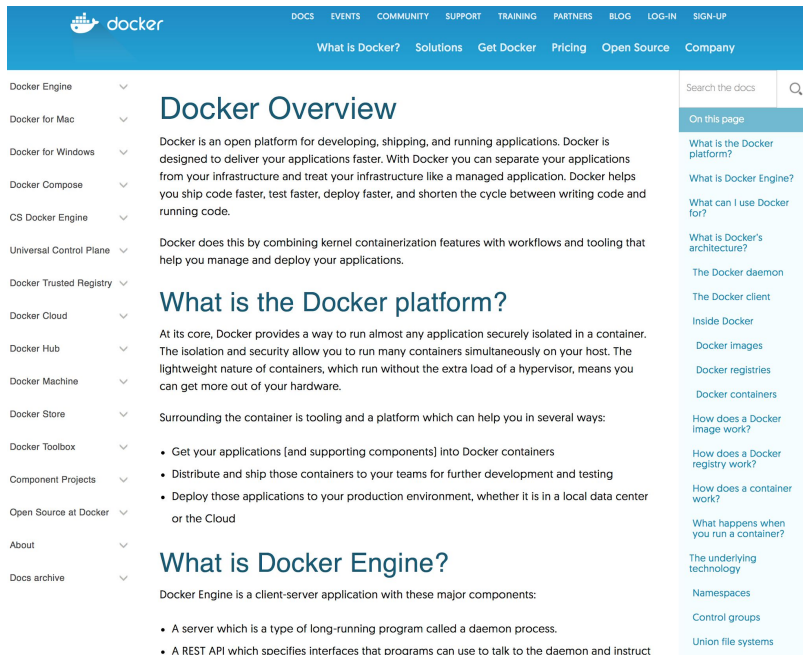
Containers

Palabras clave

Docker

Plataforma de software en su conjunto, que incluye herramientas y componentes asociados.

Es un ecosistema de software.



The screenshot shows the Docker website's 'Docker Overview' page. The navigation bar at the top includes links for DOCS, EVENTS, COMMUNITY, SUPPORT, TRAINING, PARTNERS, BLOG, LOG-IN, and SIGN-UP. Below the navigation bar, there are links for 'What is Docker?', 'Solutions', 'Get Docker', 'Pricing', 'Open Source', and 'Company'. The main content area is titled 'Docker Overview' and contains a search bar, a table of contents, and a detailed description of Docker. The table of contents lists various Docker components and their functions, including Docker Engine, Docker for Mac, Docker for Windows, Docker Compose, CS Docker Engine, Universal Control Plane, Docker Trusted Registry, Docker Cloud, Docker Hub, Docker Machine, Docker Store, Docker Toolbox, Component Projects, Open Source at Docker, About, and Docs archive. The main text describes Docker as an open platform for developing, shipping, and running applications, and provides a detailed overview of the Docker platform and engine.

Docker Overview

Docker is an open platform for developing, shipping, and running applications. Docker is designed to deliver your applications faster. With Docker you can separate your applications from your infrastructure and treat your infrastructure like a managed application. Docker helps you ship code faster, test faster, deploy faster, and shorten the cycle between writing code and running code.

Docker does this by combining kernel containerization features with workflows and tooling that help you manage and deploy your applications.

What is the Docker platform?

At its core, Docker provides a way to run almost any application securely isolated in a container. The isolation and security allow you to run many containers simultaneously on your host. The lightweight nature of containers, which run without the extra load of a hypervisor, means you can get more out of your hardware.

Surrounding the container is tooling and a platform which can help you in several ways:

- Get your applications [and supporting components] into Docker containers
- Distribute and ship those containers to your teams for further development and testing
- Deploy those applications to your production environment, whether it is in a local data center or the Cloud

What is Docker Engine?

Docker Engine is a client-server application with these major components:

- A server which is a type of long-running program called a daemon process.
- A REST API which specifies interfaces that programs can use to talk to the daemon and instruct

Search the docs

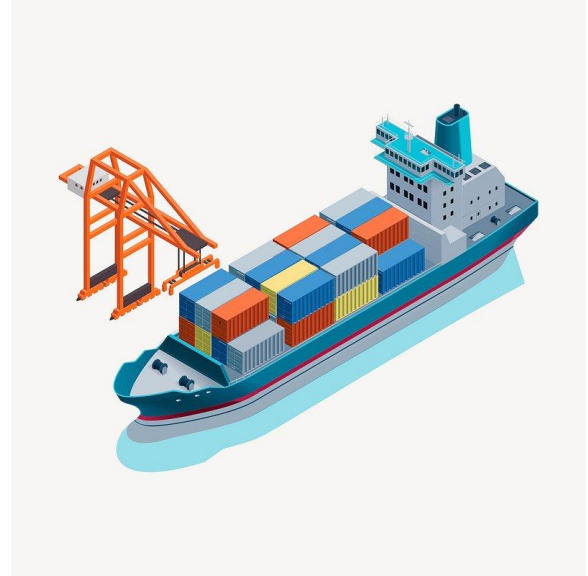
On this page

- What is the Docker platform?
- What is Docker Engine?
- What can I use Docker for?
- What is Docker's architecture?
- The Docker daemon
- The Docker client
- Inside Docker
 - Docker images
 - Docker registries
 - Docker containers
- How does a Docker image work?
- How does a Docker registry work?
- How does a container work?
- What happens when you run a container?
- The underlying technology
 - Namespaces
 - Control groups
 - Union file systems

Palabras clave

Docker Engine

Es el corazón de la plataforma Docker. Proporciona un entorno para crear, ejecutar y administrar contenedores Docker.



Palabras clave

Dockerfile

Es un archivo de texto plano que contiene instrucciones detalladas para construir una **imagen** de Docker, incluyendo qué sistema operativo base utilizar, qué paquetes instalar, qué archivos incluir y qué comandos ejecutar durante la construcción de la imagen.

```
Dockerfile U X
Dockerfile > ...
1 FROM node:14-alpine3.16
2
3 WORKDIR /app
4
5 COPY . .
6
7 RUN npm install
8
9 CMD [ "npm", "start" ]
```

Palabras clave

Imagen

Es un paquete autónomo y ligero que contiene todo lo necesario para ejecutar una aplicación, incluidos el código, las bibliotecas, las dependencias, las variables de entorno y las configuraciones del entorno.

Se crean a partir de un Dockerfile, que contiene instrucciones detalladas sobre cómo construir la imagen.



Palabras clave

Contenedor

Un contenedor Docker es una instancia en tiempo de ejecución de una **imagen** Docker.

Proporcionan un entorno aislado y ligero para ejecutar aplicaciones de manera consistente y portátil.

Se basa en una imagen Docker, pero tiene su propio espacio de archivos, procesos y recursos.





DOCKER FILE

BUILD



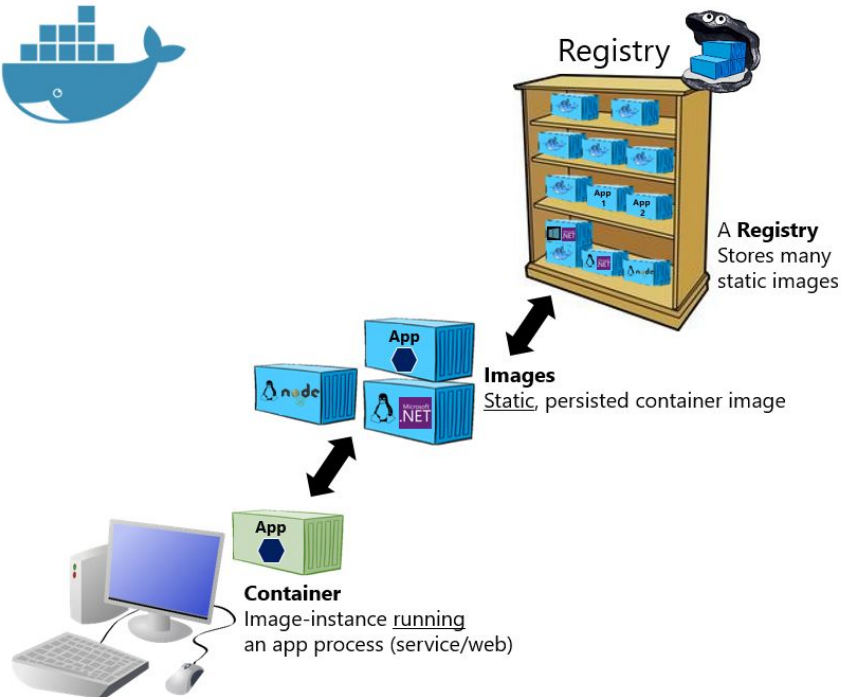
DOCKER IMAGE

RUN



DOCKER CONTAINER

Basic taxonomy in Docker



Hosted Docker Registry

Docker Trusted Registry on-prem.

On-premises

(‘n’ private organizations)

Docker Hub Registry

Docker Trusted Registry on-cloud

Azure Container Registry

AWS Container Registry

Public Cloud

(specific vendors)

Google Container Registry

Quay Registry

Other Cloud

Funcionamiento básico

