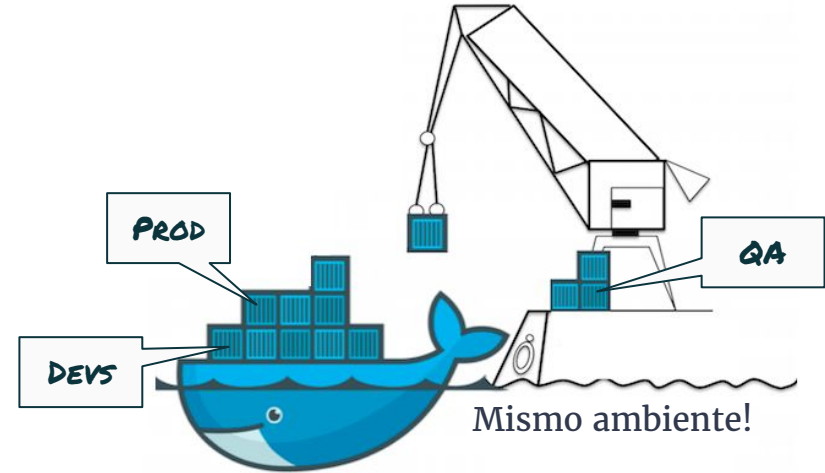
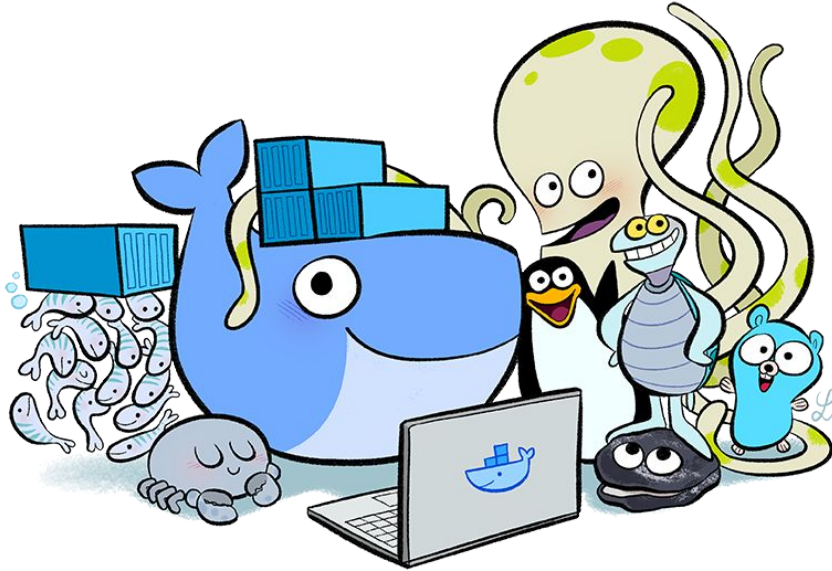




Catherin Cruz  
@UserTwoGG

TODO 1

# Que es Docker?



Ejecuta los **contenedores** independientemente dentro de una sola instancia de Linux.

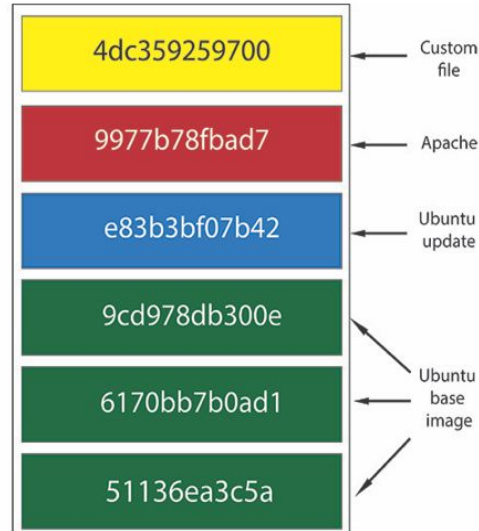
# Componentes Básicos



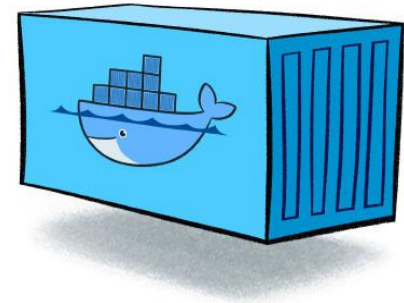
## Dockerfile



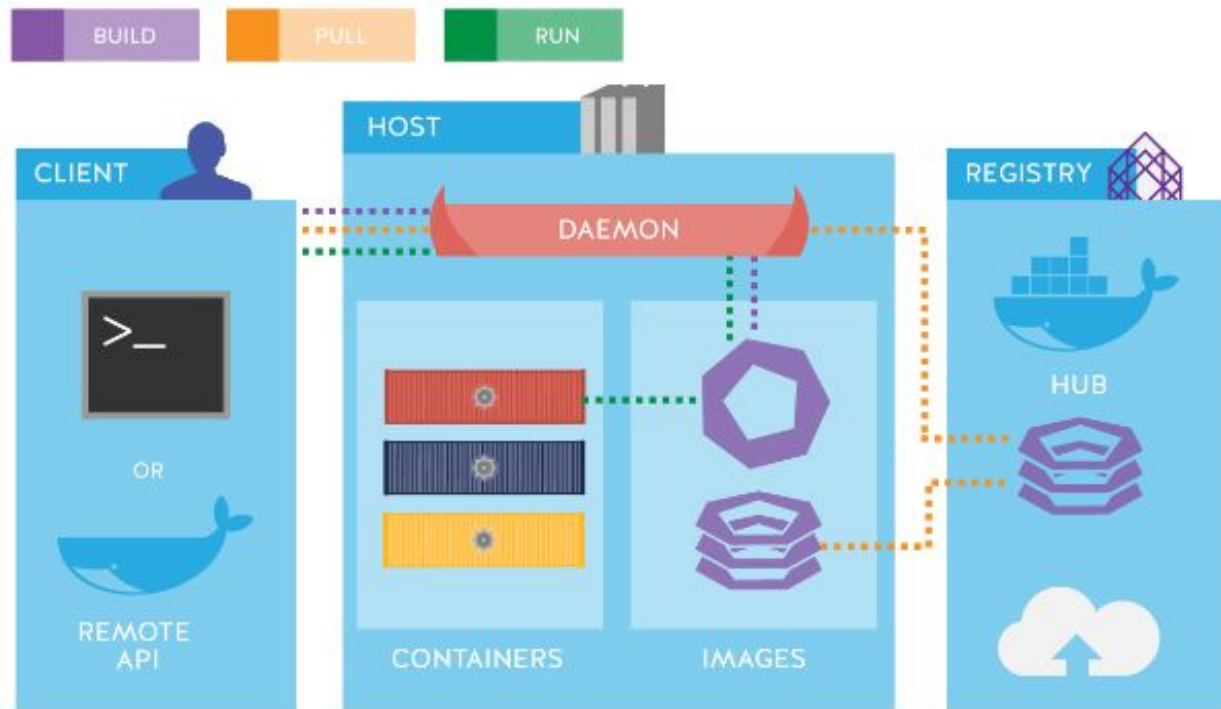
## Image



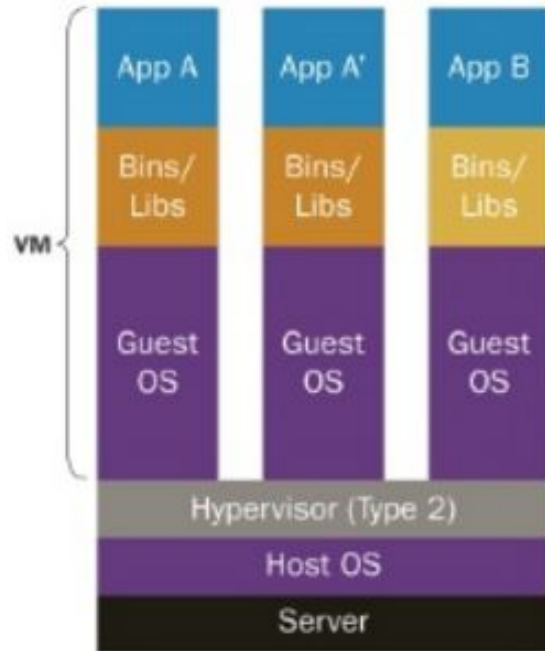
## Container



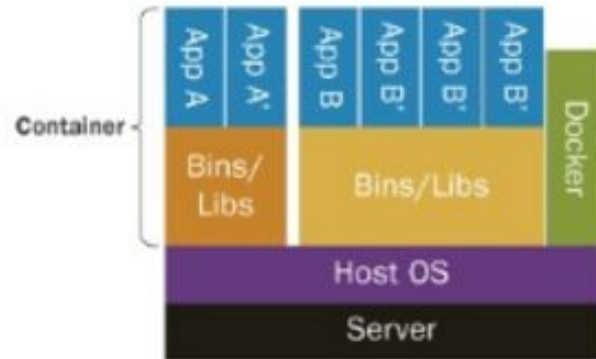
# Arquitectura



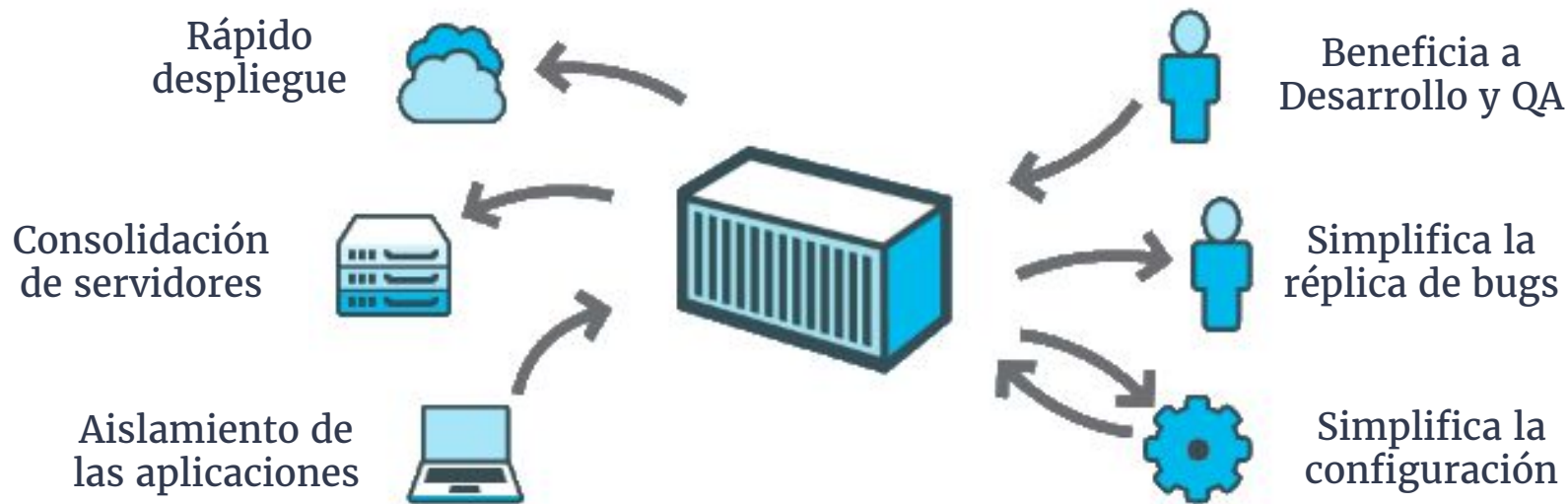
# Contenedores vs. MVs



Los contenedores están aislados pero comparten el SO, cuando es necesario binarios y librerías.



# Beneficios



# Dockerfile



- FROM
- MAINTAINER
- RUN - CMD
- EXPOSE
- ENV
- ADD
- COPY
- ENTRYPOINT
- VOLUME
- USER
- WORKDIR
- HEALTHCHECK



# DockerHub + Go



<https://github.com/twogg-git/go-docker-hub>

- ❑ Obtener una imagen
- ❑ Ejecutar un contenedor
- ❑ Renombrar una imagen
- ❑ Detener una imagen
- ❑ Eliminar contenedores

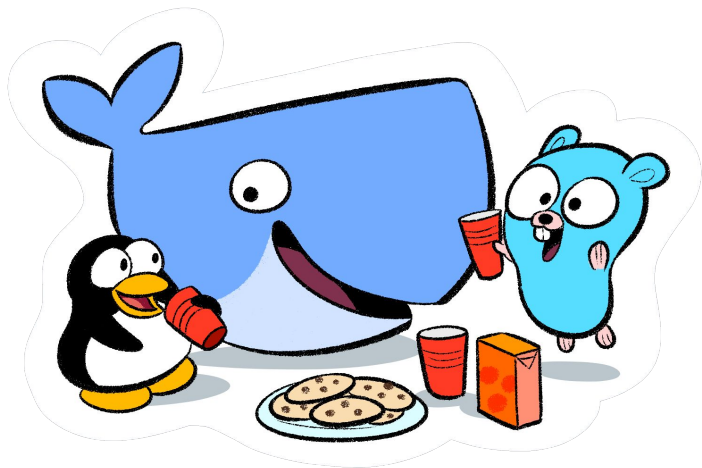




# Docker Ubuntu



```
$ docker run --name ubuntuTODO1 --rm -i -t ubuntu:rolling
```



- ❑ Ejecutar una imagen Ubuntu
- ❑ Usar el bash del contenedor
- ❑ Acceder al bash de forma externa
- ❑ Ejecutar comandos de consola



# Docker + NGinx + HTML



<https://github.com/twogg-git/docker-nginx>



Dockerfile

+



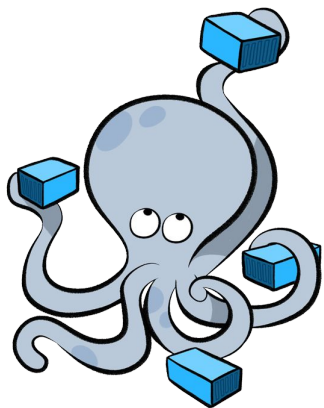
+



# Docker Compose



<https://github.com/twogg-git/docker-compose-java>



Logger DB



Rest API



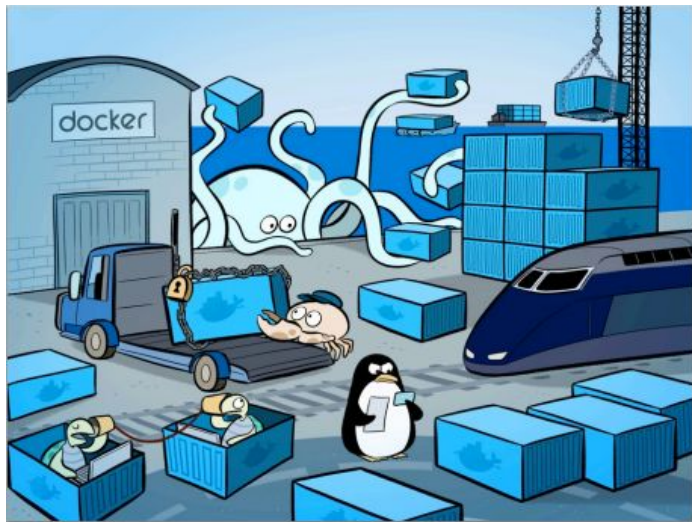
Web Server



SMTP



# Referencias & Links



<http://training.play-with-docker.com/>

<http://labs.play-with-docker.com/>

<https://www.docker.com/what-docker>

<http://blog.arungupta.me/>

<https://www.slideshare.net/Docker>

<http://cloudcomputing.sys-con.com/node/4065127>

