

docker

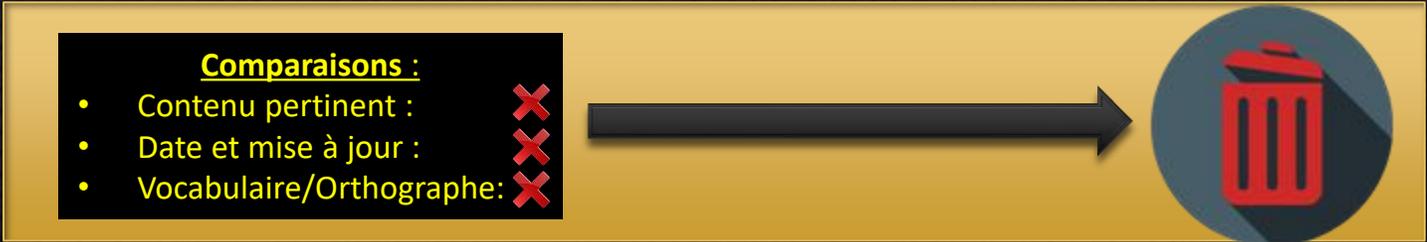
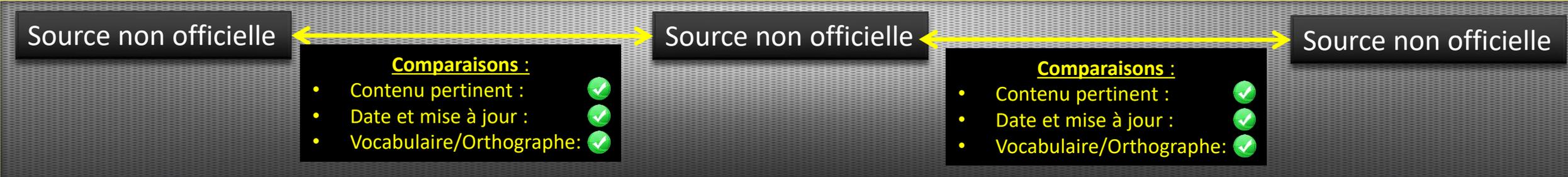
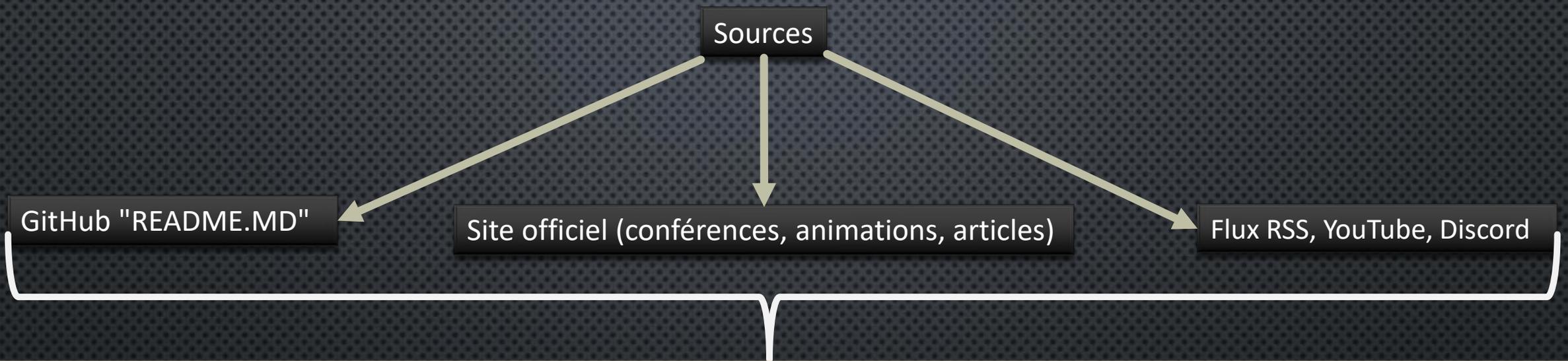
[Veille technologique]

IMIE RENNES – ADMINISTRATEUR SYSTEMES ET RESEAUX

[2017-2018]

- Nom et prénom de l'apprenti : AJDAINI Hatim
- Nom et prénom du tuteur : JANY Franck
- Entreprise : OTIMA

PROCÉDURES DE RECHERCHES DES SOURCES



QUELQUES ASTUCES DE VEILLE

DISCORD

F4FIA
WebAuthn : le futur du web sans mot de passe se rapproche | Je
https://www.journaldugeek.com/2018/04/12/webauthn-future-passe-se-rapproche/?utm_campaign=twitter&utm_medium=twitter&utm_source=twitter

13 avril 2018

Mr Reddit **BOT** Hier à 12:37

Aguay_val
LXC Sortie en version 3.0 !
<https://linuxcontainers.org/fr/lxc/news/>

14 avril 2018

Mr Reddit **BOT** Aujourd'hui à 14:32

F4FIA
Sauvegarder la mémoire de notre société sur la Lune : l'étonnant projet Sanctuary | Journal du Geek
https://www.journaldugeek.com/2018/04/11/sauvegarder-memoire-societe-lune-letonnante-mission-projet-sanctuary/?utm_campaign=twitter&utm_medium=twitter&utm_source=twitter

Envoyez **Serveur : sysadmin_fr**

Alertes

Recevez des alertes lorsque du contenu susceptible de vous intéresser est publié sur le Web

Google

docker container

Fréquence: Une fois par jour maximum

Sources: Automatique

Langue: français

Région: Toutes les régions

Nombre de résultats: Seulement les meilleurs résultats

Envoyer à: ajdaini.hatim@gmail.com

Créer l'alerte Masquer les options

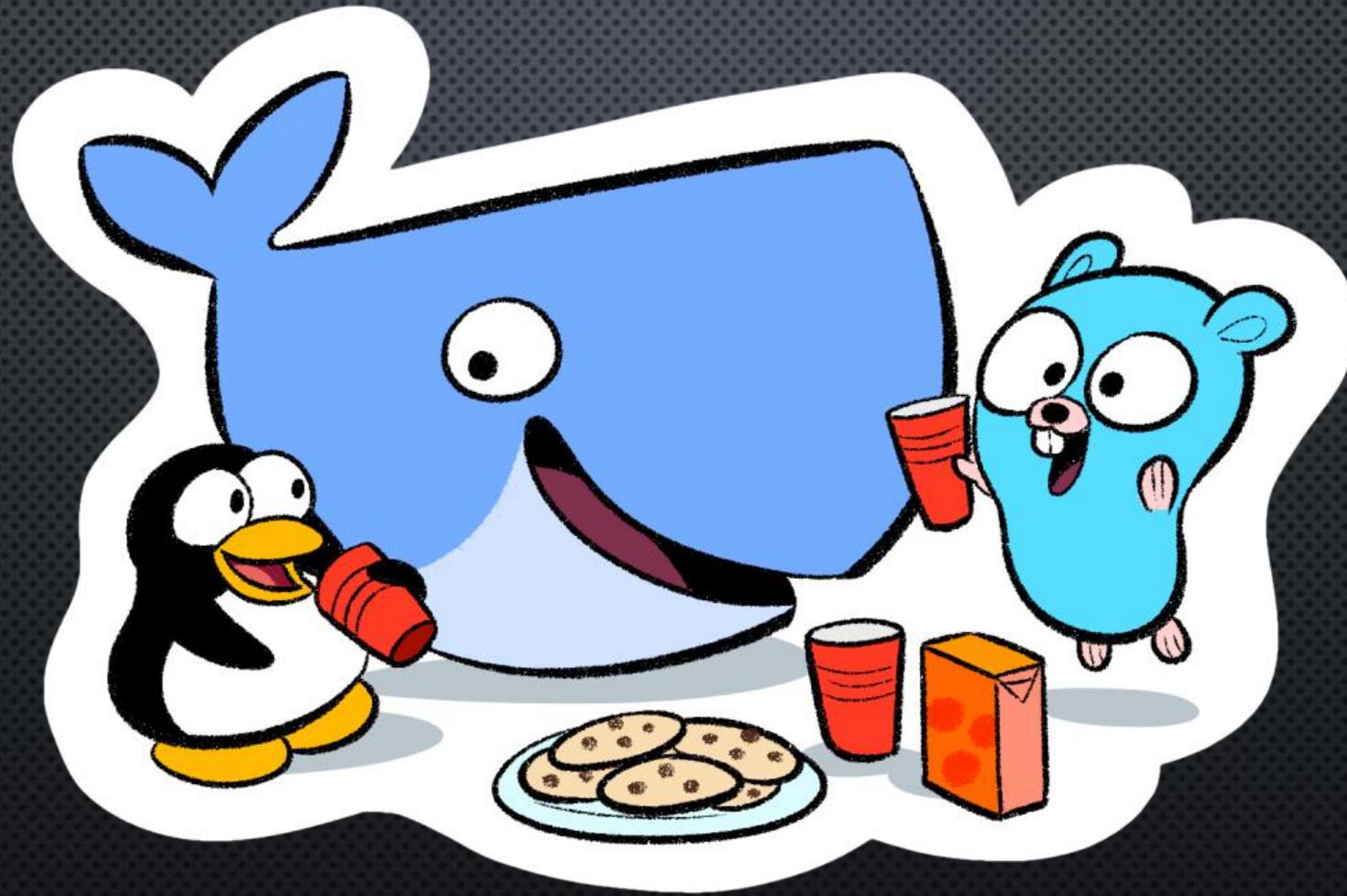
Aperçu de l'alerte

Aucun résultat récent ne correspond à votre requête de recherche. Veuillez trouver ci-dessous les autres résultats qui correspondent à votre recherche.

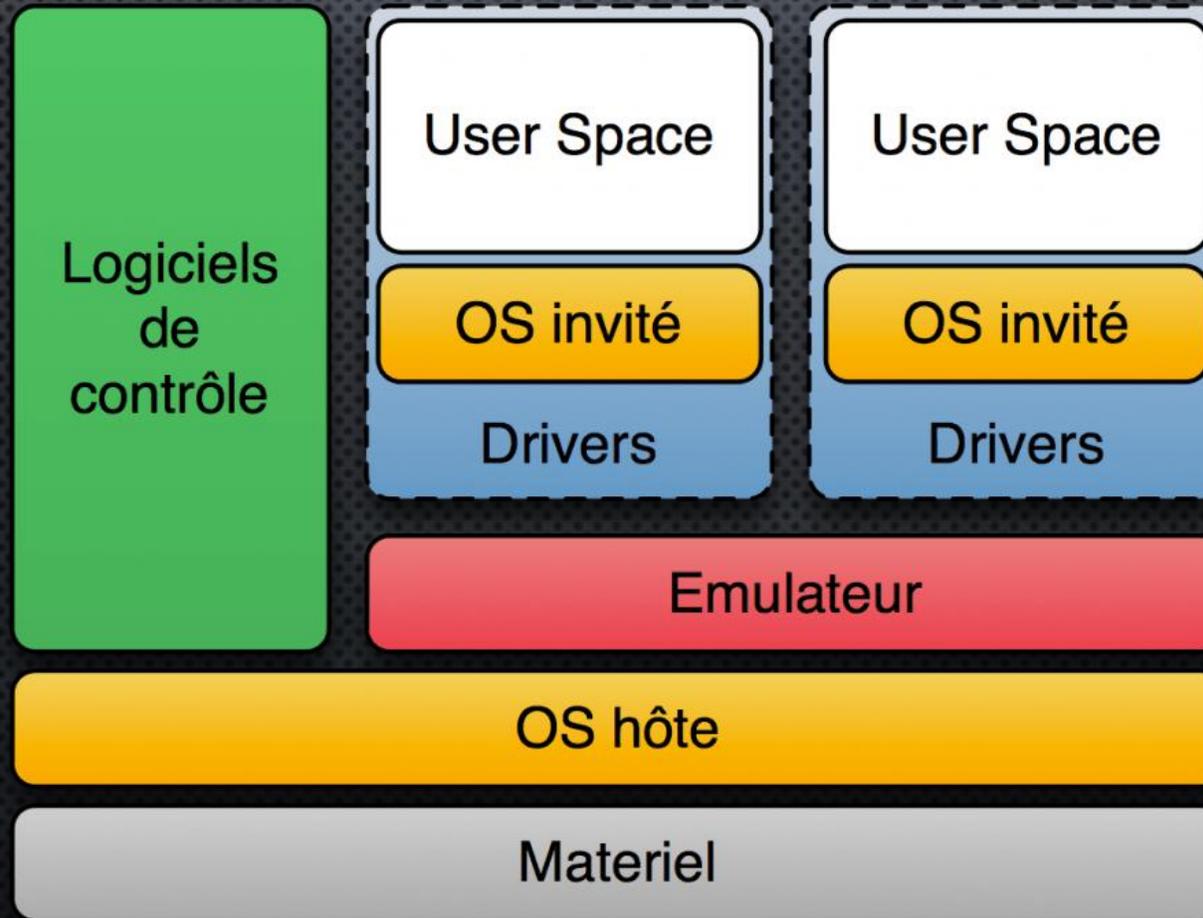
ACTUALITÉS

Docker souffle ses 5 bougies et atteint 37 milliards de **containers téléchargés**
JDN
Le 15 mars 2013, à l'occasion de la conférence PyCon, Solomon Hykes levait le voile sur la technologie Docker. Ce jeune français, alors âgé de 29 ans, dévoilait le concept de **container** logiciel. "Il est passé après tout le monde. Il ne restait que très peu de temps. Il n'a eu que cinq minutes pour faire sa ...

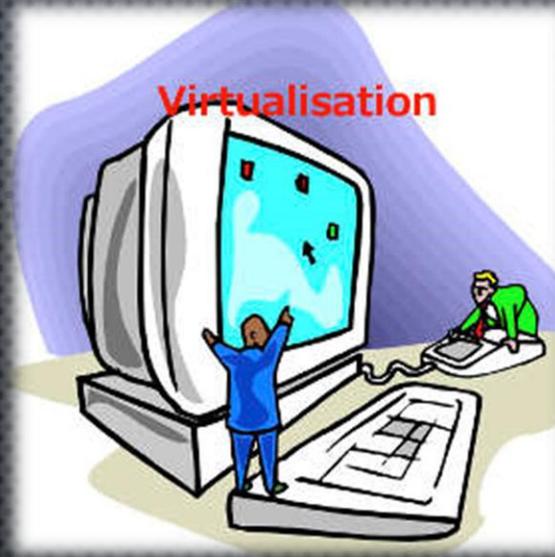
INTRODUCTION



VIRTUALISATION

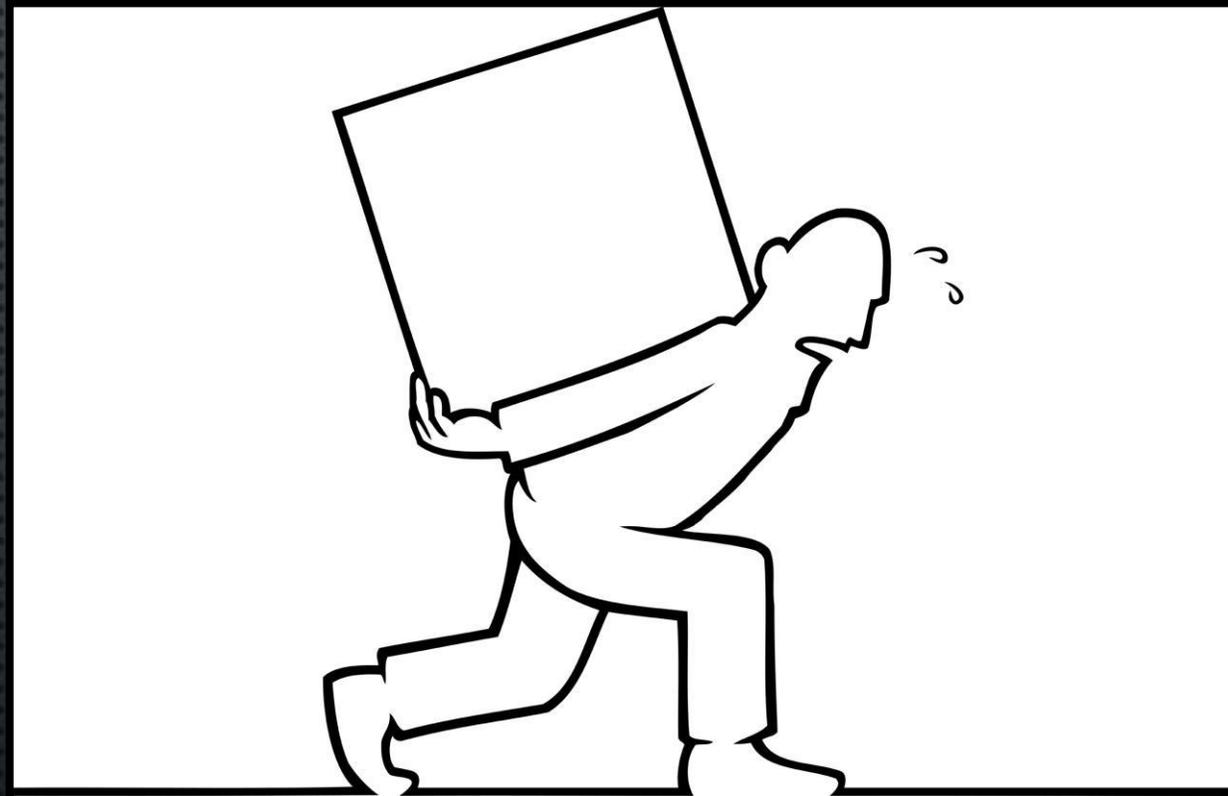


AVANTAGES VM



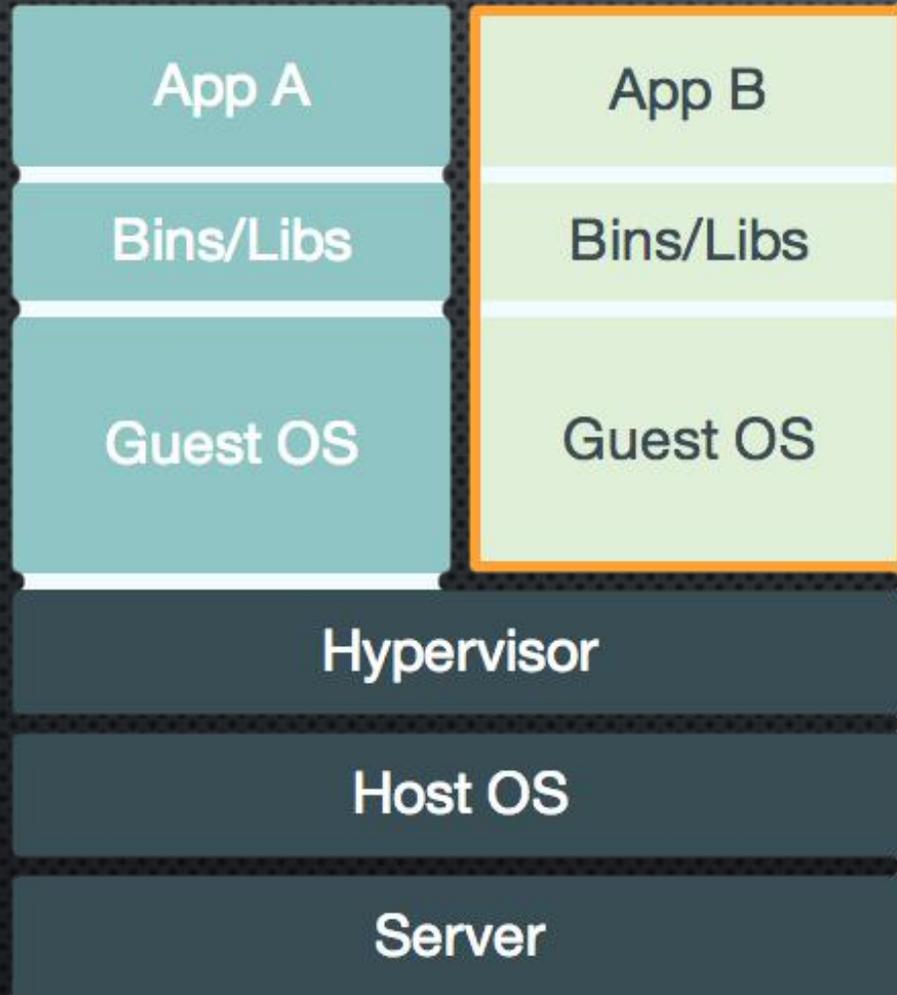
- Consacrer les ressources adaptées selon les applications utilisées, et cela à volonté.
- Les machines virtuelles sont simples à manier.
 - Sauvegarder
 - Dupliquer
- Coûts = économique:
 - Réduction de la quantité d'équipement nécessaire
 - Alimentation et de refroidissement des composants.

INCONVÉNIENTS VM



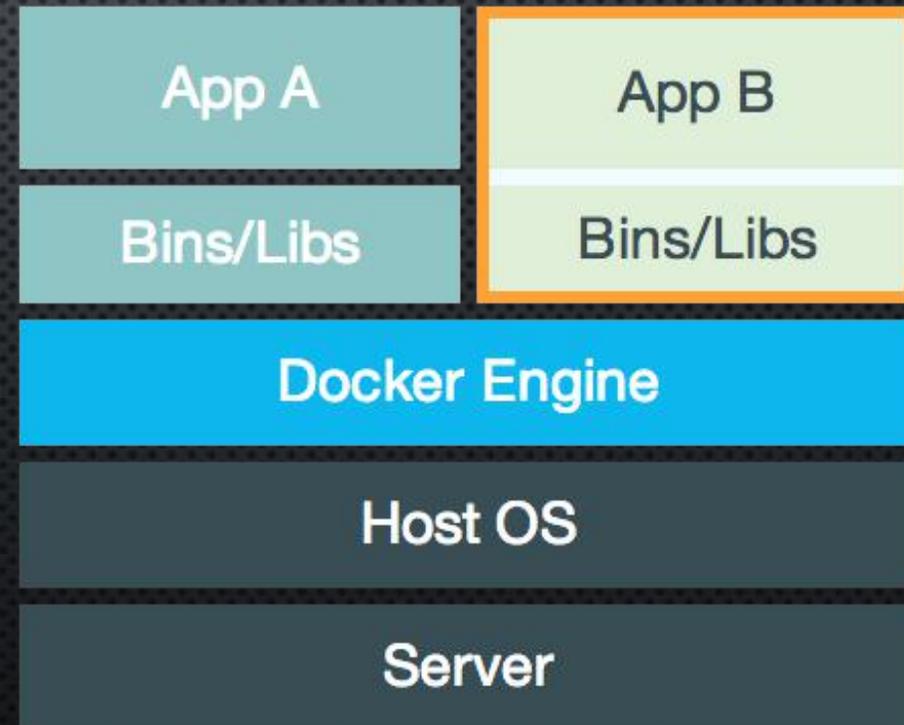
Chaque machine virtuelle embarque un OS qui consomme à lui tout seul énormément de ressources alors qu'au final, ce qui nous intéresse c'est la ou les applications qui vont tourner dessus.

DOCKER VS VM

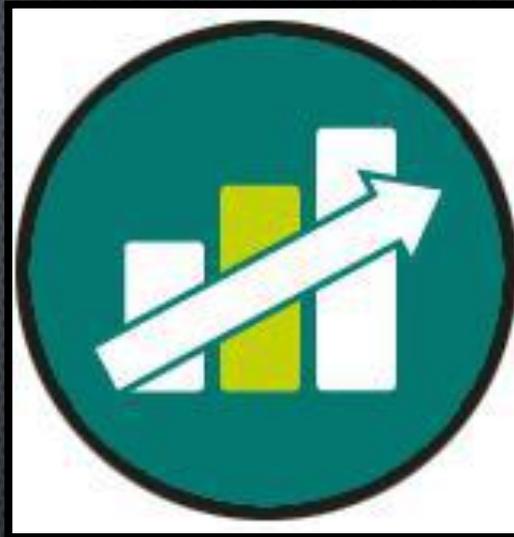


Docker ?

Une start-up spécialisée dans les conteneurs et fondée en France et devenue, ces derniers mois, l'une des étoiles montantes de la Silicon Valley



AVANTAGES DOCKER



- Pas d'OS allant de plusieurs Giga-octets. Le conteneur appelle directement l'OS pour réaliser ses appels système et exécuter ses applications. Il est beaucoup moins gourmand en ressources.
- La portabilité ! On peut déplacer les conteneurs d'un environnement à l'autre sans perte de données. À la manière de la machine virtuelle que l'on peut déplacer de serveurs à serveurs mais en gardant cette couche OS allant jusqu'à Giga-octets.

INCONVÉNIENT DOCKER

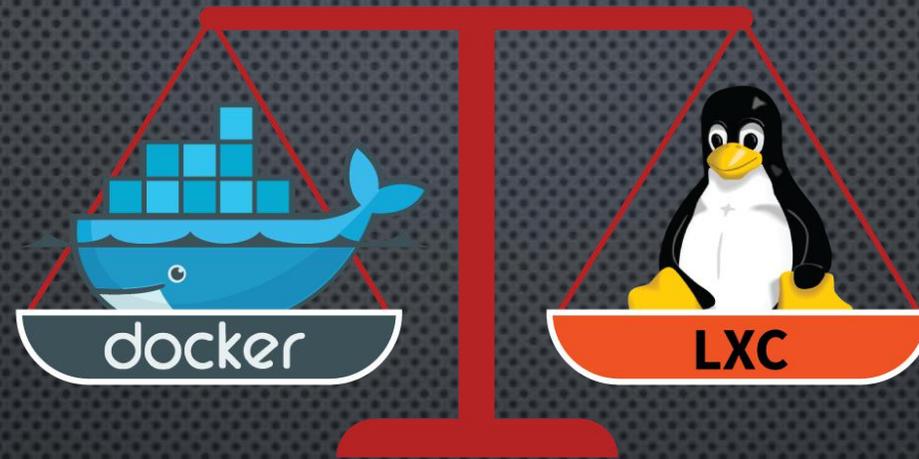


On ne peut pas déployer un conteneur prévu pour Linux sur un environnement Windows et vice-versa

HISTORIQUE DES CONTENEURS

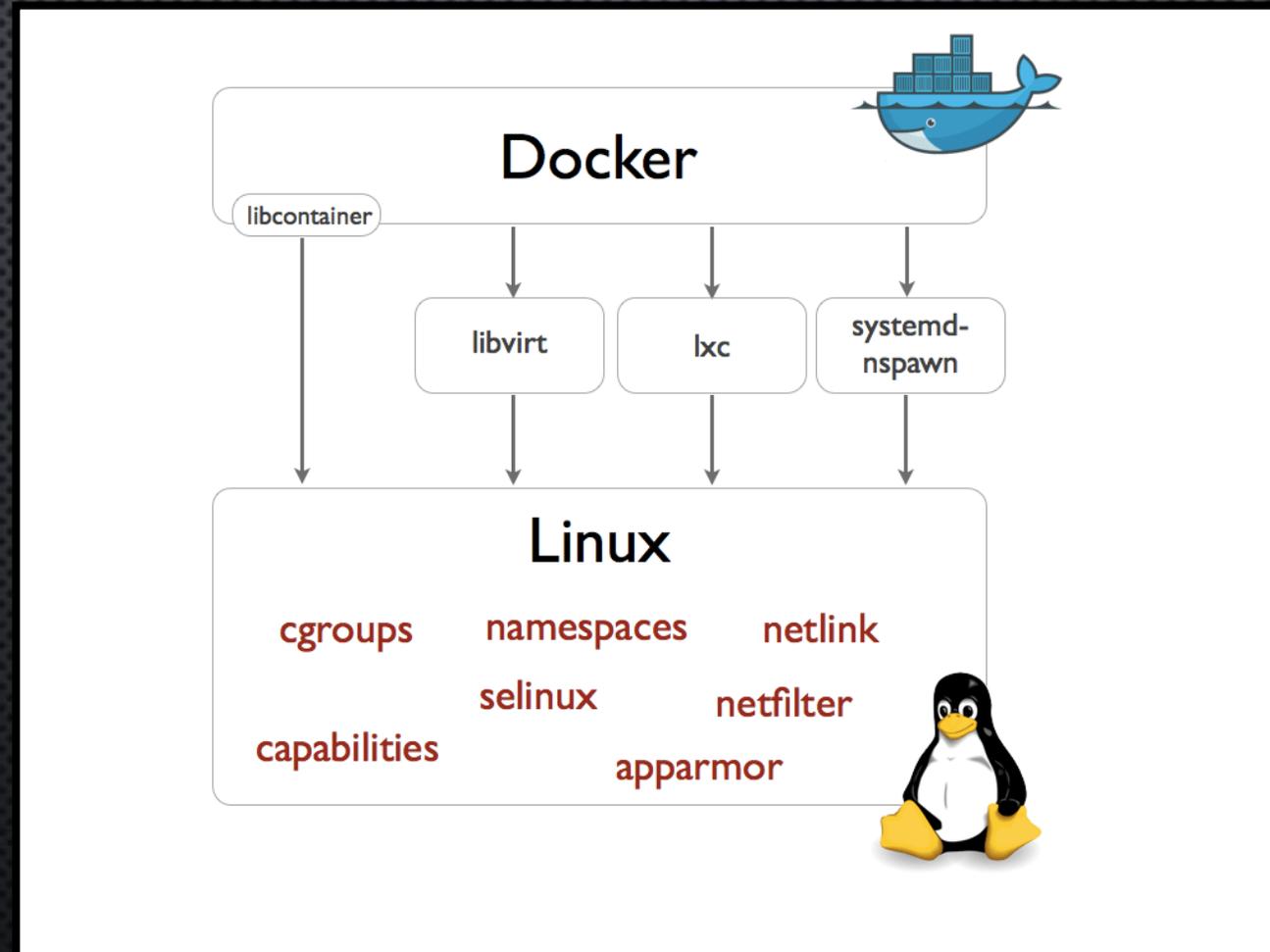
- Année 2000 jail FreeBSD
- Nouvelles technologies :
 - **cgroups** : *Contrôler et limiter l'utilisation des ressources pour un processus ou un groupe de processus*
 - **Les espaces de noms utilisateur** : *les utilisateurs et des groupes peuvent avoir des privilèges pour effectuer certaines opérations au sein du conteneur, mais que ces privilèges ne leurs sont pas accordés en dehors du conteneur*
- Le projet LXC (linux container) a ensuite enrichi la technologie qui ont permis d'améliorer l'expérience des utilisateurs de conteneurs.
- le noyau de Linux 2.6.24 intègre les LXC

DOCKER LXC AUGMENTÉ



- Docker ne remplace pas les conteneurs Linux, l'idée consiste à utiliser manipuler plus facilement les LXC.
- La technologie Open Source Docker est actuellement le projet le plus connu et utilisé pour déployer et gérer des conteneurs Linux.
- Aujourd'hui, Red Hat et Docker font partie des nombreux membres de l'OCI (Open Container initiative), qui travaille à la création de normes ouvertes pour les technologies de conteneurs à l'échelle du secteur.

LIBCONTAINER



[Blog docker par Solomon Hykes](#)

IMAGE OFFICIELS DE DOCKER

- Depuis le site : <https://hub.docker.com/explore/> (plus de 100 000 images gratuites)
- Kitematic (comme Google store)

ne (D:) > kitematic

Name	Date modified	Type	Size
locales	7/29/2016 11:37 AM	File folder	
resources	7/29/2016 11:37 AM	File folder	
content_resources_200_percent.pak	7/27/2016 9:06 PM	PAK File	1 KB
content_shell.pak	7/27/2016 9:06 PM	PAK File	8,520 KB
d3dcompiler_47.dll	7/27/2016 9:06 PM	Application extens...	4,077 KB
icudtl.dat	7/27/2016 9:06 PM	DAT File	9,968 KB
Kitematic.exe	7/27/2016 9:06 PM	Application	69,196 KB
libEGL.dll	7/27/2016 9:06 PM	Application extens...	13 KB
libGLESv2.dll	7/27/2016 9:06 PM	Application extens...	2,027 KB
LICENSE	7/27/2016 9:06 PM	File	2 KB
LICENSES.chromium.html	7/27/2016 9:06 PM	HTML File	1,092 KB
msvcp120.dll	7/27/2016 9:06 PM	Application extens...	645 KB
msvcr120.dll	7/27/2016 9:06 PM	Application extens...	941 KB
natives_blob.bin	7/27/2016 9:06 PM	BIN File	402 KB
node.dll	7/27/2016 9:06 PM	Application extens...	11,774 KB
pdf.dll	7/27/2016 9:06 PM	Application extens...	10,012 KB
snapshot_blob.bin	7/27/2016 9:06 PM	BIN File	574 KB
ui_resources_200_percent.pak	7/27/2016 9:06 PM	PAK File	128 KB
vccorlib120.dll	7/27/2016 9:06 PM	Application extens...	349 KB
version	7/27/2016 9:06 PM	File	1 KB
xinput1_3.dll	7/27/2016 9:06 PM	Application extens...	105 KB

ainers

focused_minisky
ubuntu

SEARCH Search for Docker images from Docker Hub

Recommended

 kitematic hello-world-nginx A light-weight nginx container that demonstrates the features of Kitematic ♡ 48 ↻ 354K ... CREATE	 official ghost Ghost is a free and open source blogging platform written in JavaScript ♡ 409 ↻ 2M ... CREATE	 official jenkins Official Jenkins Docker image ♡ 1.8K ↻ 8M ... CREATE
 official rethinkdb RethinkDB is an open-source, document database that makes it easy to build and scale realtime... ♡ 283 ↻ 2M ... CREATE	 kitematic minecraft The Minecraft multiplayer server allows two or more players to play Minecraft together ♡ 60 ↻ 26K ... CREATE	 official solr Solr is the popular, blazing-fast, open source enterprise search platform built on Apache... ♡ 225 ↻ 253K ... CREATE
 official postgres The PostgreSQL object-relational database system provides reliability and data integrity. ♡ 2.4K ↻ 16M ... CREATE	 official ubuntu-upstart Upstart is an event-based replacement for the /sbin/init daemon which starts processes a... ♡ 65 ↻ 217K ... CREATE	 official memcached Free & open source, high-performance, distributed memory object caching system. ♡ 473 ↻ 5M ... CREATE
 official celery Celery is an open source asynchronous task queue/job queue based on distributed... ♡ 137 ↻ 202K ... CREATE	 official mysql MySQL is a widely used, open-source relational database management system (RDBMS). ♡ 2.8K ↻ 26M ... CREATE	 official mongo MongoDB document databases provide high availability and easy scalability. ♡ 2.1K ↻ 30M ... CREATE

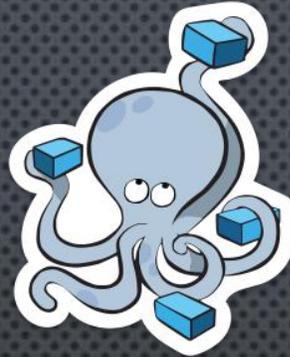
ET SI JE CHANGE QUELQUE CHOSE DANS MON CONTENEUR ?



Versioning :

- Docker inclut des fonctionnalités de type git pour le suivi des versions successives d'un conteneur, l'inspection de la différence entre les versions validant de nouvelles versions, le retour en arrière, etc.
- l'historique inclut également la manière dont un conteneur a été assemblé et par qui, vous obtenez ainsi une traçabilité complète depuis le serveur de production jusqu'au développeur en amont.
- Docker implémente également des téléchargements incrémentiels et des téléchargements, similaires à git pull, de sorte que les nouvelles versions d'un conteneur peuvent être transférées en envoyant uniquement des différences.

DOCKER COMPOSE



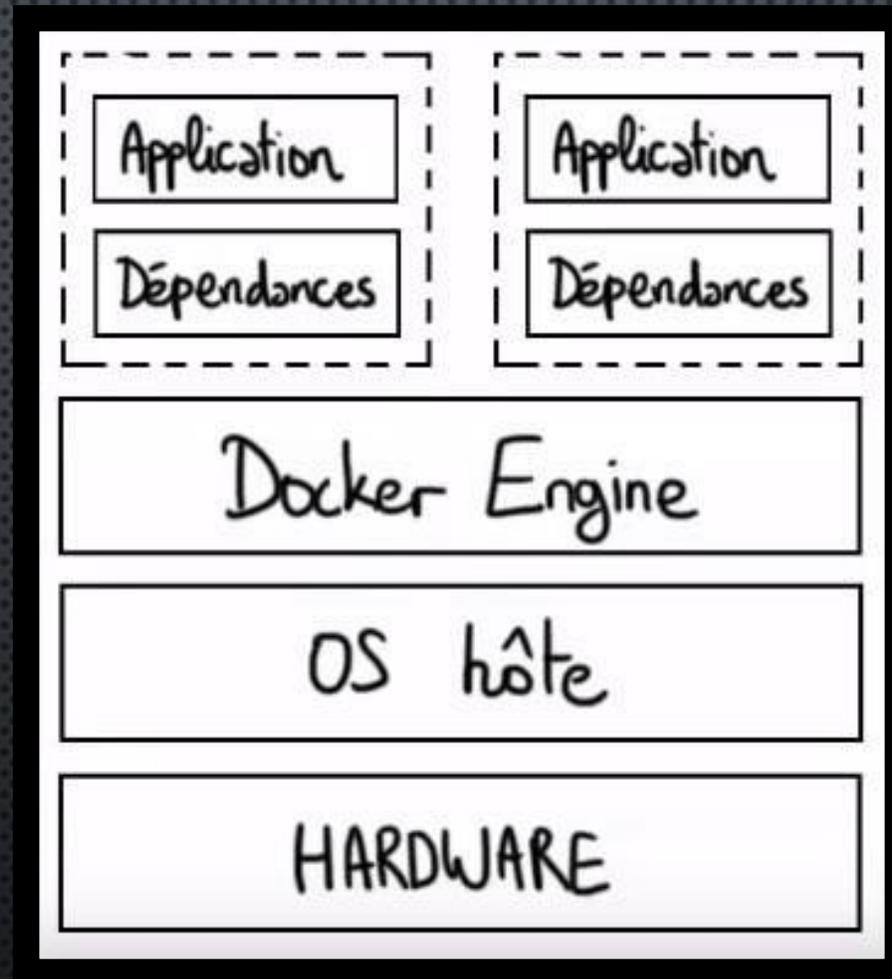
```
EXPLORER
├── OPEN EDITORS 1 UNSAVED
│   ├── docker-compose.yml
│   └── WORDPRESSSITE
│       └── docker-compose.yml
└── docker-compose.yml
    1  version: '2'
    2
    3  services:
    4
    5      wordpress:
    6        image: wordpress
    7        ports:
    8          - 8080:80
    9        environment:
   10          WORDPRESS_DB_PASSWORD: abc123
   11
   12      mysql:
   13        image: mysql:latest
   14        environment:
   15          MYSQL_ROOT_PASSWORD: abc123
   16
   17
```

Exemple : J'ai besoin d'une stack Nginx, PHP FPM, MariaDB et Memcached pour faire des tests. Pour cela nous avons besoin de créer quatre containers (un pour chaque service) et de les lancer ensemble.

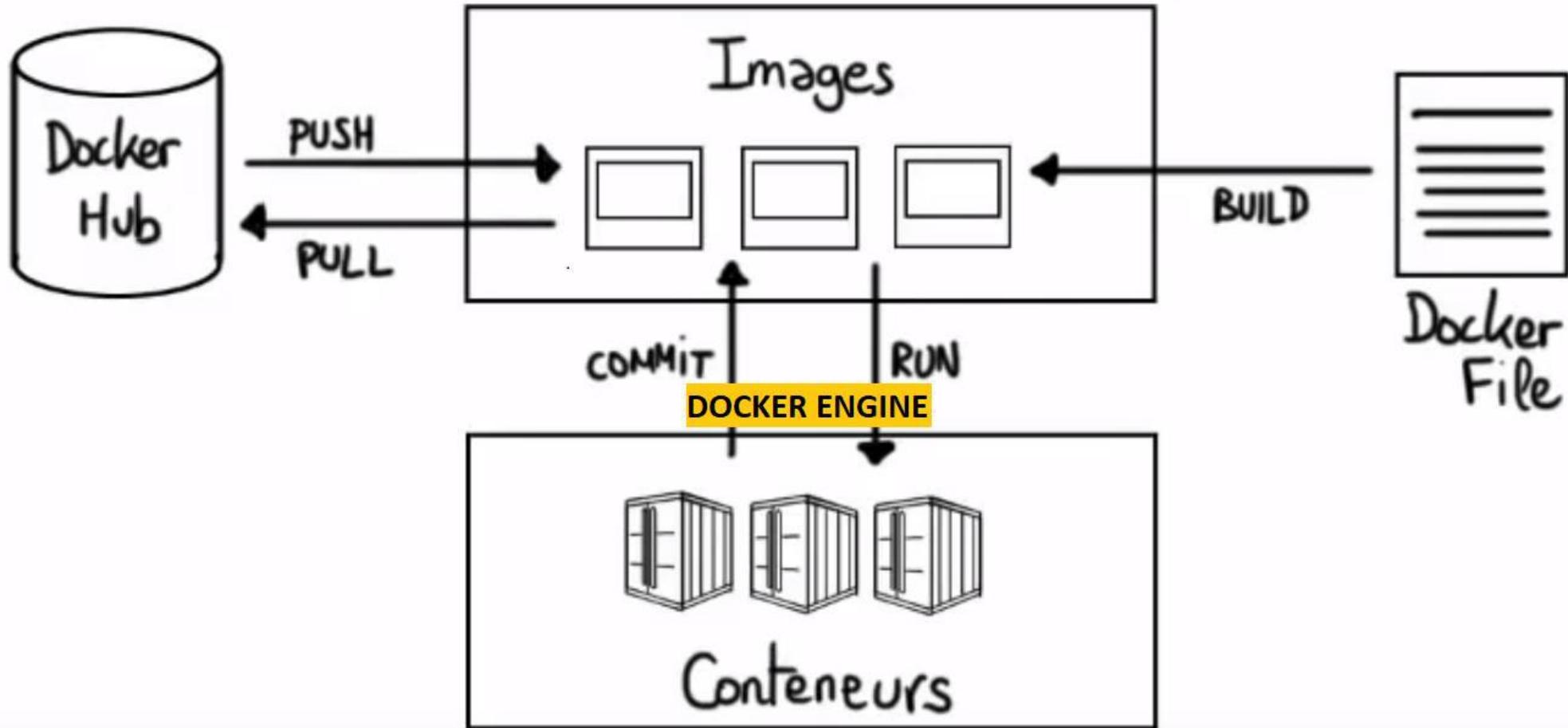
Problème : Néanmoins il faut le faire manuellement et respecter chaque étape. C'est là que docker-compose vous simplifie la vie.

Solution : En se basant sur un fichier docker-compose.yml contenant la configuration de votre stack, docker-compose va créer, lancer, ouvrir les ports, créer les volumes et lier vos containers tout seul.

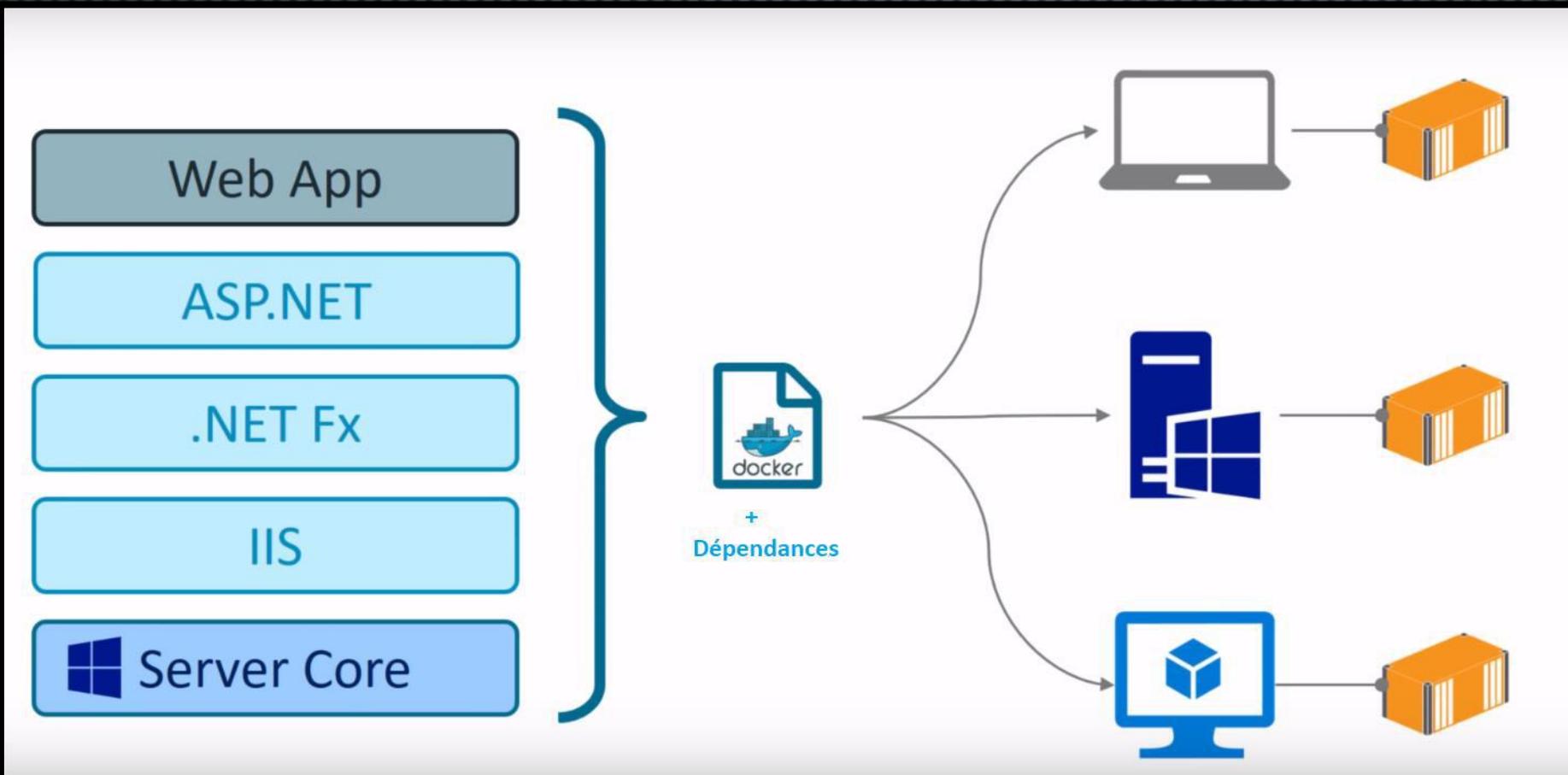
EN BREF



Docker permet de créer des conteneurs qui vont uniquement contenir des applications avec leurs dépendances et de les exécuter de manière virtuellement isolée.



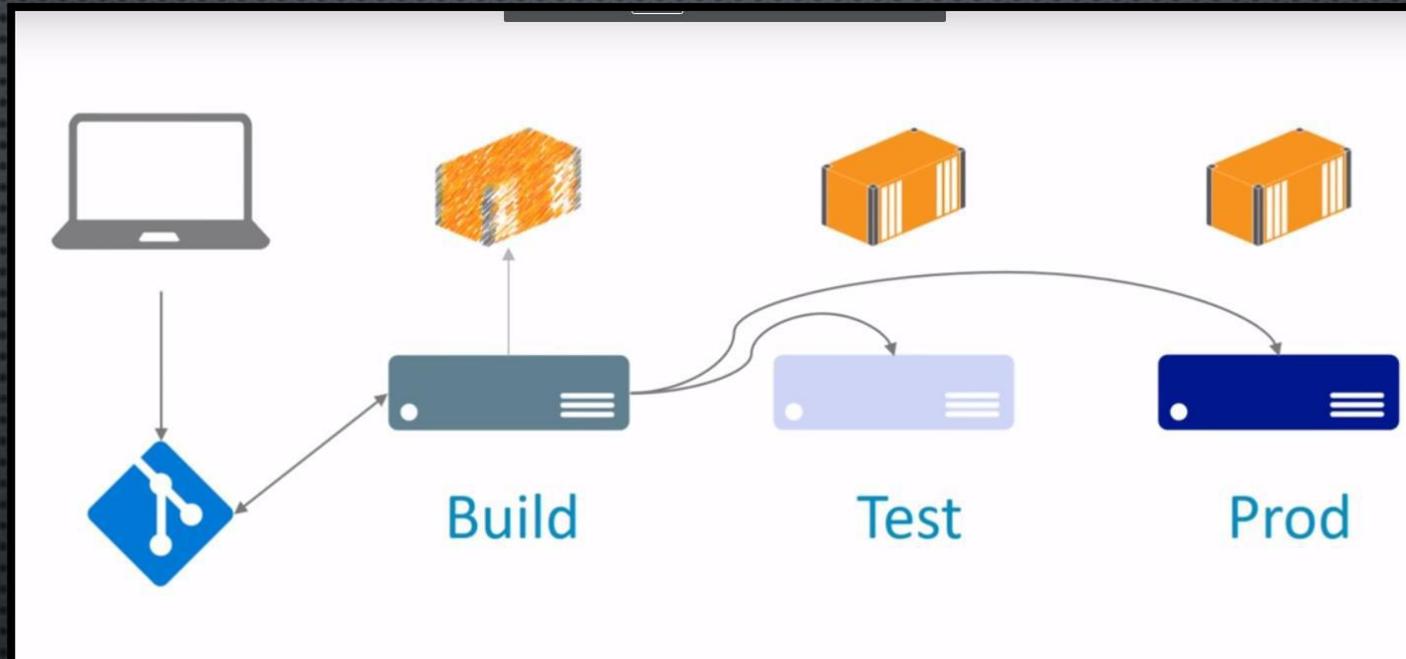
DOCKER ET MICROSOFT



Vous pouvez l'exécuter dans chaque machine sur laquelle docker est installé. Les machines qui lancent cette image n'ont pas besoin d'installer le service IIS ou autre.

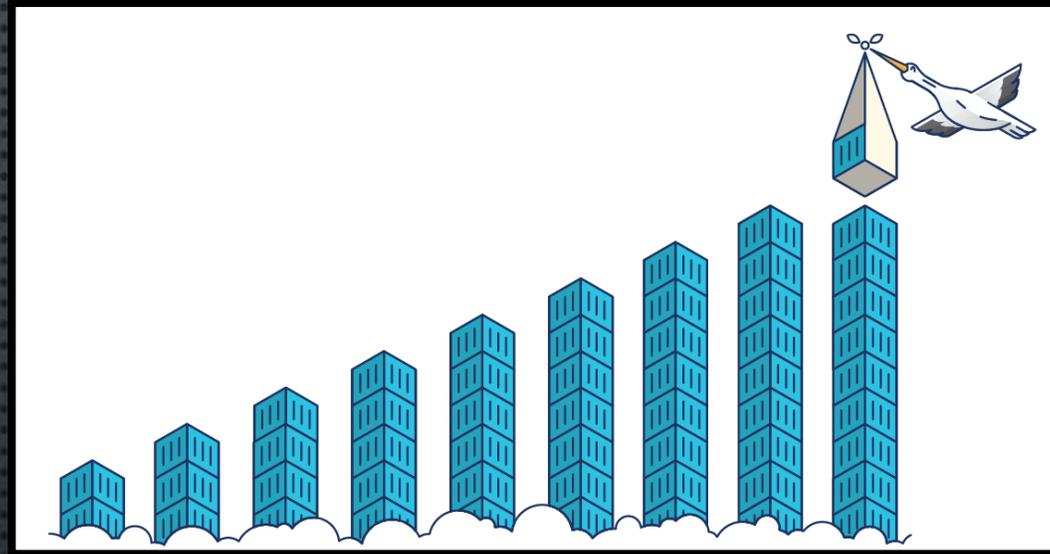
i Docker tourne nativement sur windows 10 dans l'optique d'en faire bénéficier les appareils (PC, tablettes, objets connectés...)

POURQUOI DOCKER EST POPULAIRE ?



- Faciliter et accélérer le déploiement car dans un conteneur sont regroupés toutes les dépendances d'une application.
- Transporter une application de façon fluide entre les différents environnements (depuis l'environnement de test jusqu'à l'environnement de production et inversement)
- **Conclusion** : outil incontournable pour un DevOps !

QUELQUES CHIFFRES

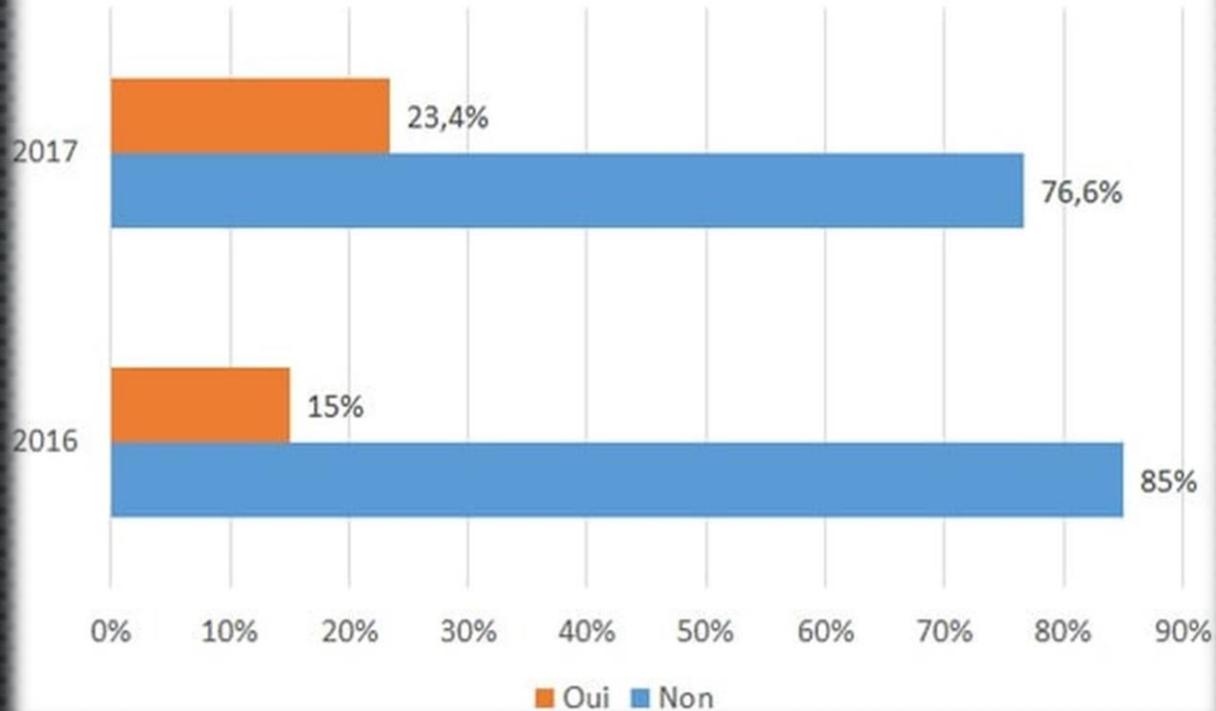


- Une levée de fonds de 40 millions de dollars en 2014.
- La dernière levée de fonds du spécialiste du conteneur remonte à avril 2015, avec 95 millions de dollars à la clé.
- Docker finaliserait une levée de fonds de 75 millions de dollars. La jeune entreprise compte profiter de ce nouvel apport financier pour renforcer son pôle commercial et attirer les grandes entreprises.
- La société américaine aura récolté au total 235 millions de dollars depuis sa création, pour une valorisation estimée à 1,3 milliard de dollars.

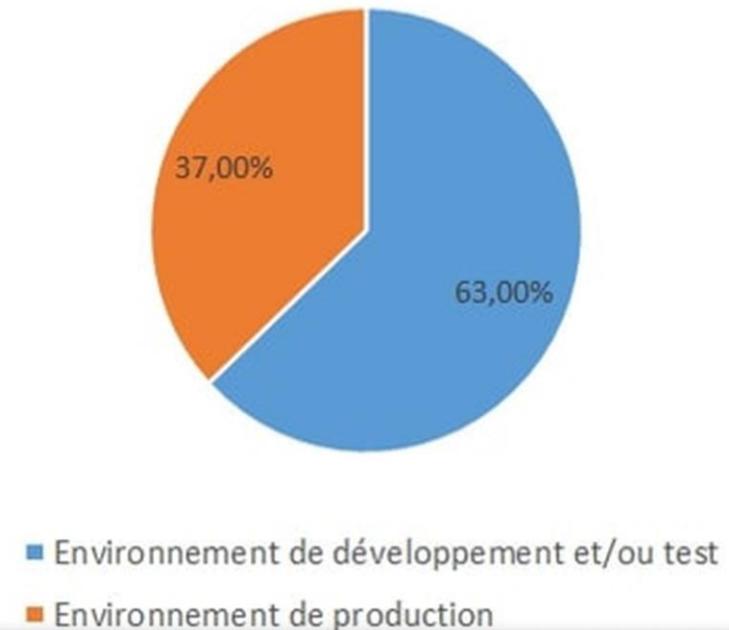
Source : <https://www.bloomberg.com/news/articles/2017-08-09/docker-is-said-to-be-raising-funding-at-1-3-billion-valuation>

DOCKER EN FRANCE

Avez-vous recours aux containers (Docker) ?



Pour quel type d'environnement avez-vous recours aux containers ?



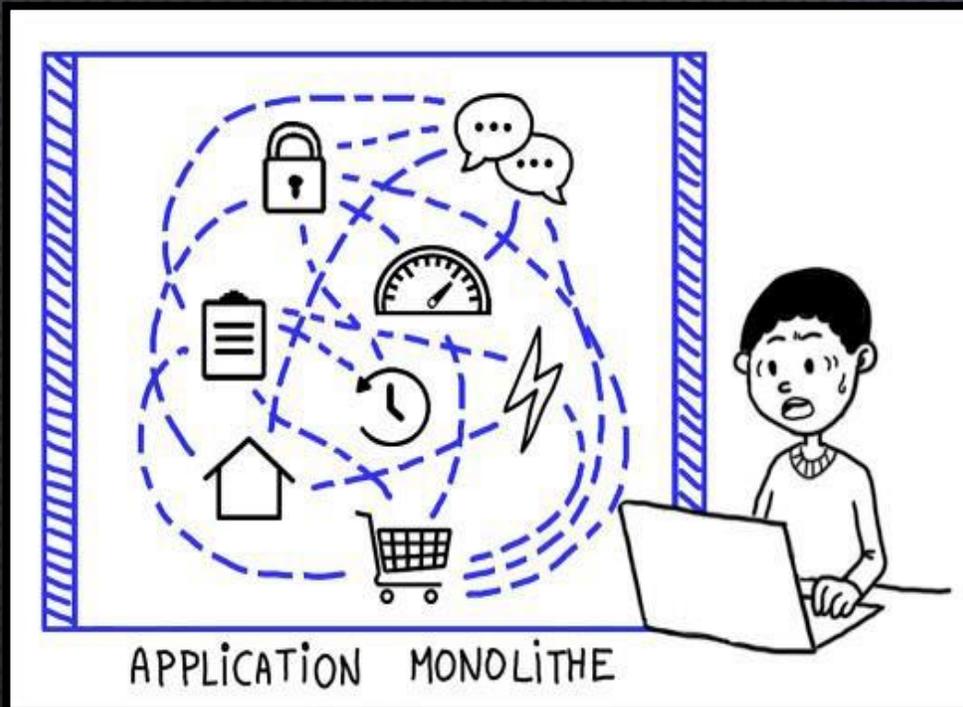
Club Décision DSI / IT Research réalisé entre le 27 septembre et le 17 octobre 2017 auprès de 486 décideurs informatiques (DSI, DOSI, directeurs de production...), tous membres du Club Décision DSI. Ils travaillent pour des organisations (privées ou publiques) de taille moyenne ou grande, de 300 à 50 000 salariés, dans 20 secteurs d'activité.

INTRODUCITON À L'ARCHITECTURE MICROSERVICES



L'architecture microservices a été inventée pour résoudre certaines des difficultés causées par les gros projets utilisant une architecture Monolithe.

ARCHITECTURE MONOLITHE



Plus le code devient complexe plus les interdépendances entre les différentes briques augmentent avec le temps.

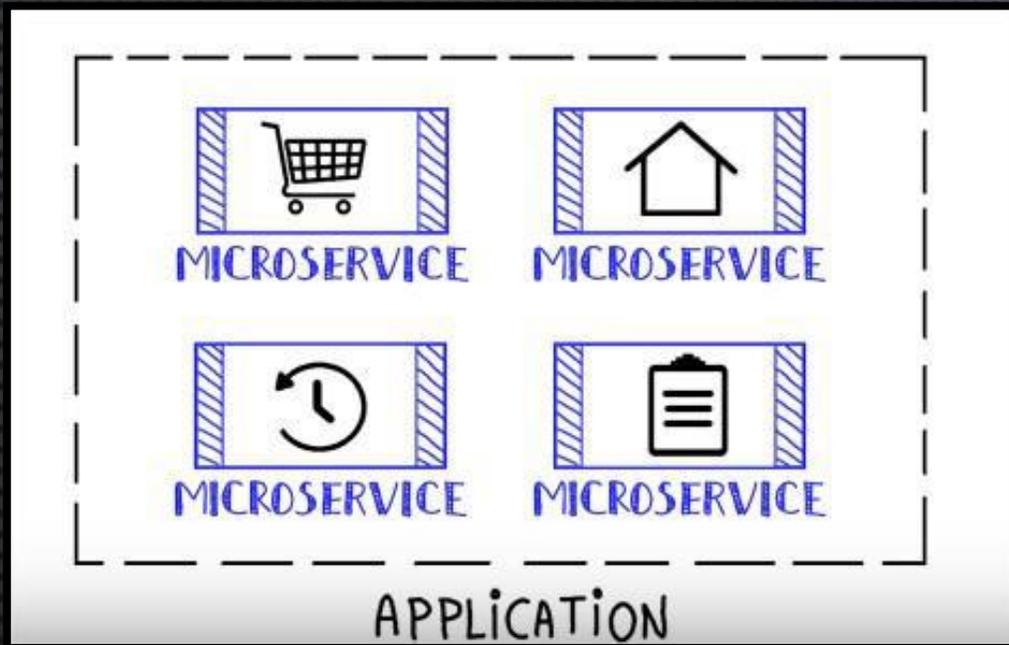
Conséquences :

- Plus on intègre de fonctionnalités, rarement on supprime les anciennes et les fonctionnalités même lorsqu'elles sont obsolète.
- Ça devient très compliqué pour les développeurs de gérer le projet et d'avoir un modèle global du projet.
- L'entreprise ne prend pas de risques pour tester des nouveautés.

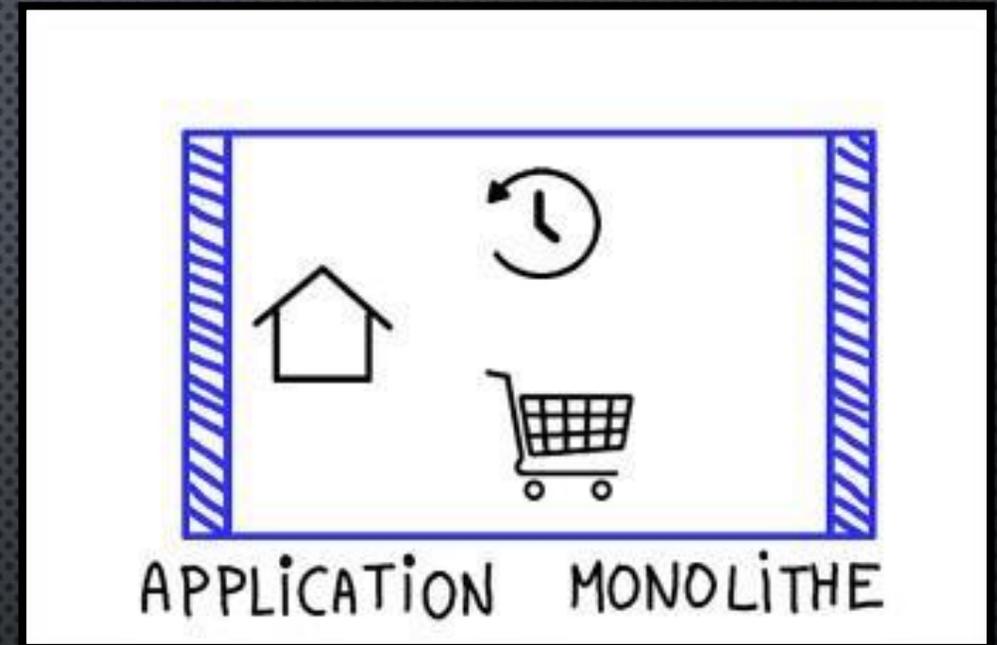
Conclusion :

- Pour une application monolithe on va préférer une stabilité à l'innovation.

L'ARCHITECTURE MICROSERVICES

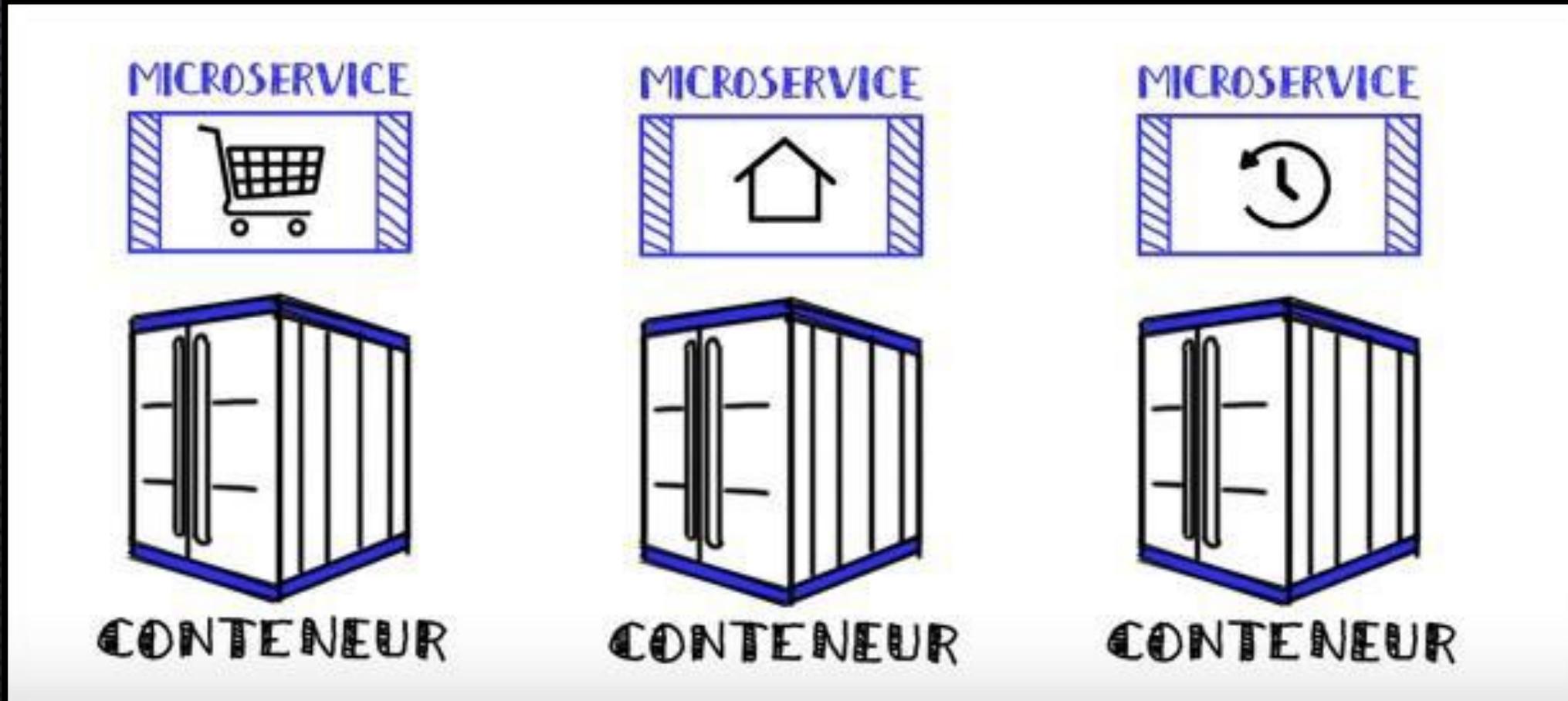


VS



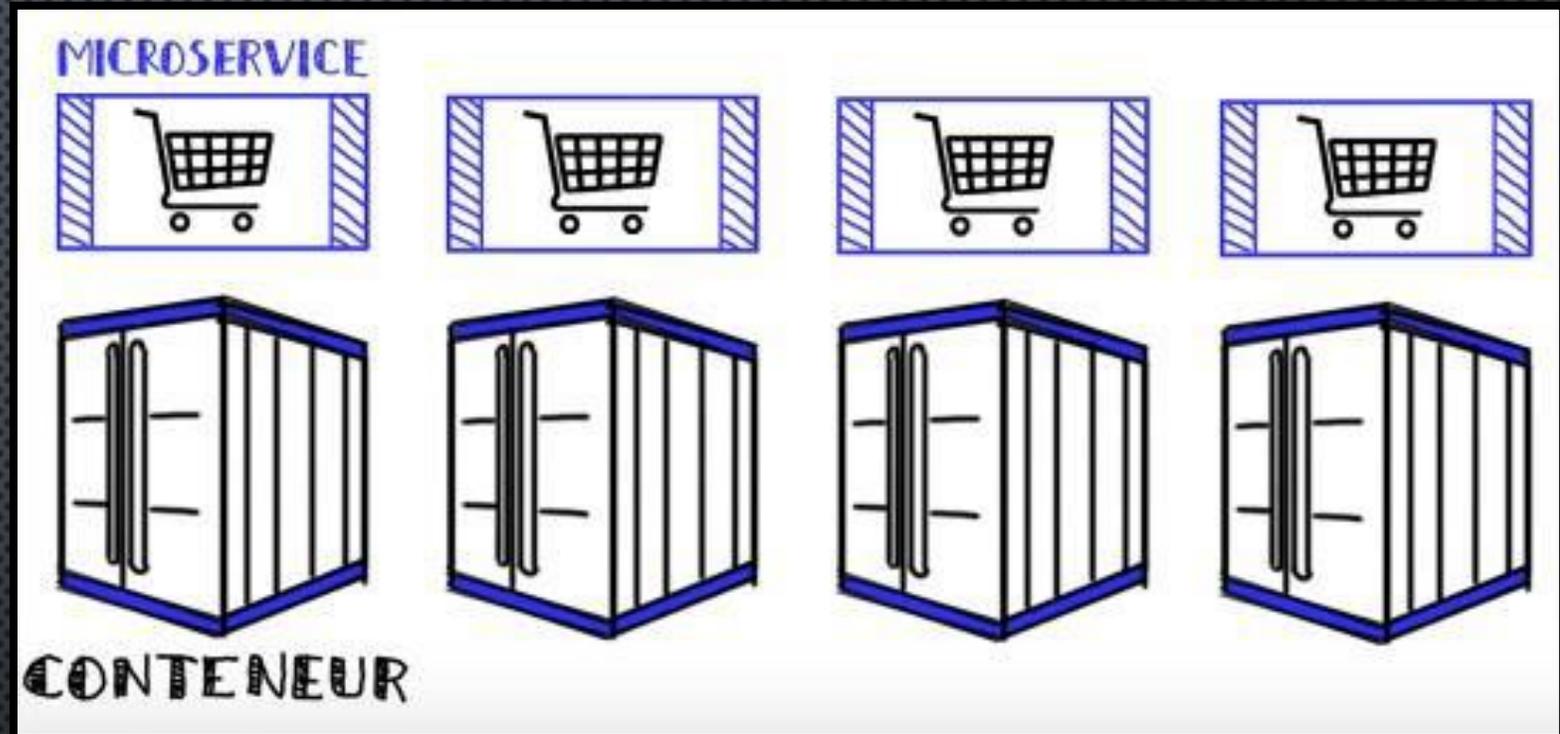
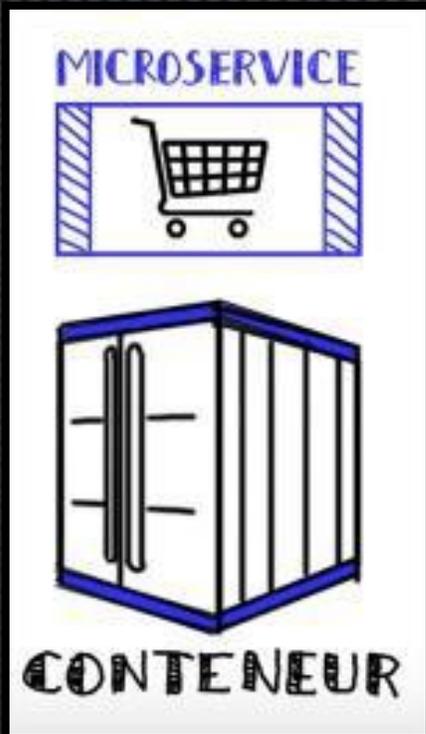
- Application composée de plusieurs micro-services (service métier) spécialisé dans une seule tâche
- Cette architecture micro service répond aux problèmes des applications Monolithes qui consistent à traiter toutes les fonctionnalités d'une seule traite.
- Idéal lorsque vous devez gérer la compatibilité avec une gamme de plates-formes et de périphériques, couvrant le web, le mobile

L'ARCHITECTURE MICROSERVICES ET LES CONTENEURS



Le but est d'avoir un conteneur par microservice. **Et ainsi chaque micro-service à son propre environnement d'exécution.**

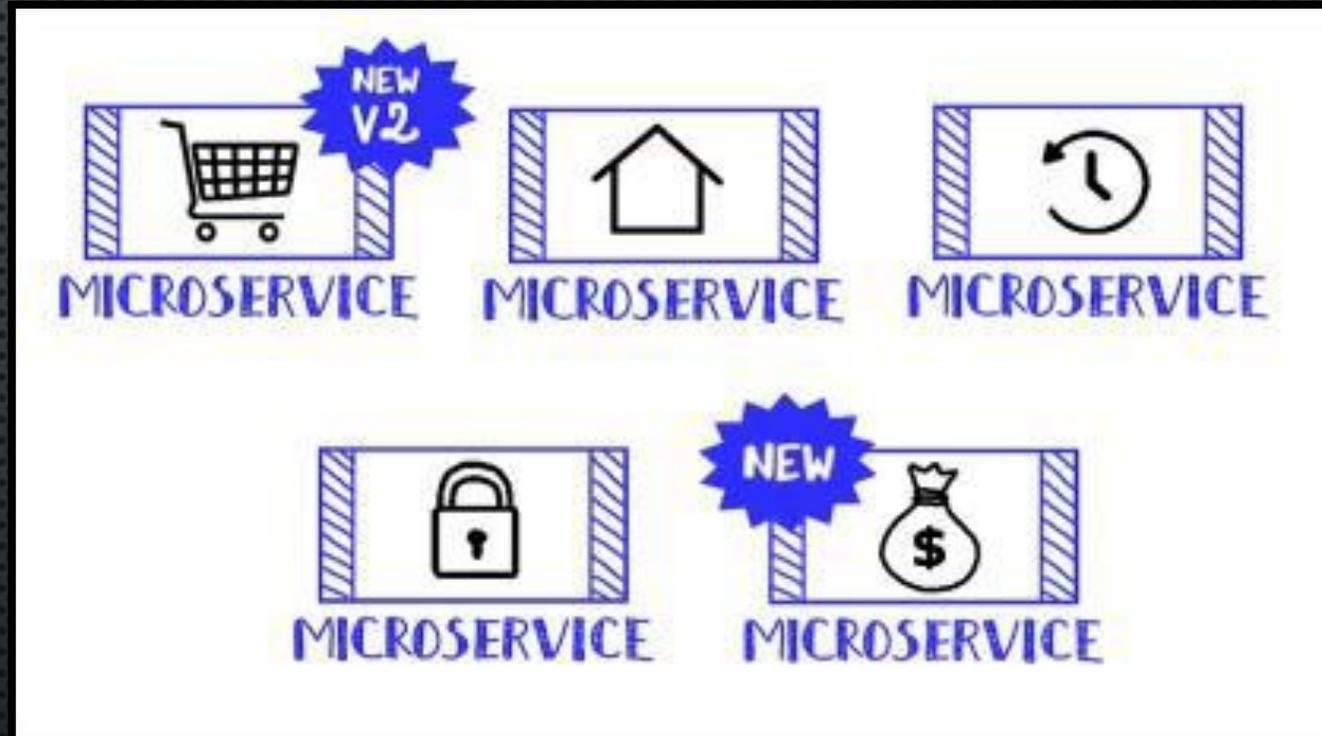
Conclusion : Offre une meilleure isolation et donc moins d'interdépendances



- Lors d'un pic de charge il suffit de créer plusieurs conteneurs pour ne pas lier à ce problème.
- Et inversement si ce service est très peu utilisé on peut limiter le nombre de conteneurs.

Conclusion : Dans une architecture micro-service il suffit juste d'adapter les conteneurs selon les besoins alors que dans une application monolithe il aurait fallu modifier l'ensemble de l'application à cause des interdépendances.

AVANTAGES DU COUPLAGE ENTRE LES MICRO-SERVICES ET CONTENEURS



Conséquences en couplant micro-services avec les conteneurs :

- les microservices deviennent autonome
- Il est par la suite plus simple cibler les différentes parties impactées lors d'une évolution.
- Les mises à jour et les fonctionnalités peuvent être déployé beaucoup plus rapidement et efficacement.



Et concernant la taille de projet humaine, on arrive à avoir des micro-équipes qui sont spécialisés pour un micro-service gérer par des petites équipes et elle est libre de ses propres choix sans impacter les autres équipes.

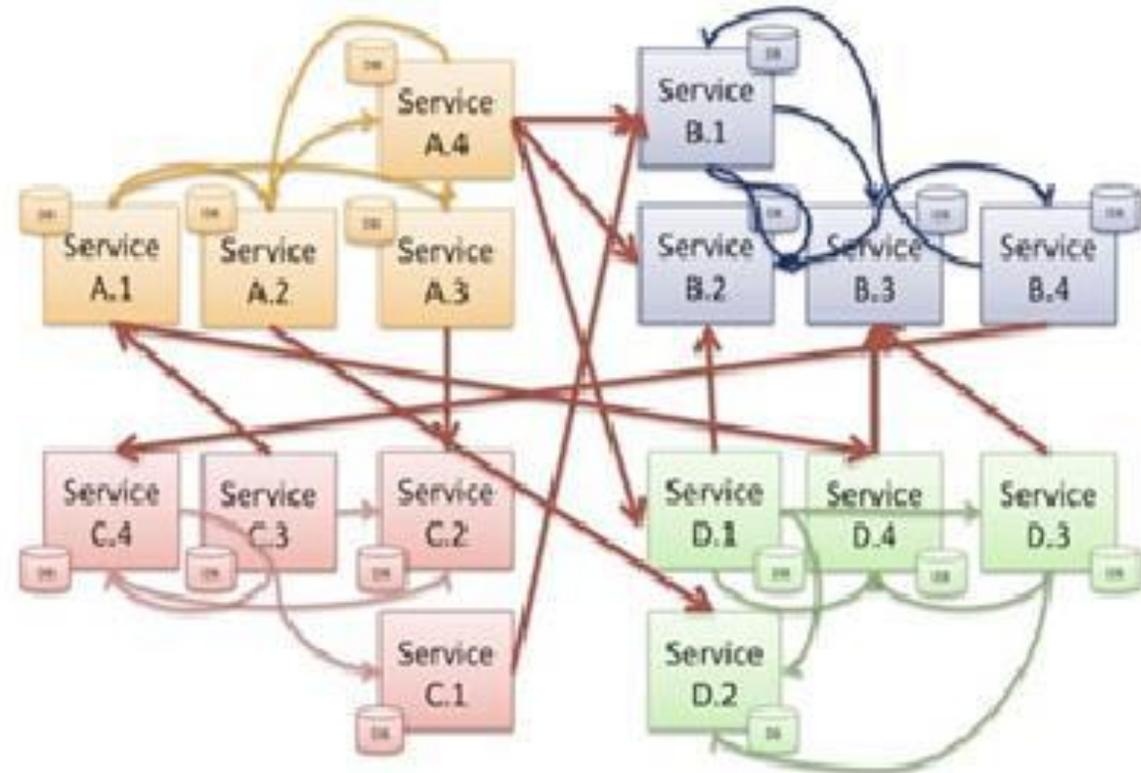
Le but de cette architecture est d'avoir ce qu'on appelle un **"time to market"** rapide qui représente de délais entre la naissance d'une idée et son lancement en production.

INCONVÉNIENTS DE L'ARCHITECTURE MICROSERVICES

- Complex networking

- Overhead

- Databases
- Servers



LES ORCHESTRATEURS DE CONTENEURS

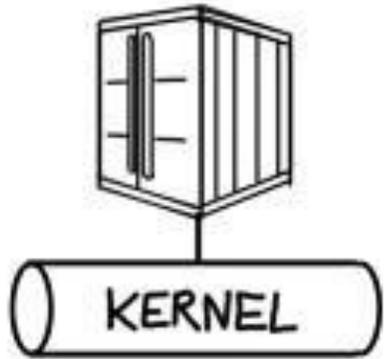
Le container devenant une "commodité", l'enjeu se situe désormais au niveau au-dessus.

Question : Comment gérer le déploiement et le cycle de vie de dizaines ou de centaines de containers ?

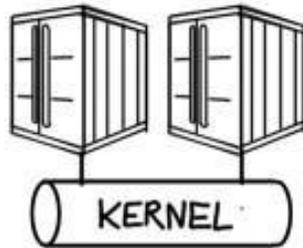
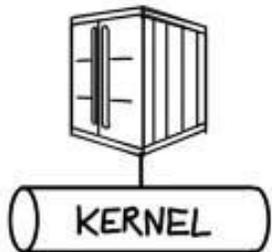
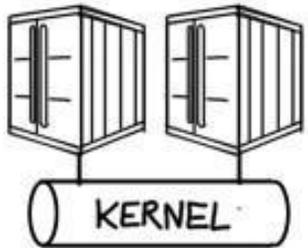
Au final ils supportent tous Docker !



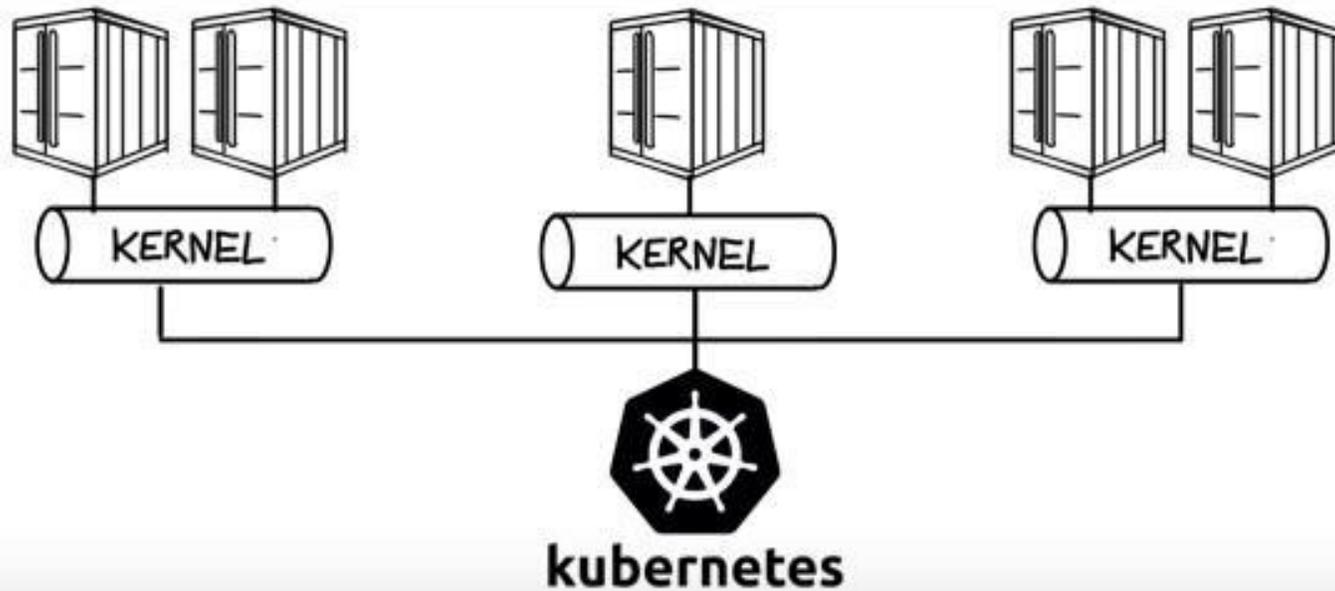
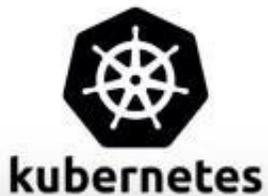
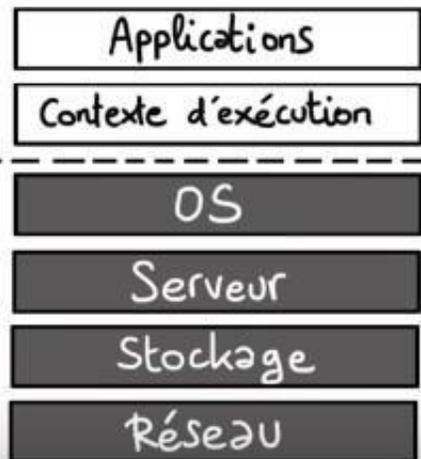
KUBERNETES ET DOCKER



Le conteneur n'a pas conscience de ce qu'il tourne en dehors de ce kernel et donc des autres machines hôtes.



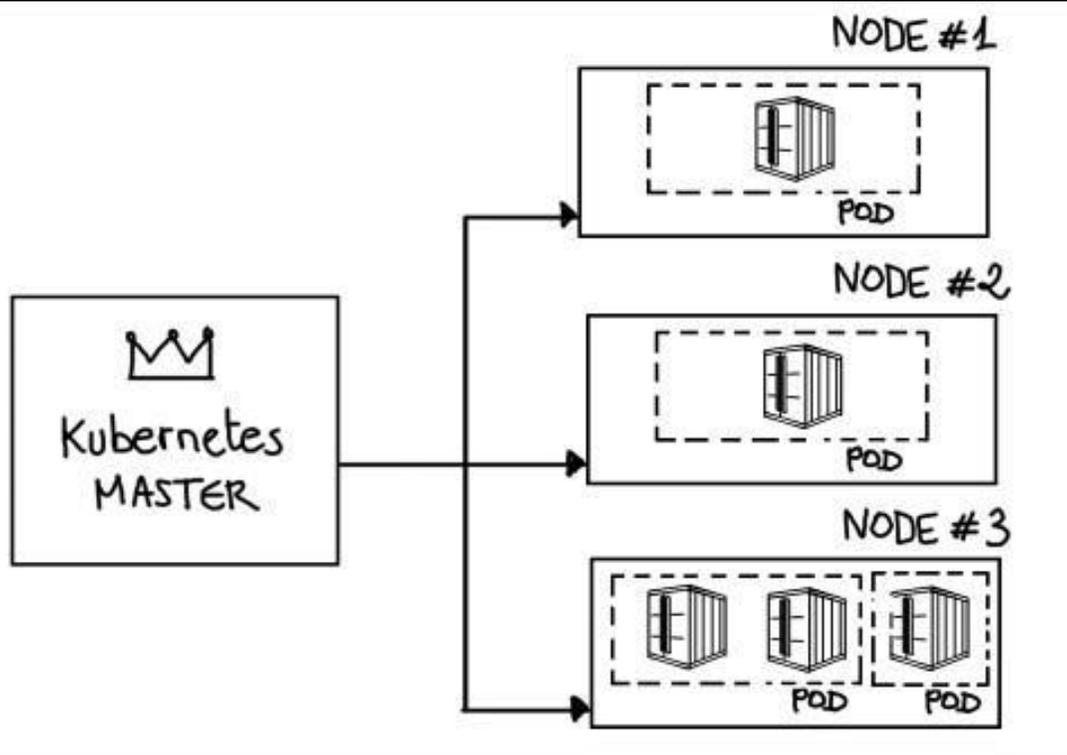
But : Prendre en charge plusieurs kernel et de pouvoir de gérer plusieurs conteneurs sur ces différents hôtes Linux que ça soit une vm, machine physique, cloud.



Conséquences:

- k8s va s'occuper des couches infrastructures.
- Le développeur n'a plus besoin de savoir sur quelle architecture son application va s'exécuter.

EN BREF KUBERNETES ET DOCKER

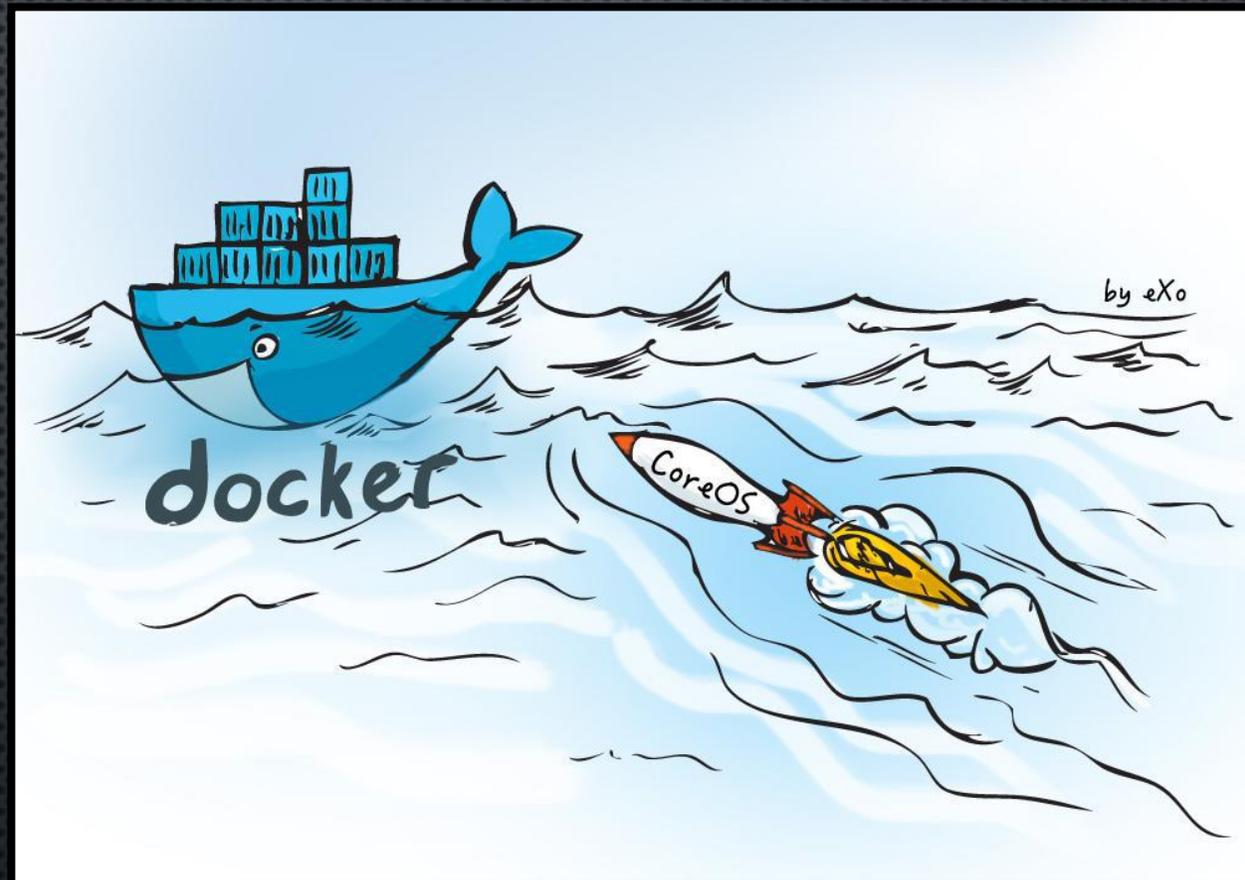


Conclusion :

Les ordres proviennent d'un système automatiser et non plus d'un administrateur qui va assigner manuellement des tâches à tous les nœuds pour chaque conteneur.

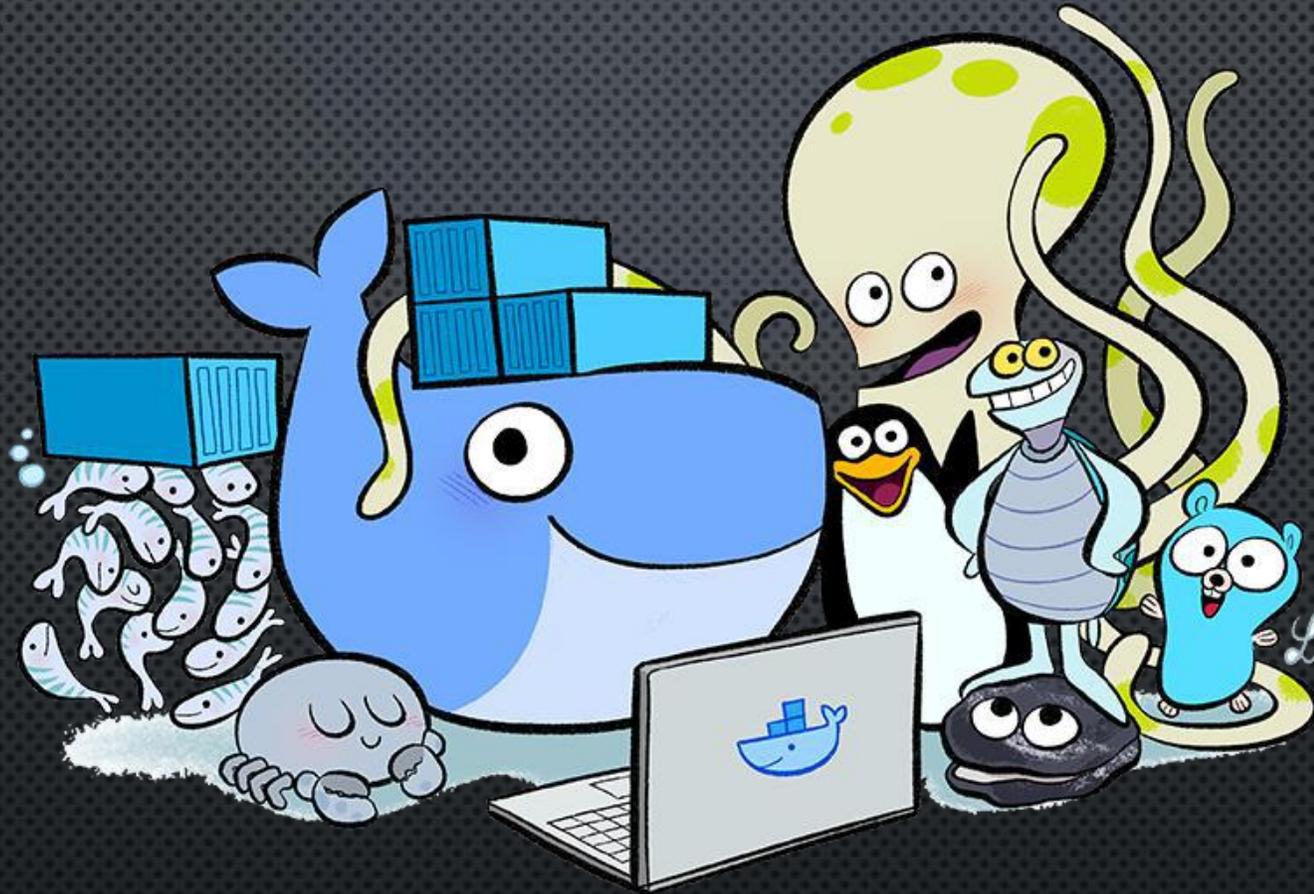
- 1) Le Master reçoit les commandes de la part d'un admin/DevOps et va relayer ces instructions aux nodes
- 2) Le nœud le plus adapté va être choisis (**LoadBalancer**) automatiquement.
- 3) il va ensuite allouer les ressources nécessaires aux pods pour qu'ils effectuent les tâches demandés.
- 4) Le kubelot (service d'un Pod) ordonne à docker de lancer les conteneurs spécifiés
- 5) kubelot collecte les statues de ses conteneurs via docker et rassemble ces informations vers le serveurs master qui est auto réparatrice.

DOCKER VS ROCKET



Les deux offres de conteneurs ont leurs avantages uniques mais en dehors de rocket et de ses avantages de sécurité décroissants sur Docker et docker plus axé sur la simplicité, les offres des deux fournisseurs sont pour la plupart complémentaires.

CONCLUSION



- Légèreté des conteneurs
- Rapidité et facilité de déploiement des applications
- Orientation cloud et micro-services