

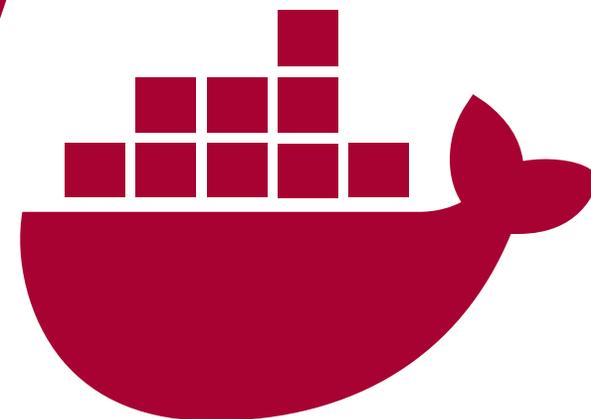
Grado en Ingeniería Informática - Ingeniería del Software

Evolución y Gestión de la Configuración



Escuela Técnica Superior de
Ingeniería Informática

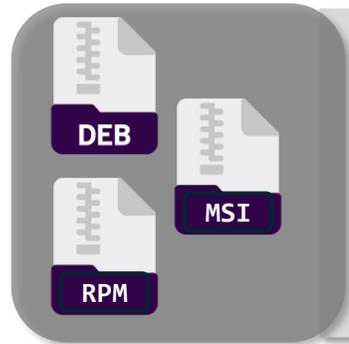
Práctica 5 Contenedores y aislamiento



1. **Introducción a Docker y DockerHub**
 2. **Imágenes**
 3. **Contenedores**
 4. **Volúmenes y redes**
 5. **Composición de servicios: Docker Compose**
 6. **Y yo, ¿qué puedo hacer en mi proyecto?**
 7. **Tutorial: *docker***
 8. **Tutorial: *dockerizando una aplicación sencilla***
 9. **Ejercicio práctico: *docker* y *uvlhub***
- 

- 1. Introducción a Docker y DockerHub**
 - 2. Imágenes**
 - 3. Contenedores**
 - 4. Volúmenes y redes**
 - 5. Composición de servicios: Docker Compose**
 - 6. Y yo, ¿qué puedo hacer en mi proyecto?**
 - 7. Tutorial: *docker***
 - 8. Tutorial: *dockerizando una aplicación sencilla***
 - 9. Ejercicio práctico: *docker* y *uvlhub***
- 

1. Introducción a Docker



Distintas instalaciones de módulos y paquetes de python de forma simultánea



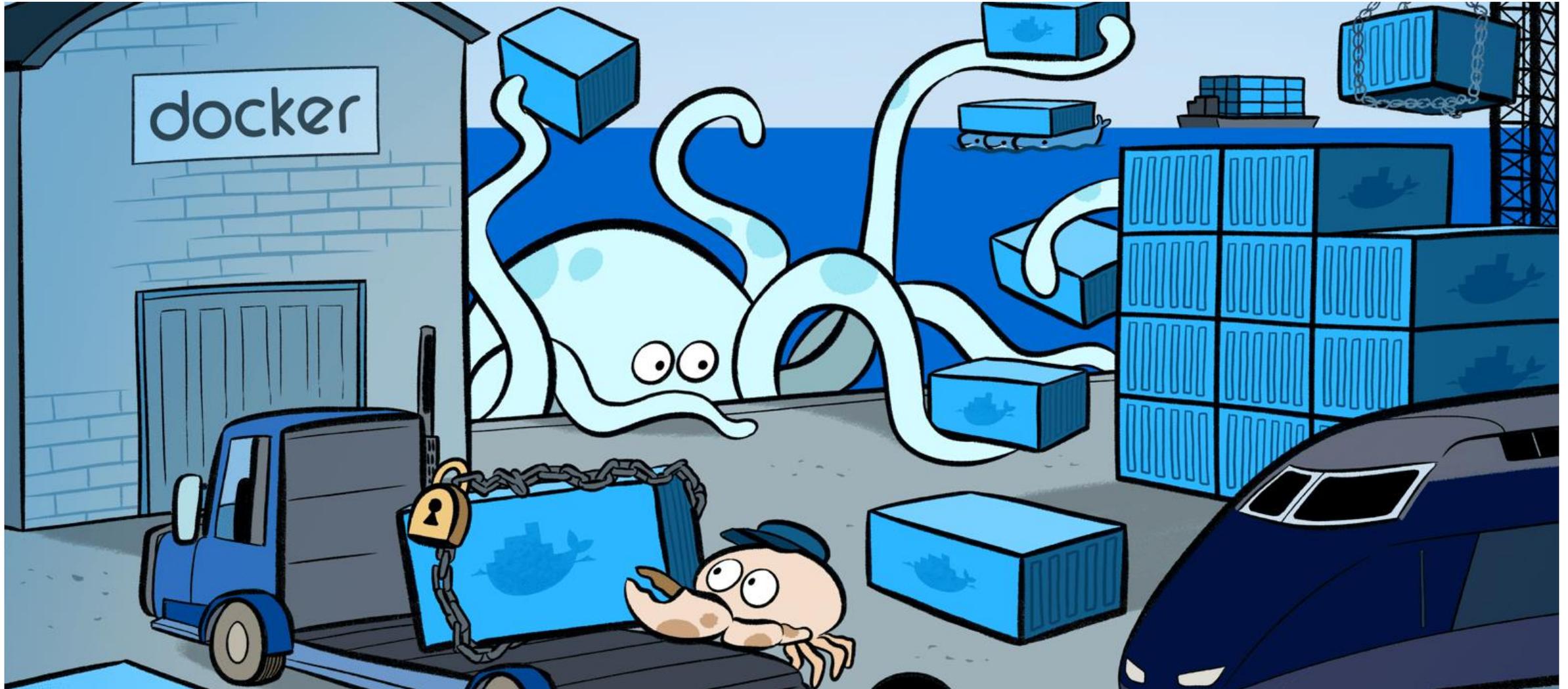
Aislar dependencias más allá de python



Aislar todas las dependencias del sistema

Overhead y aislamiento

1. Introducción a Docker

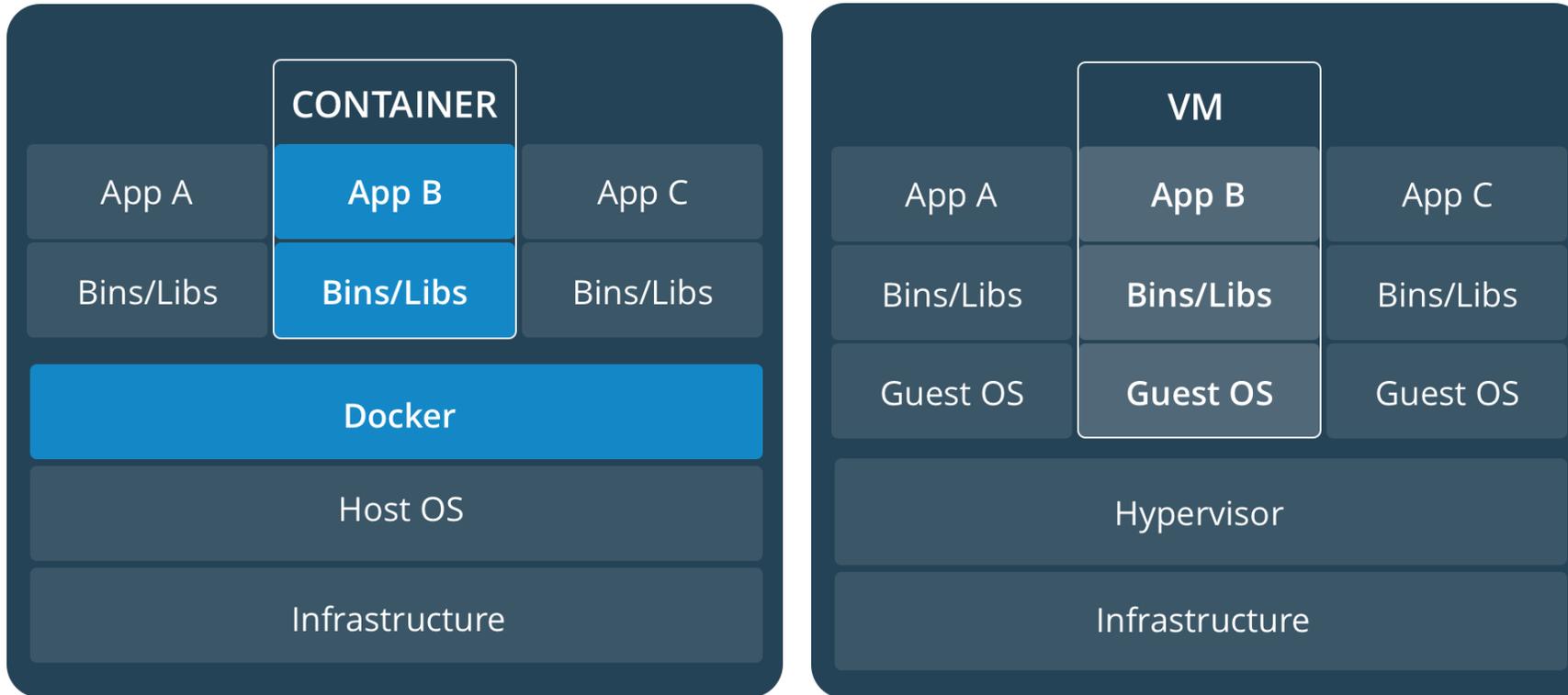


1. Introducción a Docker



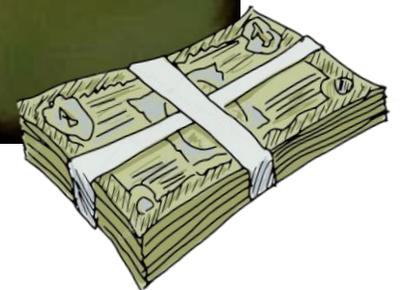
1. Introducción a Docker

¿Por qué usar contenedores si ya tenemos máquinas virtuales?

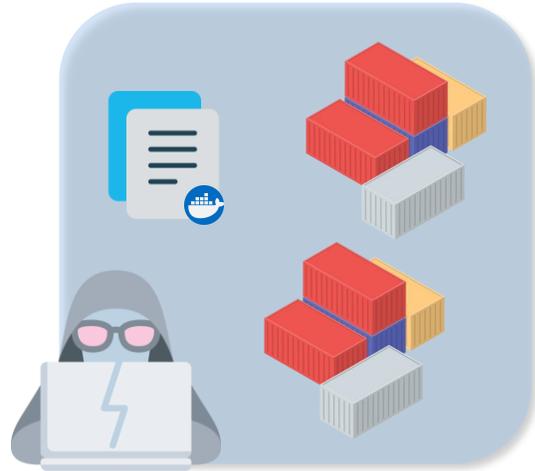


1. Introducción a Docker

¿Por qué usar contenedores si ya tenemos máquinas virtuales?



1. Introducción a DockerHub



PRIVATE

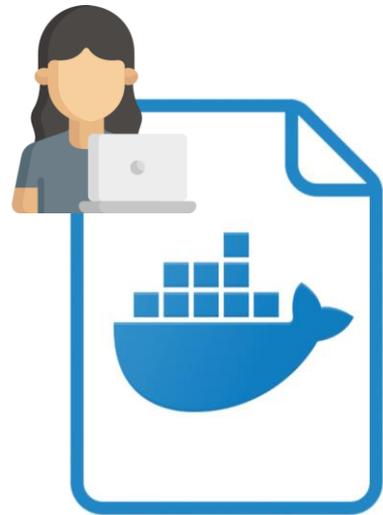


<https://hub.docker.com/>

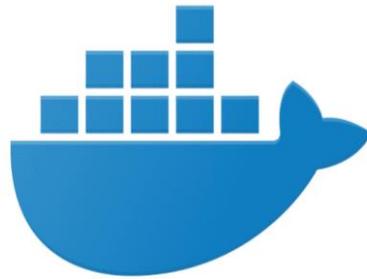
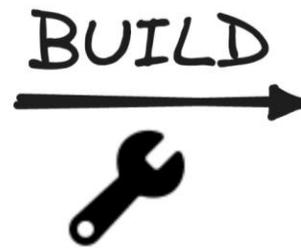


1. Introducción a Docker y DockerHub
 - 2. Imágenes**
 3. Contenedores
 4. Volúmenes y redes
 5. Composición de servicios: Docker Compose
 6. Y yo, ¿qué puedo hacer en mi proyecto?
 7. Tutorial: *docker*
 8. Tutorial: *dockerizando una aplicación sencilla*
 9. Ejercicio práctico: *docker* y *uvlhub*
- 

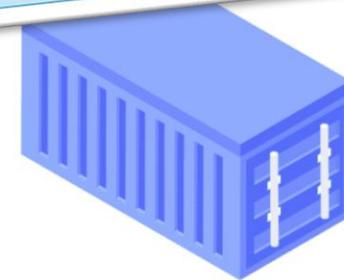
2. Imágenes



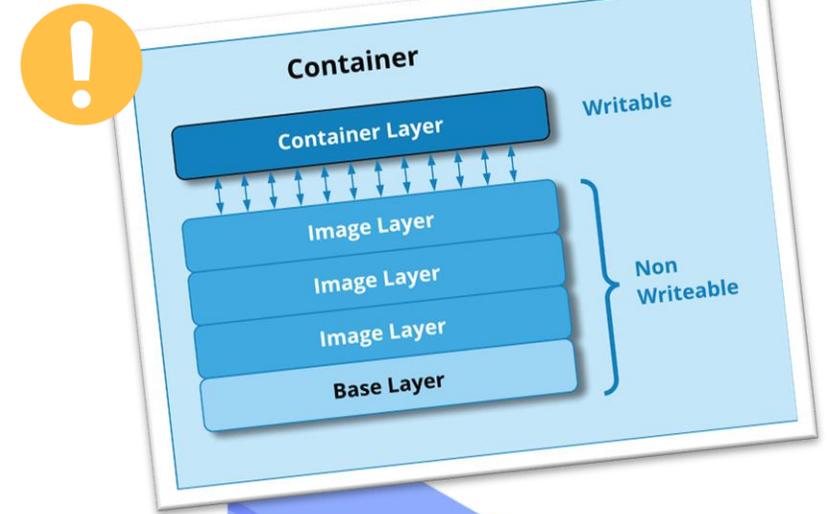
Dockerfile



Docker image



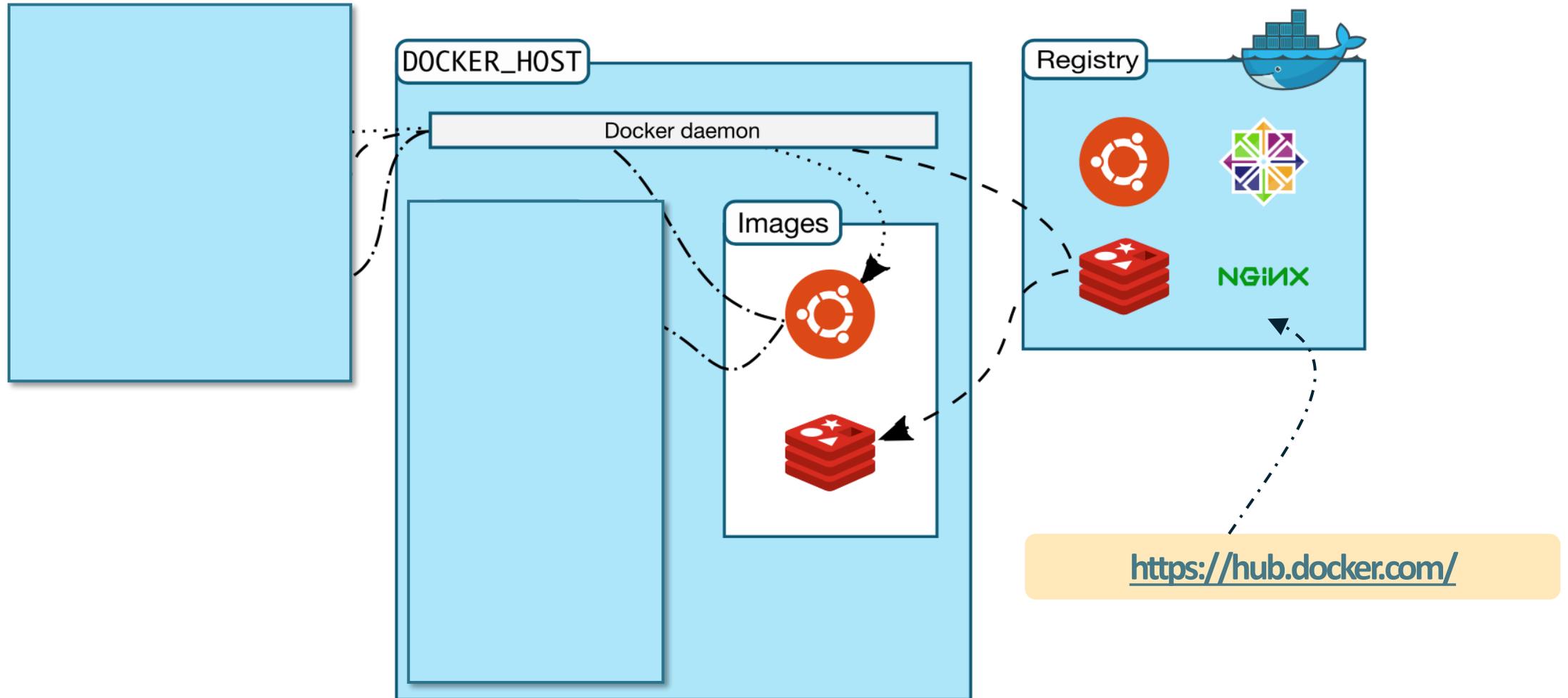
Docker container



Plantillas de solo lectura que contienen el código y las dependencias necesarias para ejecutar aplicaciones

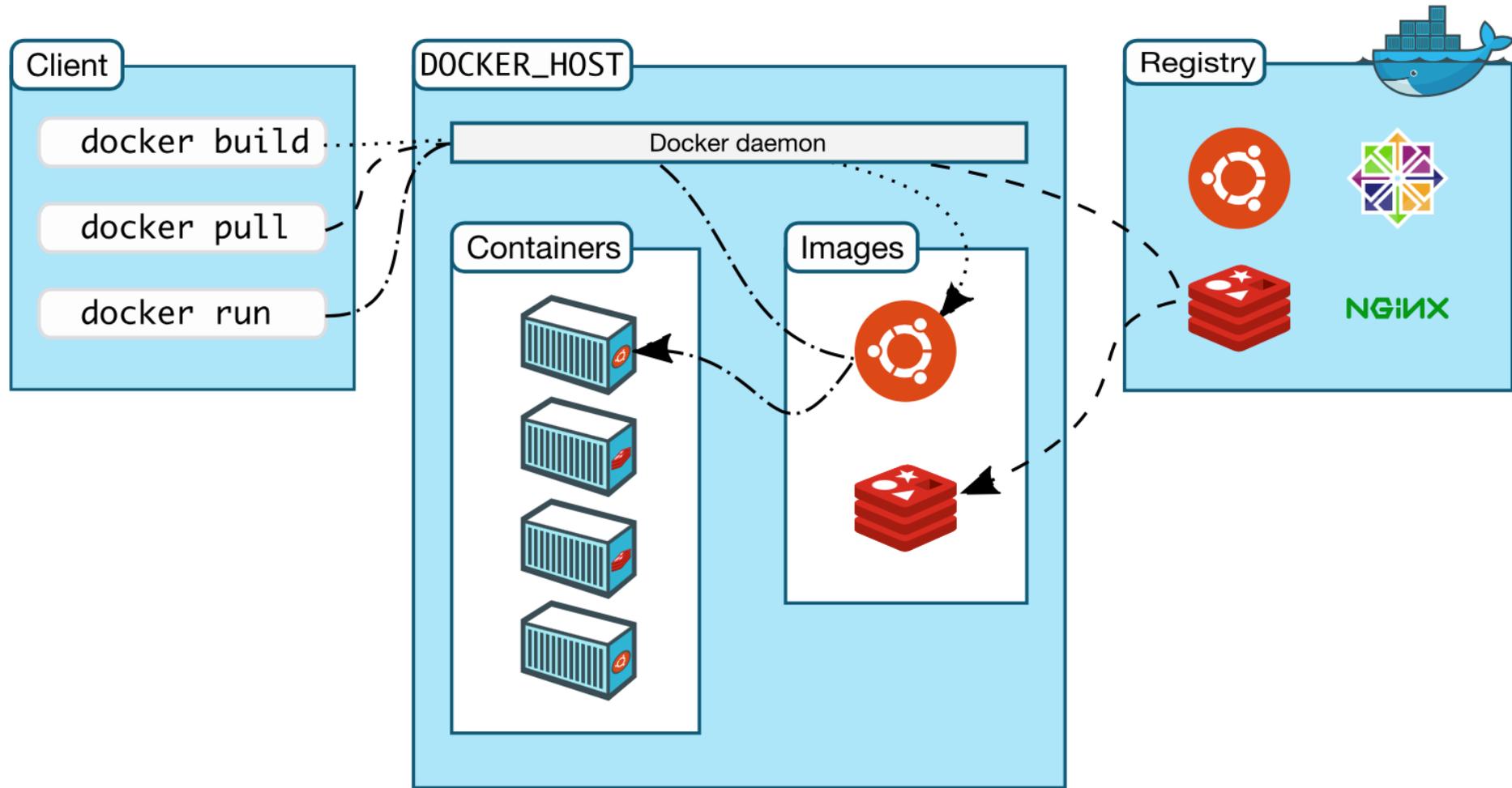
2. Imágenes

Y si no escribo un Dockerfile, ¿de dónde salen las imágenes?



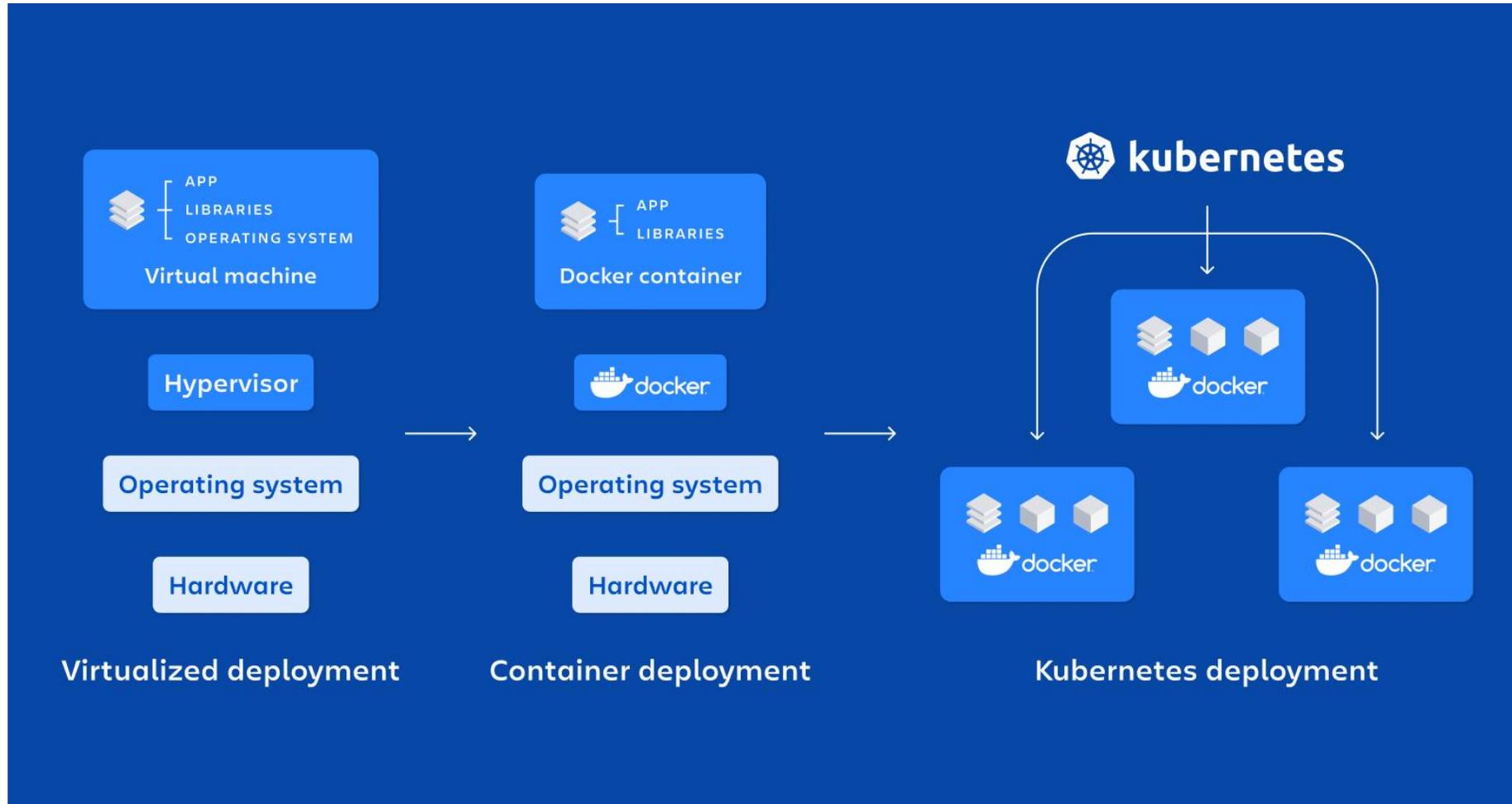
1. Introducción a Docker y DockerHub
 2. Imágenes
 - 3. Contenedores**
 4. Volúmenes y redes
 5. Composición de servicios: Docker Compose
 6. Y yo, ¿qué puedo hacer en mi proyecto?
 7. Tutorial: *docker*
 8. Tutorial: *dockerizando una aplicación sencilla*
 9. Ejercicio práctico: *docker* y *uvlhub*
- 

3. Contenedores



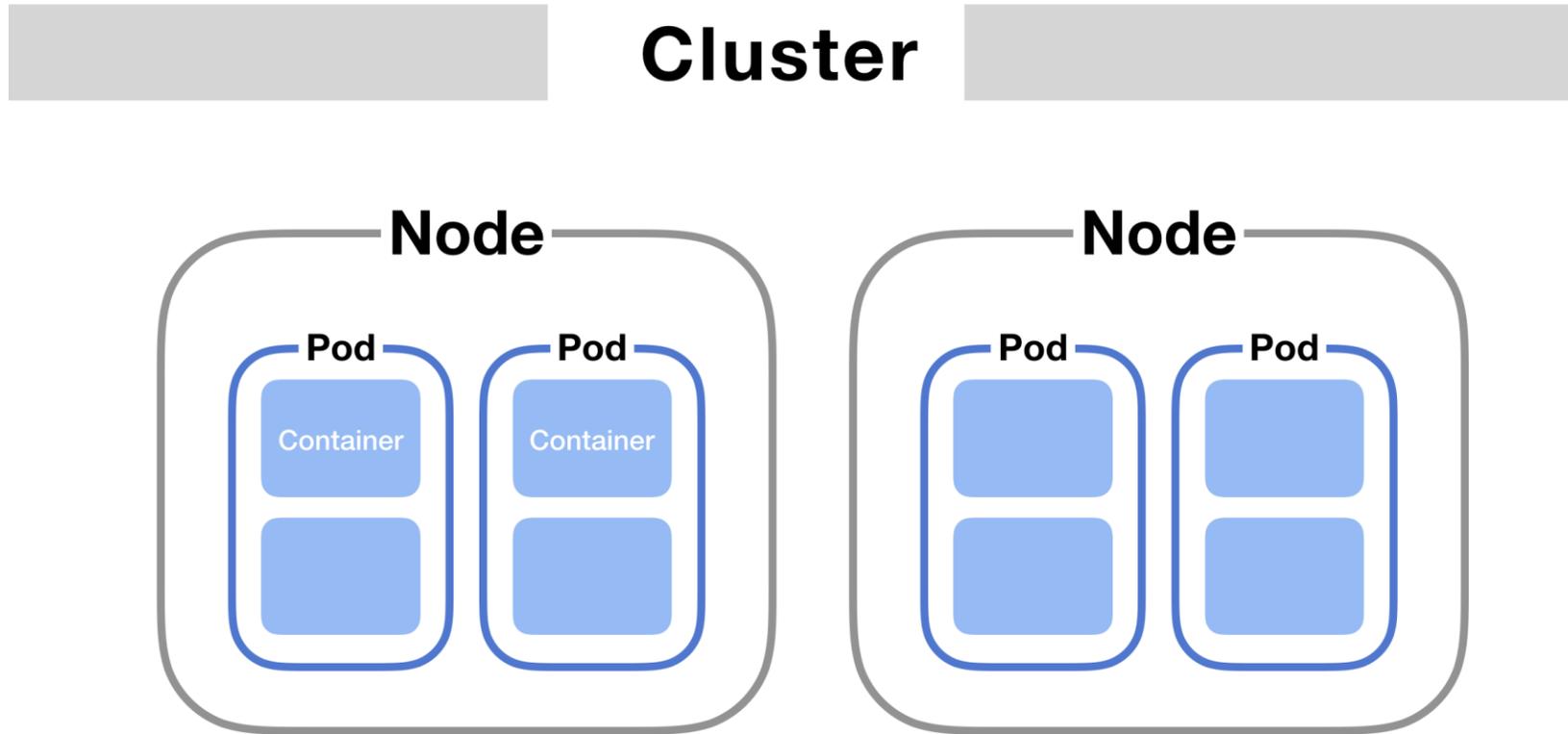
3. Contenedores

Contenedores en la nube



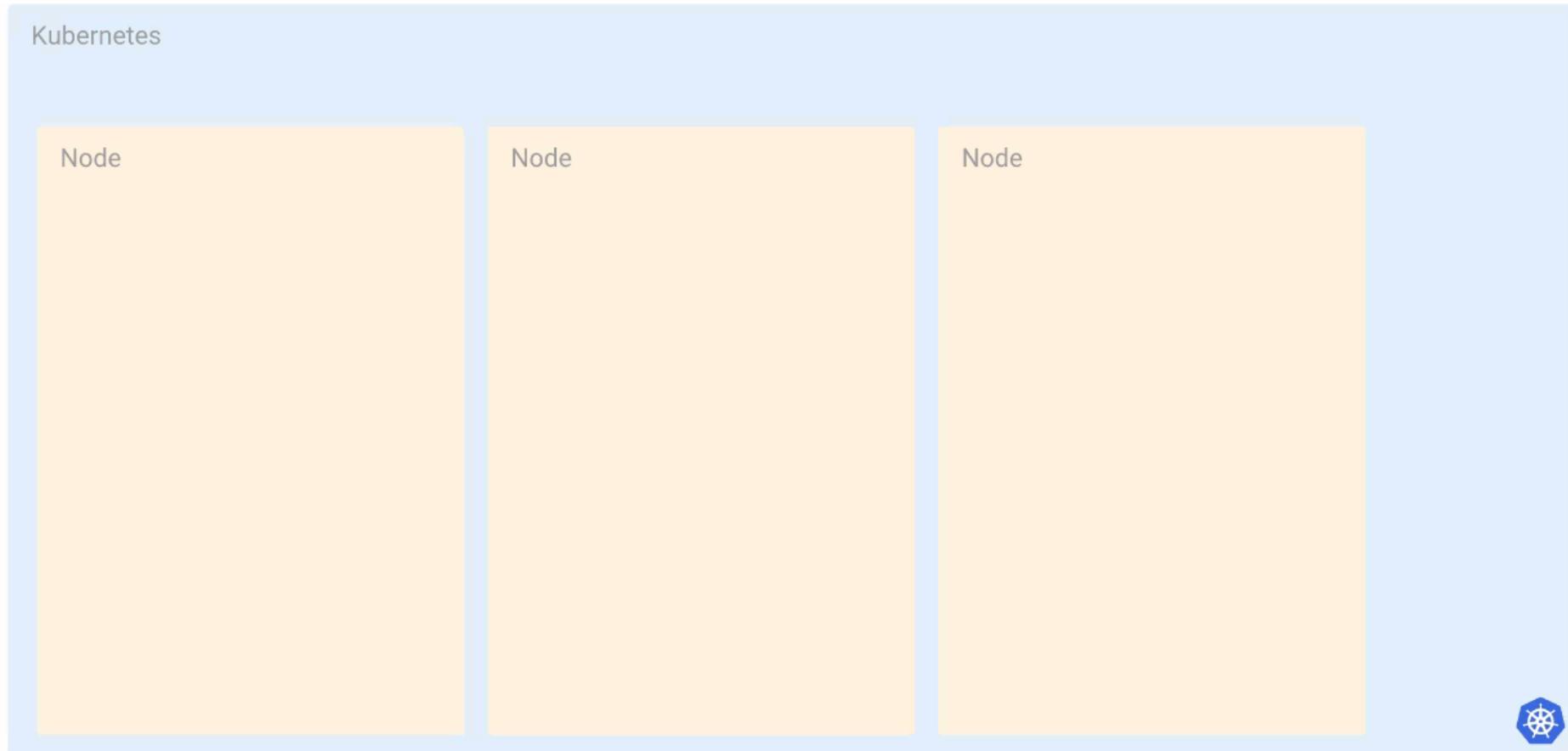
3. Contenedores

Contenedores en la nube



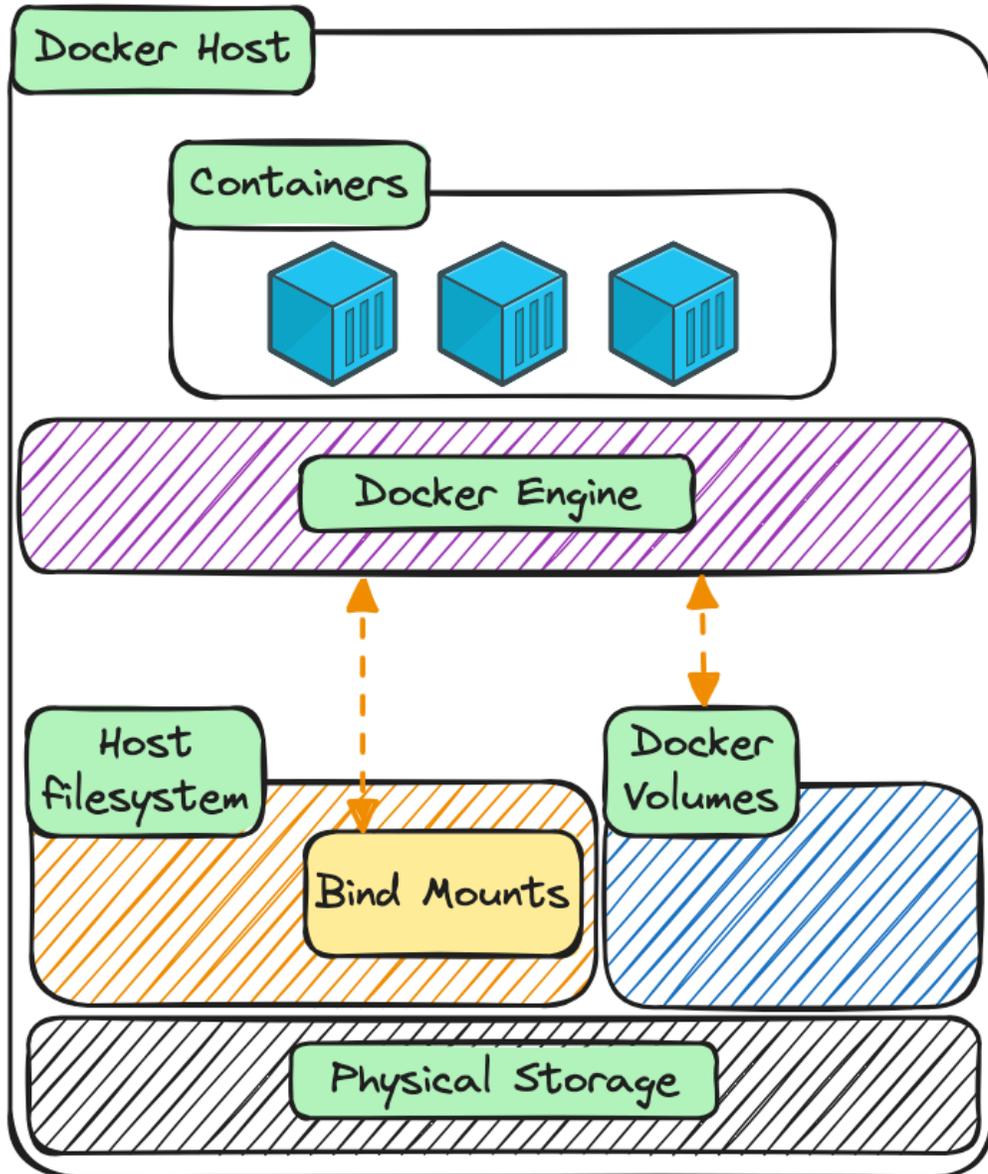
3. Contenedores

Contenedores en la nube



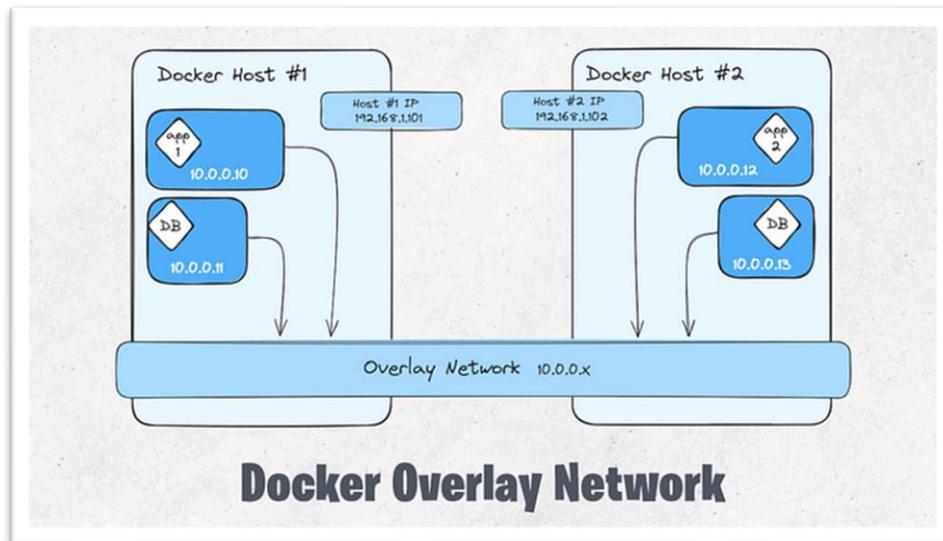
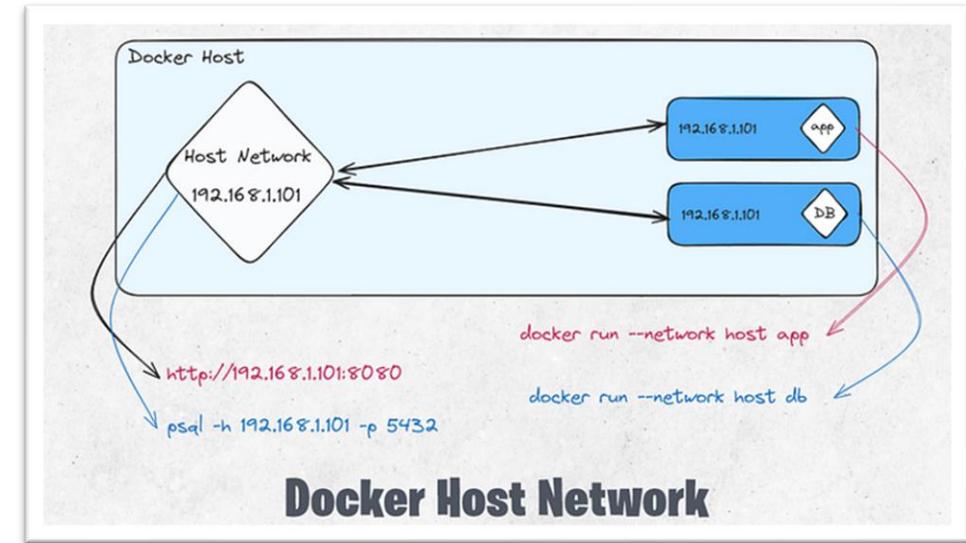
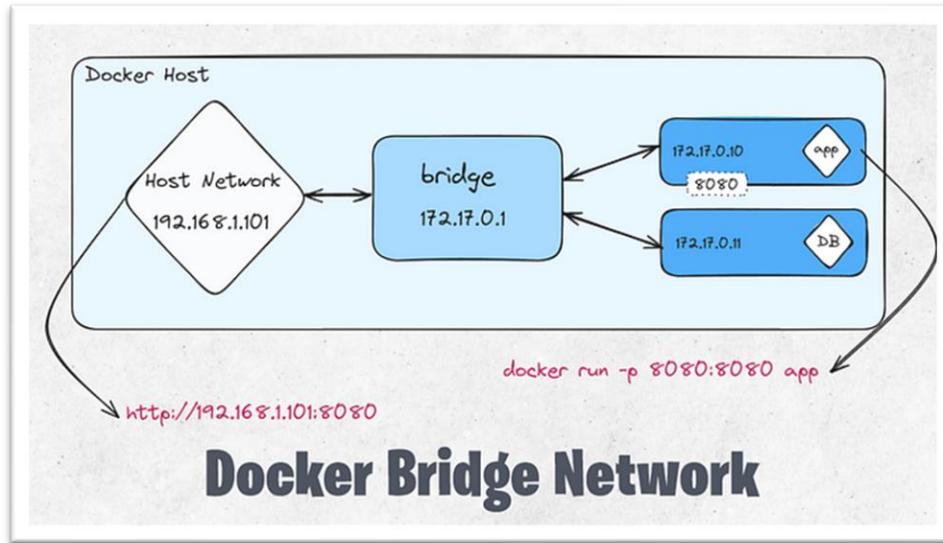
1. Introducción a Docker y DockerHub
 2. Imágenes
 3. Contenedores
 4. **Volúmenes y redes**
 5. Composición de servicios: Docker Compose
 6. Y yo, ¿qué puedo hacer en mi proyecto?
 7. Tutorial: *docker*
 8. Tutorial: *dockerizando una aplicación sencilla*
 9. Ejercicio práctico: *docker* y *uvlhub*
- 

4. Volúmenes y redes



Los volúmenes permiten que los datos **se conserven incluso si el contenedor se elimina o reinicia**, manteniendo los archivos de manera persistente en el host.

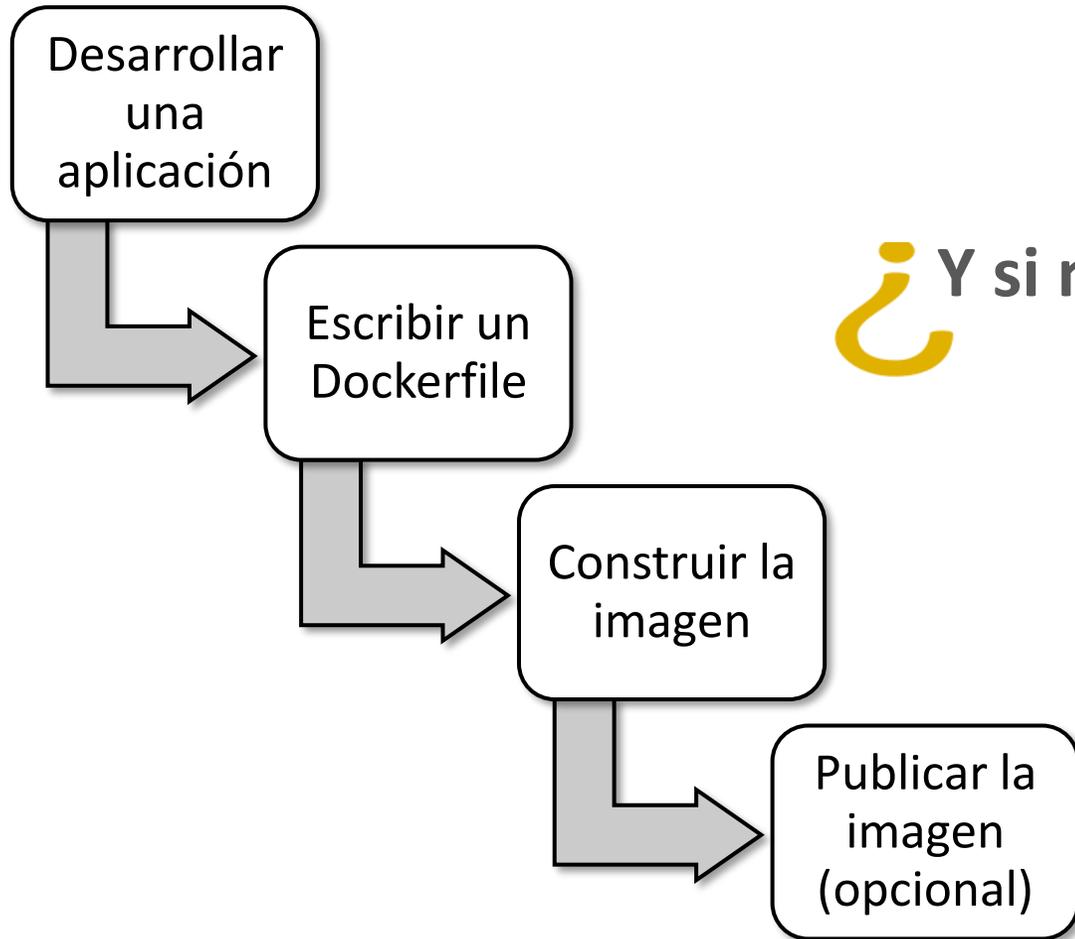
4. Volúmenes y redes



Las redes permiten que los contenedores **se comuniquen entre sí, con el host y con redes externas**. Docker proporciona **varios tipos de redes** (bridge, host, overlay, etc.) que permiten controlar el aislamiento, la seguridad y la visibilidad entre contenedores, facilitando la configuración de infraestructuras distribuidas o microservicios.

1. Introducción a Docker y DockerHub
 2. Imágenes
 3. Contenedores
 4. Volúmenes y redes
 5. **Composición de servicios: Docker Compose**
 6. Y yo, ¿qué puedo hacer en mi proyecto?
 7. Tutorial: *docker*
 8. Tutorial: *dockerizando una aplicación sencilla*
 9. Ejercicio práctico: *docker* y *uvlhub*
- 

5. Composición de servicios: Docker compose

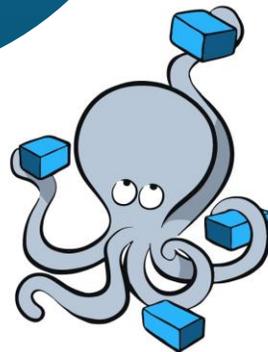
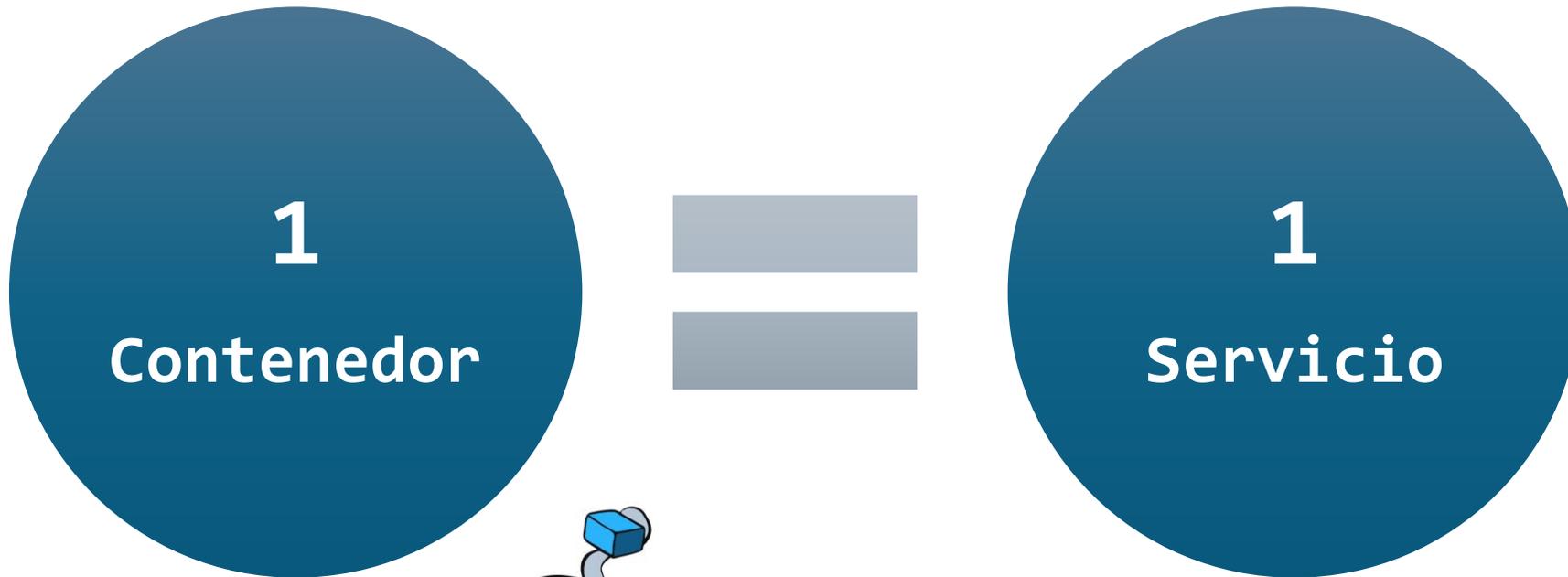


¿Y si necesito varias imágenes y varios contenedores ?



5. Composición de servicios: Docker compose

Principio de responsabilidad única

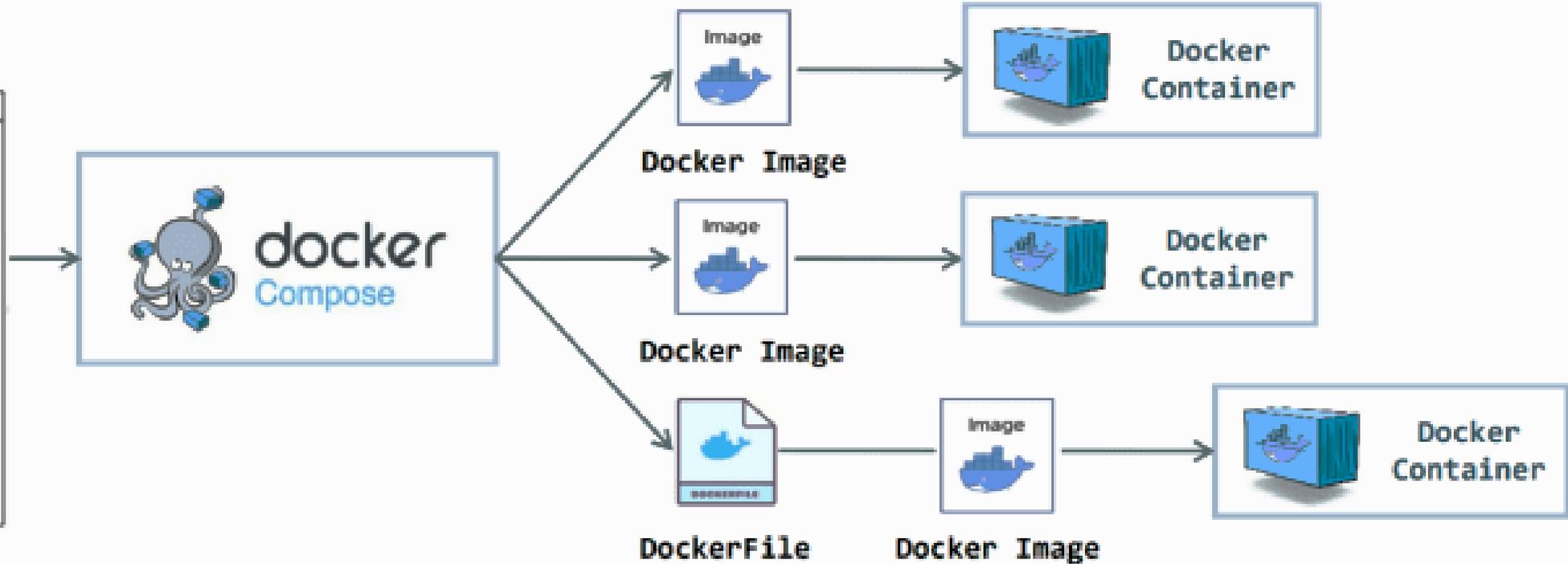


docker
Compose

5. Composición de servicios: Docker compose

docker-compose.yml

```
*** docker-compose.yml
version: "3.7"
services:
  db:
    image: mysql:5.6.19
    restart: always
    environment:
      - MYSQL_DATABASE=example
      - MYSQL_ROOT_PASSWORD=password
  app:
    build: app
    restart: always
  web:
    build: web
    restart: always
    ports:
      - 80:80
```



1. Introducción a Docker y DockerHub
 2. Imágenes
 3. Contenedores
 4. Volúmenes y redes
 5. Composición de servicios: Docker Compose
 6. **Y yo, ¿qué puedo hacer en mi proyecto?**
 7. Tutorial: *docker*
 8. Tutorial: *dockerizando una aplicación sencilla*
 9. Ejercicio práctico: *docker* y *uvlhub*
- 

6. Y yo, ¿qué puedo hacer en mi proyecto?



¡Diseña tu propio despliegue en Docker!

Modifica y/o crea ***Dockerfiles*** propios

Añade nuevos contenedores, volúmenes y redes a tus ***docker-compose.yml***

1. Introducción a Docker y DockerHub
 2. Imágenes
 3. Contenedores
 4. Volúmenes y redes
 5. Composición de servicios: Docker Compose
 6. Y yo, ¿qué puedo hacer en mi proyecto?
 - 7. Tutorial: *docker***
 8. Tutorial: *dockerizando una aplicación sencilla*
 9. Ejercicio práctico: *docker* y uvlhub
- 

7. Tutorial: *docker*



https://1984.lsi.us.es/wiki-egc/index.php/Tutorial_Campo_de_entrenamiento_de_Docker

1. Introducción a Docker y DockerHub
 2. Imágenes
 3. Contenedores
 4. Volúmenes y redes
 5. Composición de servicios: Docker Compose
 6. Y yo, ¿qué puedo hacer en mi proyecto?
 7. Tutorial: *docker*
 8. Tutorial: *dockerizando una aplicación sencilla*
 9. Ejercicio práctico: *docker* y *uvlhub*
- 

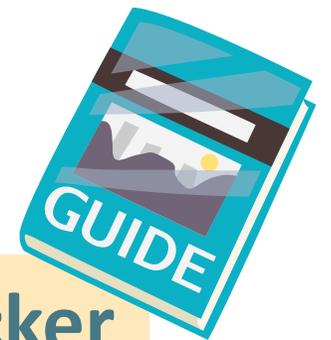
8. Tutorial: *dockerizando* una aplicación sencilla



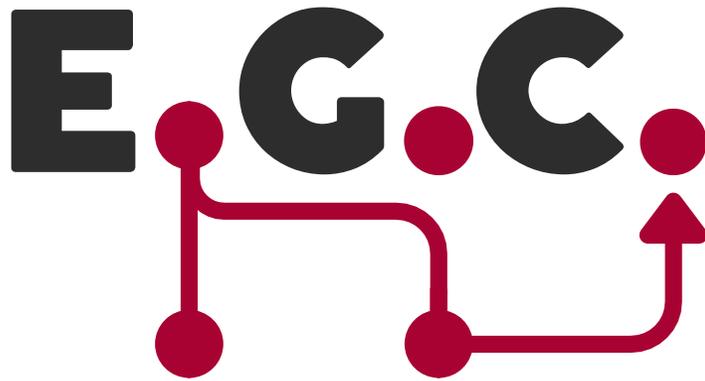
https://1984.lsi.us.es/wiki-egc/index.php/Tutorial_Dockerizando_una_aplicación

1. Introducción a Docker y DockerHub
 2. Imágenes
 3. Contenedores
 4. Volúmenes y redes
 5. Composición de servicios: Docker Compose
 6. Y yo, ¿qué puedo hacer en mi proyecto?
 7. Tutorial: *docker*
 8. Tutorial: *dockerizando una aplicación sencilla*
 9. **Ejercicio práctico: *docker y uvlhub***
- 

9. Ejercicio práctico: docker y uvlhub



docs.uvlhub.io/installation/installation_with_docker



Grado en Ingeniería Informática - Ingeniería del Software

Evolución y Gestión de la Configuración



Escuela Técnica Superior de
Ingeniería Informática

¡Gracias!

*“Dado un número
suficientemente elevado de
ojos, todos los errores se
vuelven obvios.”*

- Eric S. Raymond (ley de Linus)