

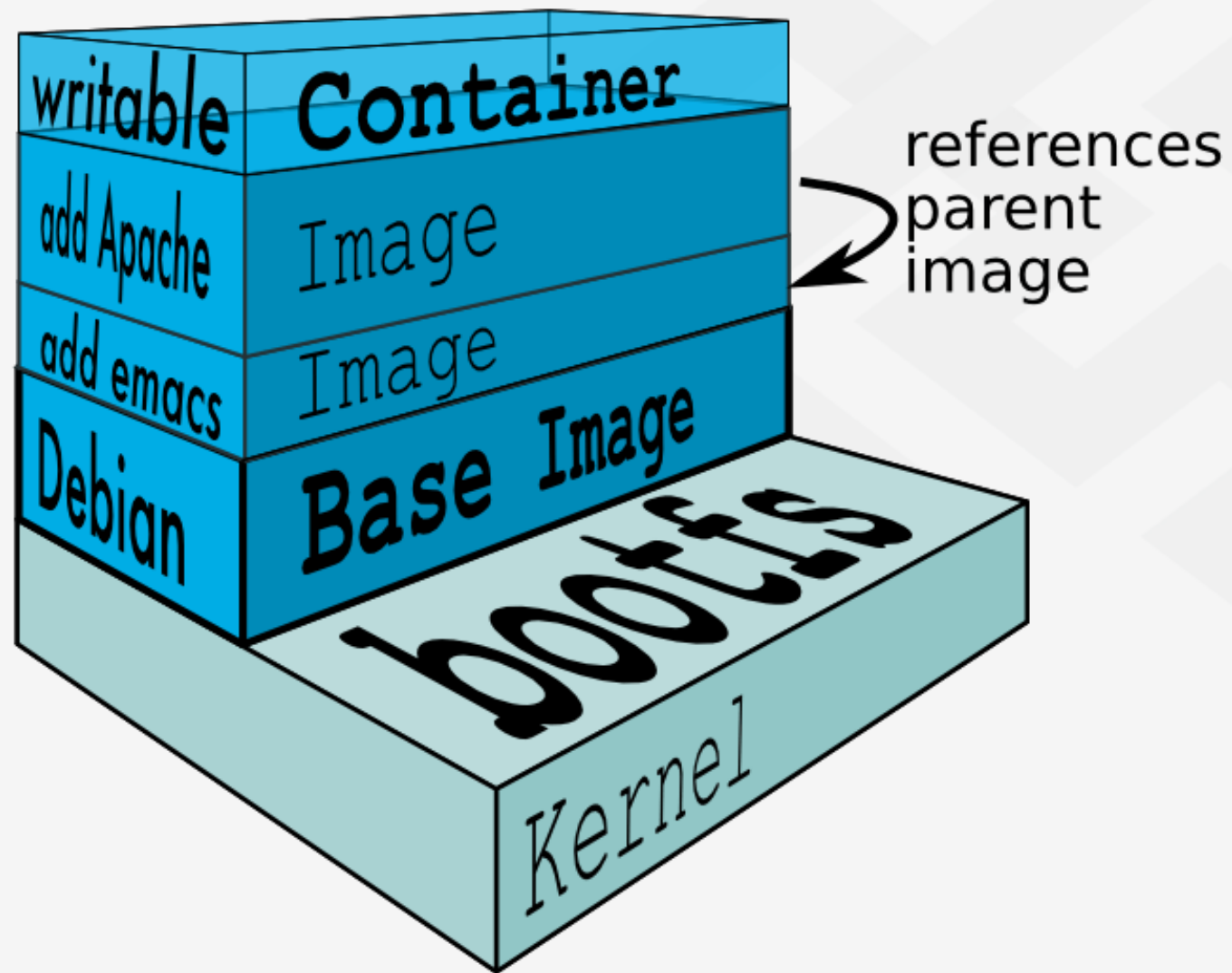


Docker

Temario

Docker Images:

- Imágenes Oficiales?
- Creación de una Imagen
- Docker File
- From/Run/Add/Copy
- Label/User/Volume
- Env/WorkDir/Expose
- CMD - Dockerignore
- Construcción MultiStage-Build Imagen Java
- Construcción MultiStage-Build Angular-NGINX
- Eliminar Imágenes
- Docker Registry
- Conceptos de Docker Hub
- Darse de Alta en Docker Hub
- Crear una Imagen Portable en Docker Hub



Como Comparten las Imágenes los desarrolladores con Docker

Command:

```
docker pull registry.access.redhat.com/rhel7/rhel:latest
```

Decomposition:

access.registry.redhat.com

/

rhel7

/

rhel

:

latest

Generalization:

Registry Server

/

namespace

/

repo

:

tag

Dockerfile

- Es un archivo de texto simple con un conjunto de comandos o instrucciones. Estos comandos / instrucciones se ejecutan sucesivamente para realizar acciones en la imagen base para crear una nueva imagen de la ventana acoplable.

```
FROM ubuntu
MAINTAINER jason@docker.com

RUN apt-get update && apt-get install -y nginx
# Add the nginx configuration file
RUN apt-get install -y nginx
# Add nginx to the repository source list
RUN echo "deb http://download.opensuse.org/repositories/nginx/1.4/nginx1.4.repo" >> /etc/apt/sources.list
RUN apt-get update
# Install nginx package (.deb)
RUN apt-get install -y nginx
# Create the default data directory
RUN mkdir -p /data
# Expose the default port
EXPOSE 80
# Default port to execute the container (nginx)
CMD ["nginx"]
# Set default container command
ENTRYPOINT ["nginx"]
```

Dockerfile

build



Docker Image

run



Docker Container

Comandos de DockerFile

- FROM/RUN/ADD/COPY
- LABEL/USER/VOLUME
- ENV/WORKDIR/EXPOSE
- CMD

- **FROM** indica la imagen base en la que reside este contenedor
- **RUN** La instrucción RUN se usa para ejecutar comandos en la terminal del contenedor
- **ENV** lo ayudará a configurar una variable de entorno en su contenedor. Por lo tanto, puede establecer nombres de usuario, contraseñas, claves API o cualquier cosa que valga la pena establecer como variable de entorno.
- **VOLUME** La instrucción VOLUME indica que los archivos de datos se almacenarán en el directorio dado.
- **COPY** La instrucción COPY se usa para copiar algo de la máquina host a un directorio contenedor
- **EXPOSE** se utiliza para abrir puertos en el contenedor. Esto es muy útil ya que podemos abrir solo los puertos requeridos y mantener nuestro contenedor seguro.
- **CMD** Se usa para iniciar un comando o script cada vez que se inicia el contenedor.

MultiStage-Build

- Las compilaciones de Docker de varias etapas le permiten escribir DockerFile con varias FROM declaraciones. Esto significa que puede crear imágenes que se derivan de varias bases, lo que puede ayudar a reducir el tamaño de su construcción final.
- Las imágenes de Docker se crean seleccionando una imagen base mediante la FROM declaración. Luego agrega capas a esa imagen agregando comandos a su Dockerfile.
- Con compilaciones de varias etapas , puede dividir su Dockerfile en varias secciones. Cada etapa tiene su propia FROM declaración, por lo que puede incluir más de una imagen en sus compilaciones. Las etapas se construyen secuencialmente y pueden hacer referencia a sus predecesoras, por lo que puede copiar la salida de una capa a la siguiente.