

# Docker

Atelier d'aide à la programmation

Léo BAUDOUIN

`baudouin.leo@gmail.com`

03-04 juin 2019

# Définition



## Wikipedia

« *Docker est un outil qui peut emballer une application et ses dépendances dans un conteneur isolé, qui pourra être exécuté sur n'importe quel serveur* ».

*Ceci permet d'étendre la flexibilité et la portabilité d'exécution d'une application, que ce soit sur la machine locale, un cloud privé ou public, une machine nue...*

# Installer Docker

## Installer et configurer

```
sudo apt-get install -y docker.io  
sudo groupadd docker  
sudo usermod -aG docker $USER
```

## Docker for Windows

<https://docs.docker.com/docker-for-windows/install/>

# Démarrer un conteneur

Télécharger une image

```
docker pull ubuntu:18.04
```

Démarrer un conteneur

```
docker run --rm -it ubuntu:18.04
```

# Créer une image pour L<sup>A</sup>T<sub>E</sub>X

## Dockerfile

```
FROM ubuntu:18.04
```

```
RUN apt-get update
```

```
RUN apt-get upgrade -y
```

```
RUN apt-get install -y texlive-full
```

# Créer une image pour Angular

## Dockerfile

```
FROM ubuntu:18.04

RUN apt-get update && apt-get -y upgrade
RUN apt-get install -y npm

RUN npm install -g npm@latest
RUN npm install -g @angular/cli
RUN npm install --save-dev @angular-devkit/build-angular
RUN npm install --save-dev @angular/compiler-cli
RUN npm install --save-dev @angular/compiler
WORKDIR /sources

EXPOSE 8082
EXPOSE 4200

# ng serve --host 0.0.0.0
```

# Créer une image avec une interface

## Dockerfile

```
FROM ubuntu:16.04

RUN apt-get update
RUN apt-get install -y x-window-system
RUN apt-get install -y binutils
RUN apt-get install -y mesa-utils
RUN apt-get install -y module-init-tools

ADD nvidia-driver.run /tmp/nvidia-driver.run
RUN sh /tmp/nvidia-driver.run -a -N --ui=none --no-kernel-module
RUN rm /tmp/nvidia-driver.run

RUN apt-get install -y nano

CMD /bin/bash

WORKDIR /sources
```

# docker-compose

## Wikipedia

*Docker **compose** est un outil très intéressant de gestion de package docker. Cet outil va lancer vos conteneurs et leurs éventuels liens à partir d'un fichier de configuration écrit en yaml.*



## docker-compose.yml

```
version: '3.3'
services:
  db:
    image: mysql:5.7
    volumes:
      - db_data:/var/lib/mysql
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: somewordpress
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wordpress
      MYSQL_PASSWORD: wordpress
  wordpress:
    depends_on:
      - db
    image: wordpress:latest
    ports:
      - "8000:80"
    restart: always
    environment:
      WORDPRESS_DB_HOST: db:3306
      WORDPRESS_DB_USER: wordpress
      WORDPRESS_DB_PASSWORD: wordpress
  adminer:
    image: adminer
    depends_on:
      - db
    ports:
      - "8080:8080"
    restart: always
volumes:
  db_data:
```

# Démarrer / Arrêter

## Gestion

```
docker-compose up [-d]
```

**-d** = run containers in the background

```
docker-compose down
```

# Gérer ses dockers

## CLI

```
docker ps [-a]  
docker images
```

## Portainer

```
docker volume create portainer_data  
docker run -d -p 9000:9000 --restart always -v  
/var/run/docker.sock:/var/run/docker.sock -v  
portainer_data:/data portainer/portainer
```

# Gérer ses dockers

## Portainer

The screenshot shows the Portainer web interface in a browser. The address bar displays '127.0.0.1:9000/#/containers'. The interface has a dark blue sidebar with navigation icons. The main content area is titled 'Container list' and includes a 'Containers' section with a search bar and a table of running and stopped containers. The table columns are Name, State, Quick actions, Stack, Image, Created, and IP Address. The 'State' column uses color-coded labels: green for 'running' and red for 'stopped'.

Container list

Containers

Portainer support admin  
[my account](#) [log out](#)

Containers Columns Settings

Start Stop Kill Restart Pause Resume Remove Add container

Search...

| <input type="checkbox"/> | Name              | State<br>Filter | Quick actions | Stack | Image                       | Created             | IP Address |
|--------------------------|-------------------|-----------------|---------------|-------|-----------------------------|---------------------|------------|
| <input type="checkbox"/> | portainer         | running         |               | -     | portainer/portainer         | 2019-04-29 21:36:15 | 172.17.0.1 |
| <input type="checkbox"/> | nginx             | running         |               | -     | nginx:latest                | 2018-10-03 00:32:19 | 172.17.0.1 |
| <input type="checkbox"/> | gitlab-runner     | stopped         |               | -     | gitlab/gitlab-runner:latest | 2019-04-28 20:19:24 | -          |
| <input type="checkbox"/> | gitlab            | stopped         |               | -     | gitlab/gitlab-ce:latest     | 2019-04-28 11:59:14 | -          |
| <input type="checkbox"/> | pedantic_pascal   | stopped         |               | -     | 0eb3f8a30aab                | 2019-04-10 10:08:42 | -          |
| <input type="checkbox"/> | recurring_hodgkin | stopped         |               | -     | 6267943e9b32                | 2019-04-10 10:01:53 | -          |