



Docker

Build, Ship, Run



- Docker, c'est quoi ?
- Pour les développeurs
- Et l'intégration et le déploiement ?



- Projet jeune débuté en janvier 2013
- Français mais déporté aux États-Unis
- Projet très actif
- Pré-requis
 - Linux
 - Kernel 3.8+
 - 64 bits



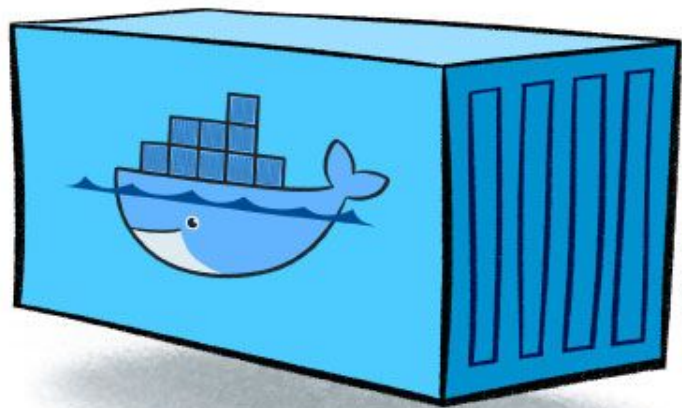
- Empaqueter une application & ses dépendances dans un container facile à déployer
- Ce n'est **pas** :
 - équivalent à une VM
 - 1 logiciel de gestion de conf (Chef, Puppet, Ansible)
- Dans quel cas l'utiliser
 - Automatiser le packaging et le déploiement d'application
 - Création de PaaS
 - Intégration et déploiement continue
 - Tests automatisés
 - Gestion de déploiement d'application scalable



C'est une des notions essentielles

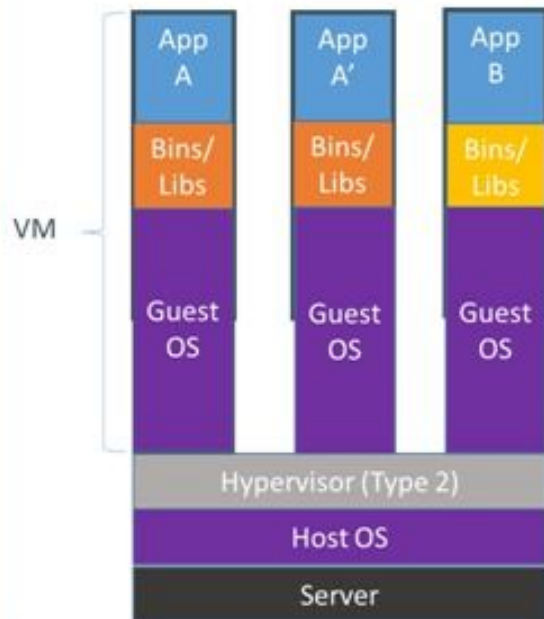
- ce n'est pas nouveau
- LXC
- Isolation

Pourquoi c'est bien ?

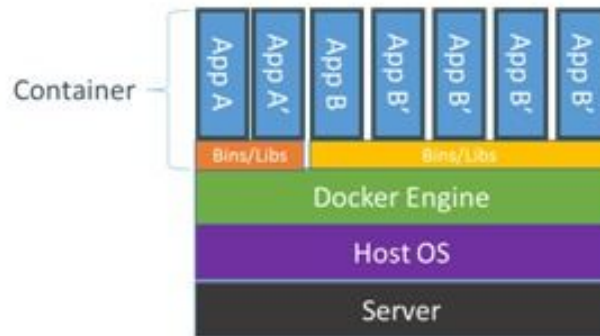


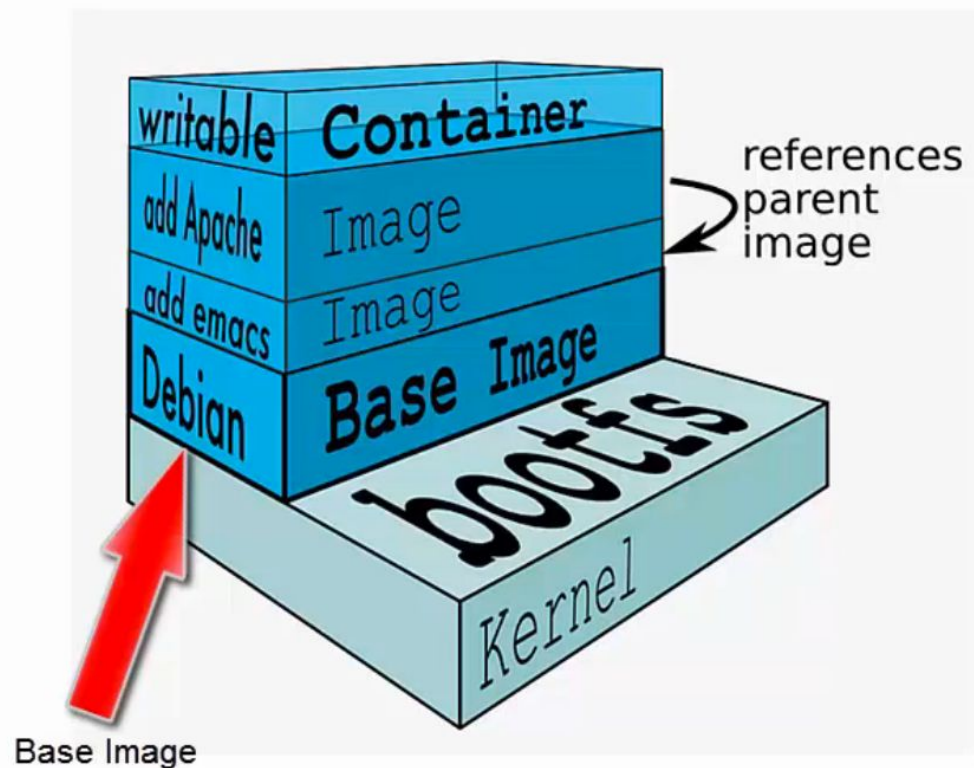


Containers vs. VMs



Containers are isolated, but share OS and, where appropriate, bins/libraries







```
# VERSION 0.1

FROM ubuntu:latest
MAINTAINER polohb <polohb@gmail.com>

# Install node and npm
RUN apt-get update \
    && apt-get install --no-install-recommends -y nodejs npm \
    && apt-get clean \
    && rm -rf /var/lib/apt/lists/* /tmp/* /var/tmp/*

# add link @see install node on ubuntu
RUN ln -s /usr/bin/nodejs /usr/local/bin/node \
    && ln -s /usr/bin/npm /usr/local/bin/npm

# mount volume (current host folder "." as "/src" in the container)
ADD . /src

# Install app dependencies
RUN cd /src; npm install

EXPOSE 8080
CMD ["node", "/src/index.js"]
```




Construction du conteneur :

- `docker build -t polohb/ubuntu-node-hello .`

Exécution du conteneur :

- `docker run --name node-hello -p 49160:8080 -d polohb/ubuntu-node-hello`

Arrêter le container :

- `docker stop node-hello`

Supprimer le conteneur :

- `docker rm node-hello`

Envoyer / Récupérer un container du docker hub :

- `docker push polohb/ubuntu-node-hello`
- `docker pull polohb/ubuntu-node-hello`



Pourquoi ?

- avoir un environnement cohérent entre développeurs
- l'environnement peut-être le même que la prod
- seul docker suffit pour développer
- il est simple d'utiliser plusieurs versions d'un même langage
- le déploiement peut être très simple
- on peut toujours utiliser son éditeur comme avant



Comment faire ?

- pratiquement rien ne change
- s'assurer d'avoir toute les dépendances dans son dossier de travail
- lancer les commandes de build and run dans un container docker



- exemple java simple
- exemple java avec maven
- autres exemples node ?



- La suite une autre fois peut-être :)