

DevOps avec Docker

Guide pratique

Les avantages de Docker

- Flexible
- Rapide à build
- Performant (aussi une contrainte 🧑🏻)
- Sécurisé (*les containers sont isolés et optionnellement sur des sous réseaux différents*)
- Scalable (k8s)
- Substituable (Principe SOLID https://fr.wikipedia.org/wiki/Principe_de_substitution_de_Liskov)
- Simple (construire le projet peut se faire sur n'importe quel environnement)

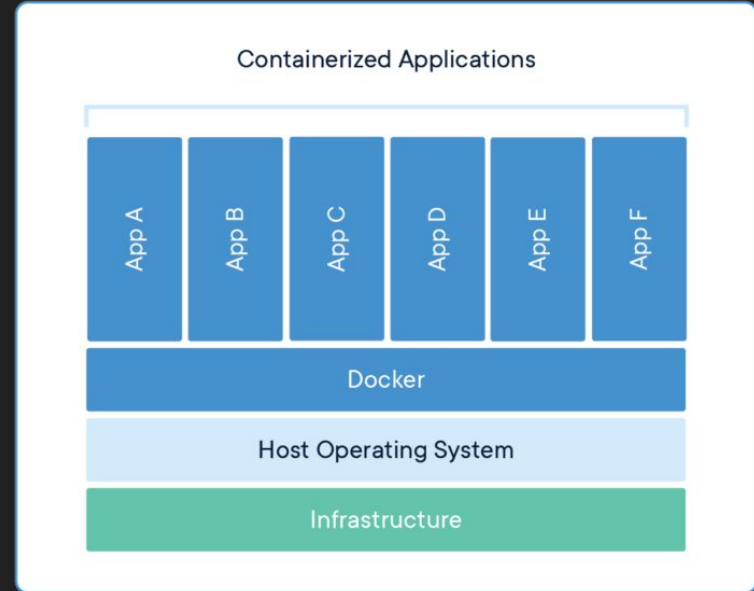
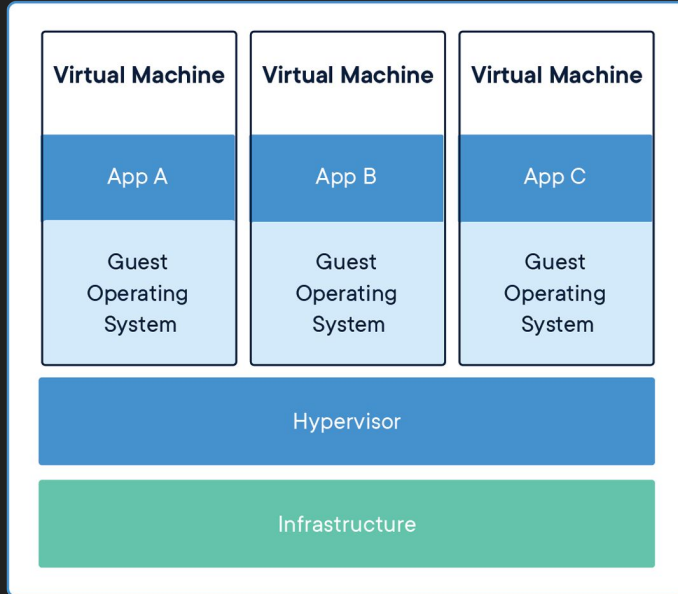
Les contraintes de docker

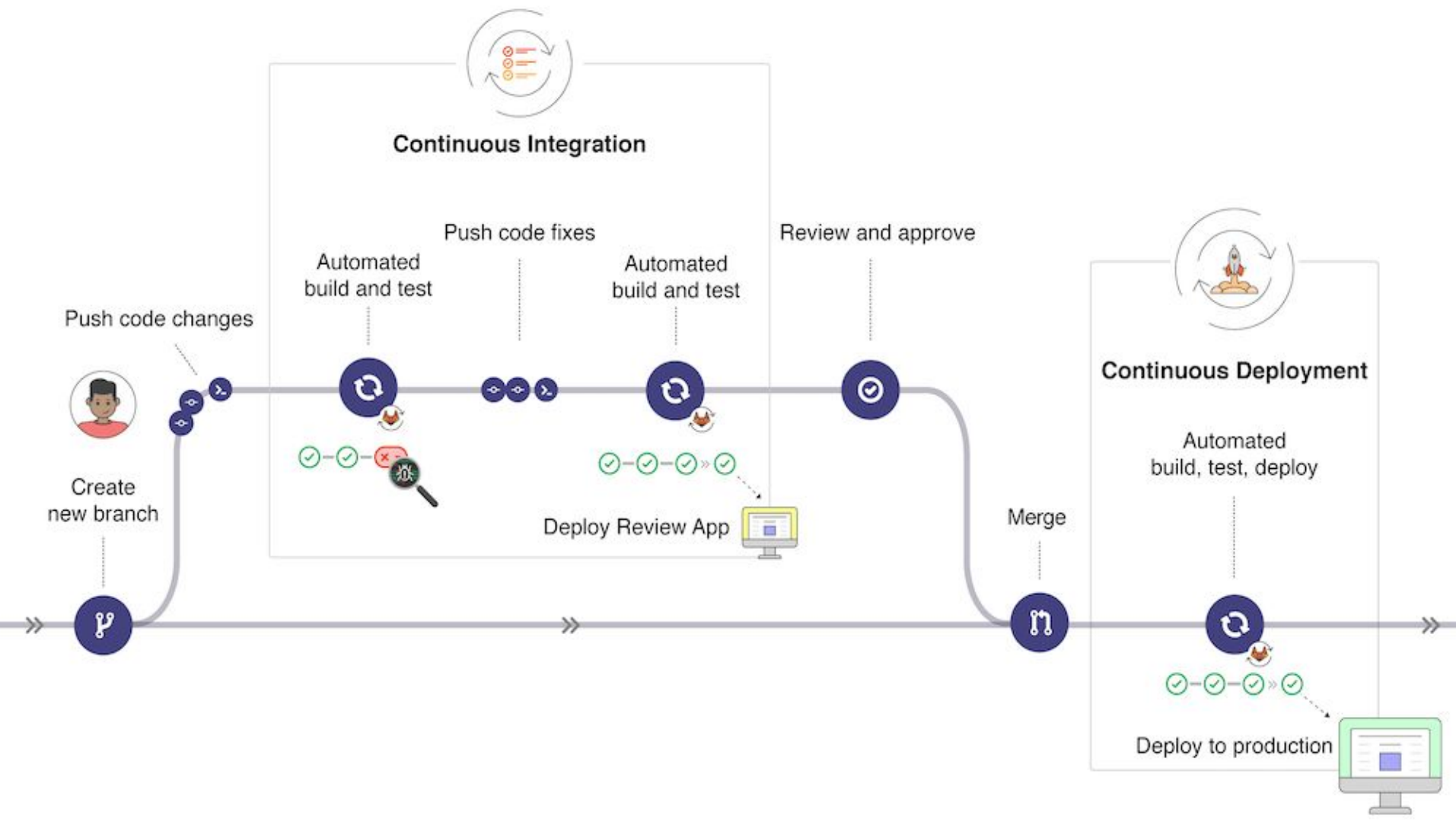
- Optimisé pour Linux (c'est pas un avantage ca??!)
- Ne correspond à tous les projets

(GUI desktop mais en vrai ca reste utile quand même pour simuler des parties du système, l'API par exemple)

- Système de fichier limité en taille (peut être aisément contourné avec un CDN)
- Difficile à monitorer
- Moins performant que l'équivalent non conteneurisé
- La sécurité, oui les containers sont isolés, mais une politique trop ouverte sur un des conteneurs peut aussi corrompre toute la sécurité du projet.

Conteneurs VS machine virtuelle





Docker cheatsheet

Lister les conteneurs

```
docker ps -a
```

Se connecter dans un container en cours d'exécution:

```
docker exec -it (ID du container issue de docker ps) bash
```

Afficher les logs d'un conteneur

```
docker logs (ID du container issue de docker ps)
```

Stopper tous les conteneurs

```
docker stop $(docker ps -aq)
```

Supprimer tous les conteneurs

```
docker rm $(docker ps -aq)
```

Supprimer toutes les images téléchargés

```
docker rmi $(docker images -q)
```

Récupérer tout l'espace disque alloué aux images et aux conteneurs (utiliser le flag --volumes pour cleaner aussi les volumes)

```
docker system prune
```

Docker compose cheatsheet

Build et run un projet

`docker-compose up --build` (l'option `-d` permet de faire tourner en background)

Rebuild from scratch les containers (attention perte de données)

`docker-compose down && docker-compose up --build`

Afficher tous les logs des conteneur

`docker-compose logs`

Stopper tous les conteneurs

`docker stop $(docker ps -aq)`

Supprimer tous les conteneurs

`docker rm $(docker ps -aq)`

Supprimer toutes les images téléchargés

`docker rmi $(docker images -q)`

Dépendances entre les containers

```
37 ► php-fpm:
38     container_name: php-fpm
39     build:
40     context: ../php-fpm
41     depends_on:
42     - db
43     environment:
44     - APP_ENV=${APP_ENV}
45     - APP_SECRET=${APP_SECRET}
46     - DATABASE_URL=${DATABASE_URL}
47     volumes:
48     - ../../symfony/:/var/www
```


Dockerfile obligatoire?

```
24  redis:
25    image: "redis:alpine"
26    command: redis-server --requirepass ${REDIS_DEFAULT_PASS}
27    ports:
28      - "6379:6379"
29    volumes:
30      - $PWD/redis/data:/var/lib/redis
31      - $PWD/redis/redis.conf:/usr/local/etc/redis/redis.conf
32    environment:
33      - REDIS_REPLICATION_MODE=master
34    depends_on:
```

Utiliser des variables d'environnement?

Conf avec Dotenv (fichier .env à la racine)

```
37 ► php-fpm:
38     container_name: php-fpm
39     build:
40     context: ../php-fpm
41     depends_on:
42     - db
43     environment:
44     - APP_ENV=${APP_ENV}
45     - APP_SECRET=${APP_SECRET}
46     - DATABASE_URL=${DATABASE_URL}
47     volumes:
48     - ../../symfony/:/var/www
```

```
17     APP_ENV=dev
18     APP_SECRET=743df4115e7e1dea13b473da07c09fe6
19     ###< symfony/framework-bundle ###
20
21     ###> doctrine/doctrine-bundle ###
22     # Format described at https://www.doctrine-project.org/projects/doctri
23     # IMPORTANT: You MUST configure your server version, either here or in
24     #
25     # DATABASE_URL="sqlite:///kernel.project_dir%/var/data.db"
26     # DATABASE_URL="mysql://db_user:db_password@127.0.0.1:3306/db_name?ser
27
28     DATABASE_URL="postgresql://postgres:password@db:5432/crypto-api-db"
```

Alternatives libres

LXC <https://linuxcontainers.org/lxc/introduction/>

BSD Jails <https://docs.freebsd.org/en/books/handbook/jails/>

RKT <https://cloud.redhat.com/learn/topics/rkt>

...

Solaris containers

<https://www.oracle.com/solaris/technologies/solaris-containers.html>