

DevOps

Introduction

Antoine Balliet

antoine.balliet@dauphine.psl.eu

Avant de commencer

- Matériel requis : un ordinateur par binôme
- Expérience en DevOps parmi les étudiants ?
- Expérience en dev, outils ?

Présentation

Organisation du cours

- 18h au total : 5h de cours / 13h de TP
 - dont 1 TP en autonomie évalué (2h30)
- Slides de cours disponibles sur sharepoint (?)

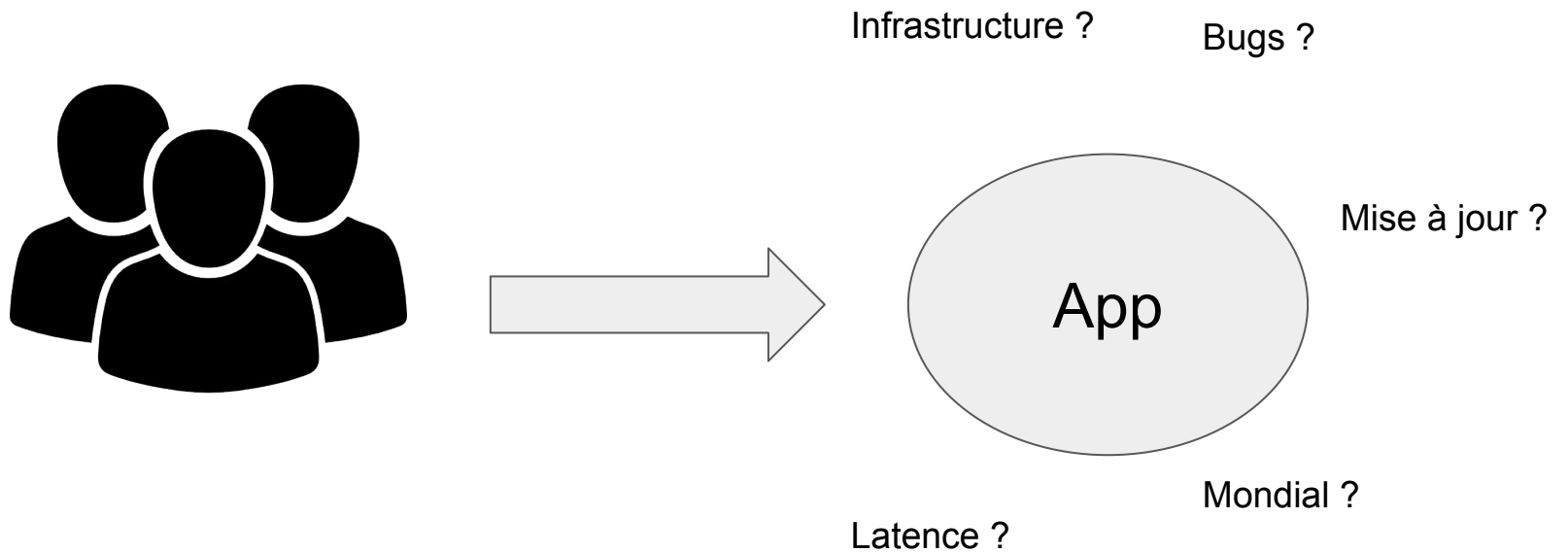
Evaluation

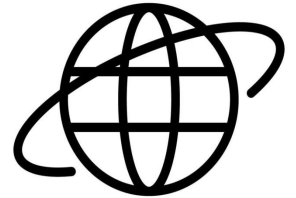
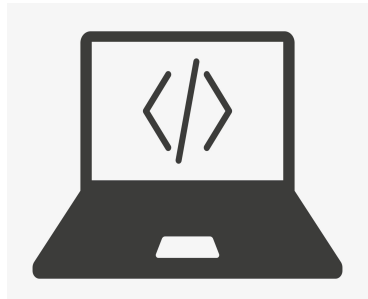
- Note de TP (50%)
 - ▶ Note du projet
 - ▶ + prise en compte du travail en TP
- Devoir sur table (50%) : 6 Novembre

Prérequis

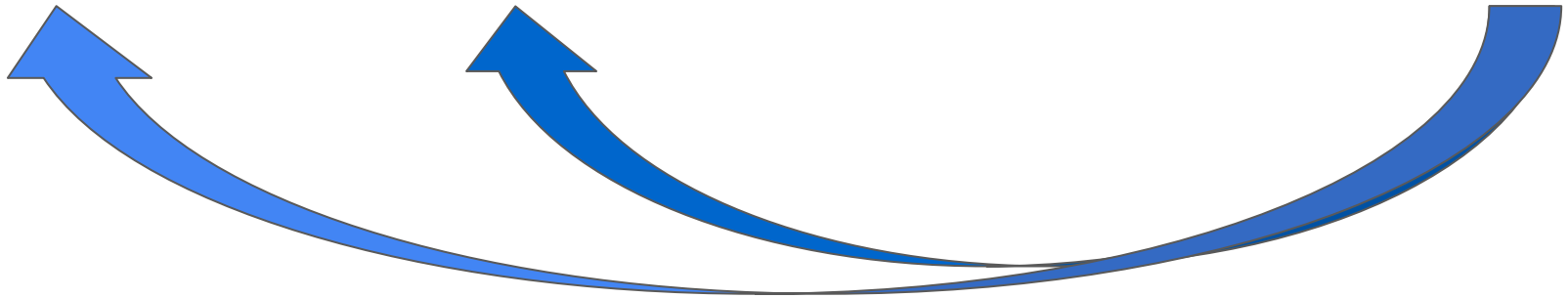
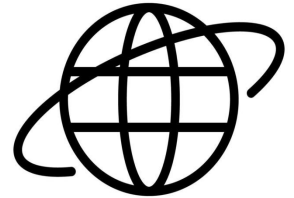
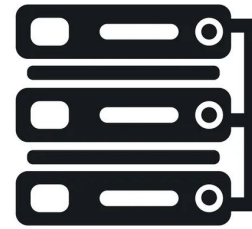
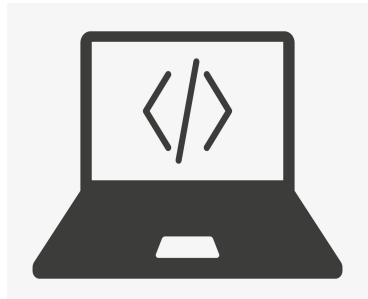
- Pas de développement applicatif en soi mais connaissance shell / bash requis
- 1 ordinateur pour 2
- Curiosité : n'hésitez pas à chercher sur internet :)

Enjeux majeur du secteur : déploiement en production

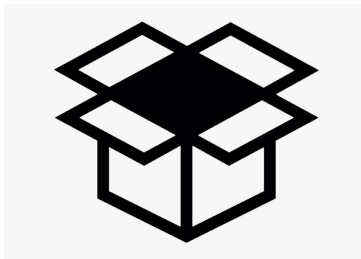




- Mesure et amélioration continue des performances
- Nouvelles fonctionnalités
- Vitesse d'exécution



Processus de livraison d'une application



- Installer les dépendances
- Configurer Firewall
- Déployer l'application
- Vérifier son comportement
- Adapter les ressources à la charge
- ...

Pourquoi ce cours ?

Initier aux **outils** et **pratiques** modernes pour le déploiement et la gestion des applications à travers la culture DevOps

Permettant de démocratiser notre usage et créer une confiance dans le logiciel

“\$5,600 per minute to nearly \$9,000 per minute for average”

“For eCommerce giant Amazon, whose entire business model relies on uptime, estimated costs are around \$13.22 million per hour.”

<https://www.atlassian.com/incident-management/kpis/cost-of-downtime>

● devops
Search term

+ Compare

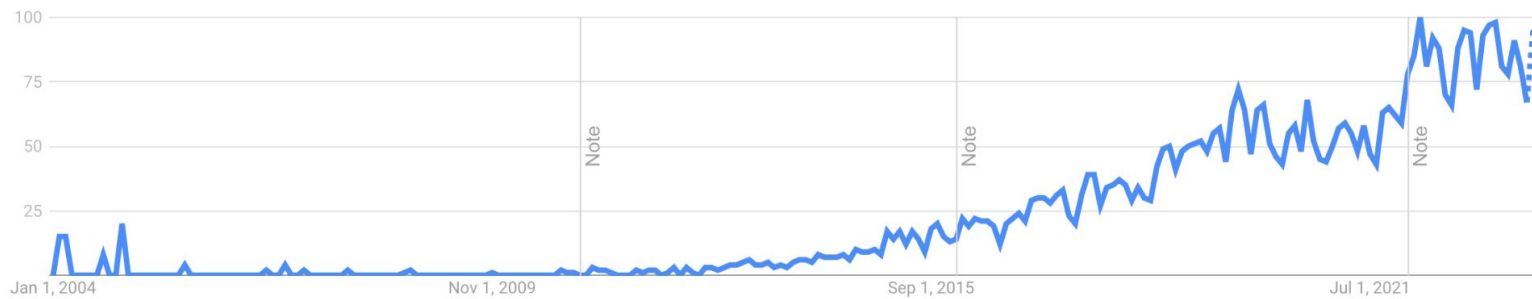
France ▼

2004 - present ▼

All categories ▼

Web Search ▼

Interest over time ?



<https://trends.google.com/trends/explore?date=all&geo=FR&q=devops&hl=en>

● devops
Search term

+ Compare

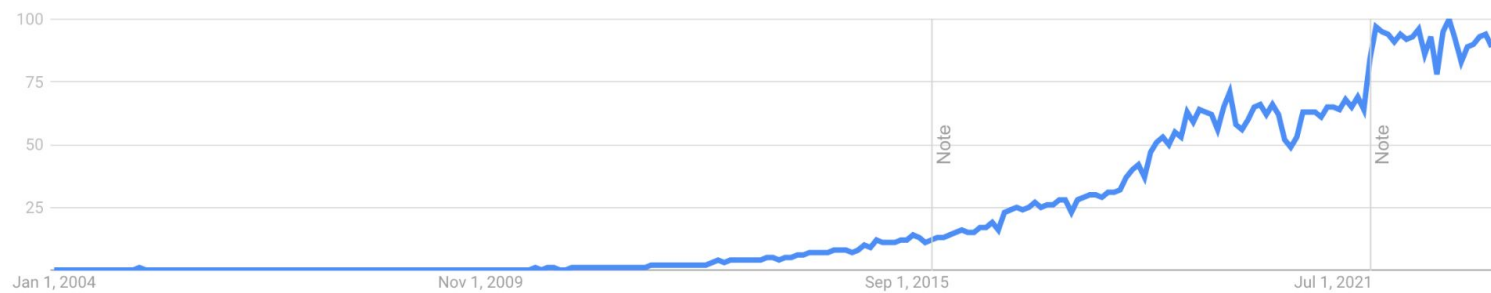
Worldwide ▼

2004 - present ▼

All categories ▼

Web Search ▼

Interest over time ⓘ

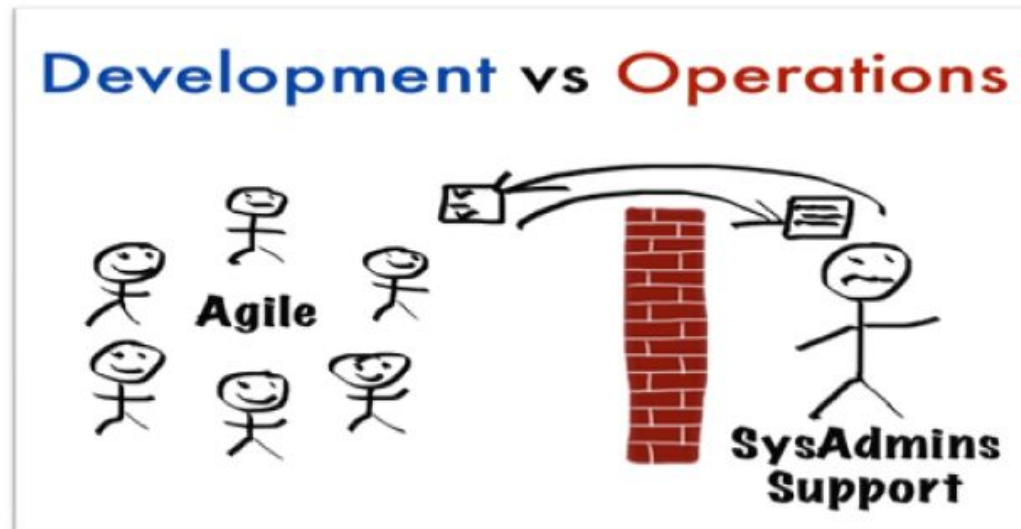


<https://trends.google.com/trends/explore?date=all&geo=FR&q=devops&hl=en>

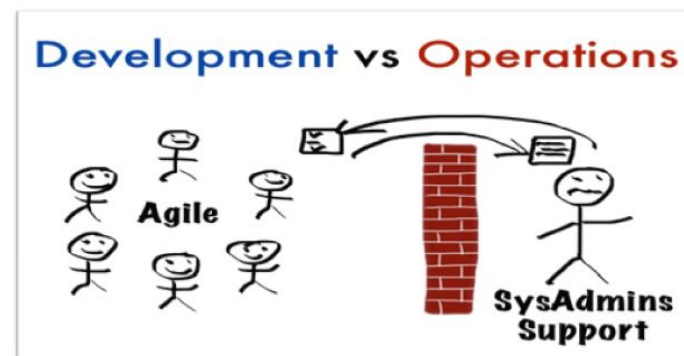
Avant le “DevOps”

2000 - 2007:

- Applications se complexifient (millions de ligne)
- Investissement propre en infrastructure
- Dépendance croissante à du code existant externe



Relation entre Dev et Ops



Dev: équipes de développeurs logiciels

Ops: en charge de la mise en production des produits

Antagonisme fort:

- **Dev:** Modifications aux moindres coûts, le plus vite possible
- **Ops:** Stabilité du système, qualité

L'automatisation est au cœur de l'approche DevOps

Approche DevOps

“You build it, you run it” (2006) - Werner Vogels
(CTO & VP @Amazon)

“The most powerful tool we have as developers is automation”
Scott Hanselman (VP @Microsoft)

“Deployment celebrations should be about the value of the new features, not joyful relief that nothing went horribly wrong”
Rebecca Parsons (CTO @Thoughtworks)

Notions abordées du cours

Développement collaboratif : CI/CD, versioning, git

Virtualisation, conteneurisation et orchestration : docker, kubernetes

Cloud providers / hyperscalers : concepts du cloud, principaux services

Infrastructure as Code : cas d'usage, terraform

DevOps

Ensemble de techniques et d'outils facilitant le passage du développement à la production.

Bien plus que ça:

- Modèle de fonctionnement de l'entreprise
 - ▶ Impliquant tous les maillons de la chaîne (RHs, finances, etc.)
- Modèle d'interactions entre les équipes
 - Intégration du retour sur expérience
 - Une “culture”

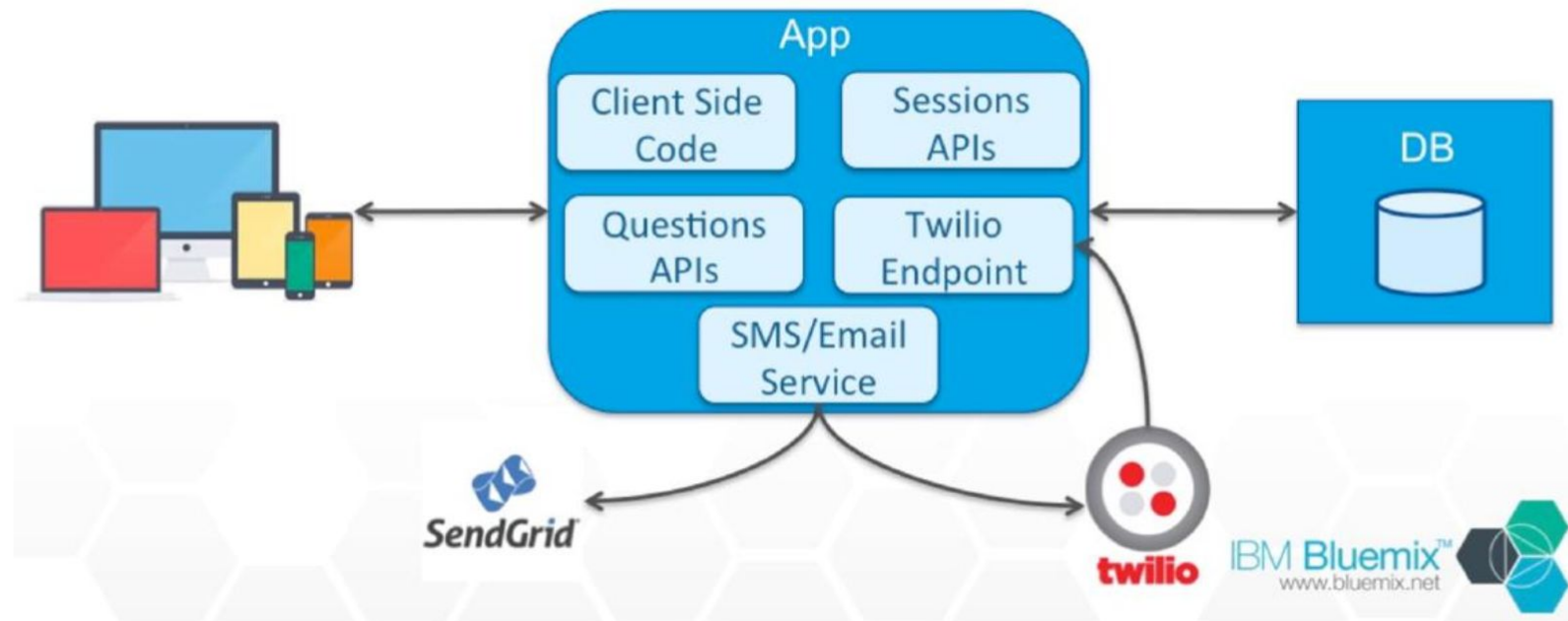
DevOps

Cours 1 : Docker

Antoine Balliet

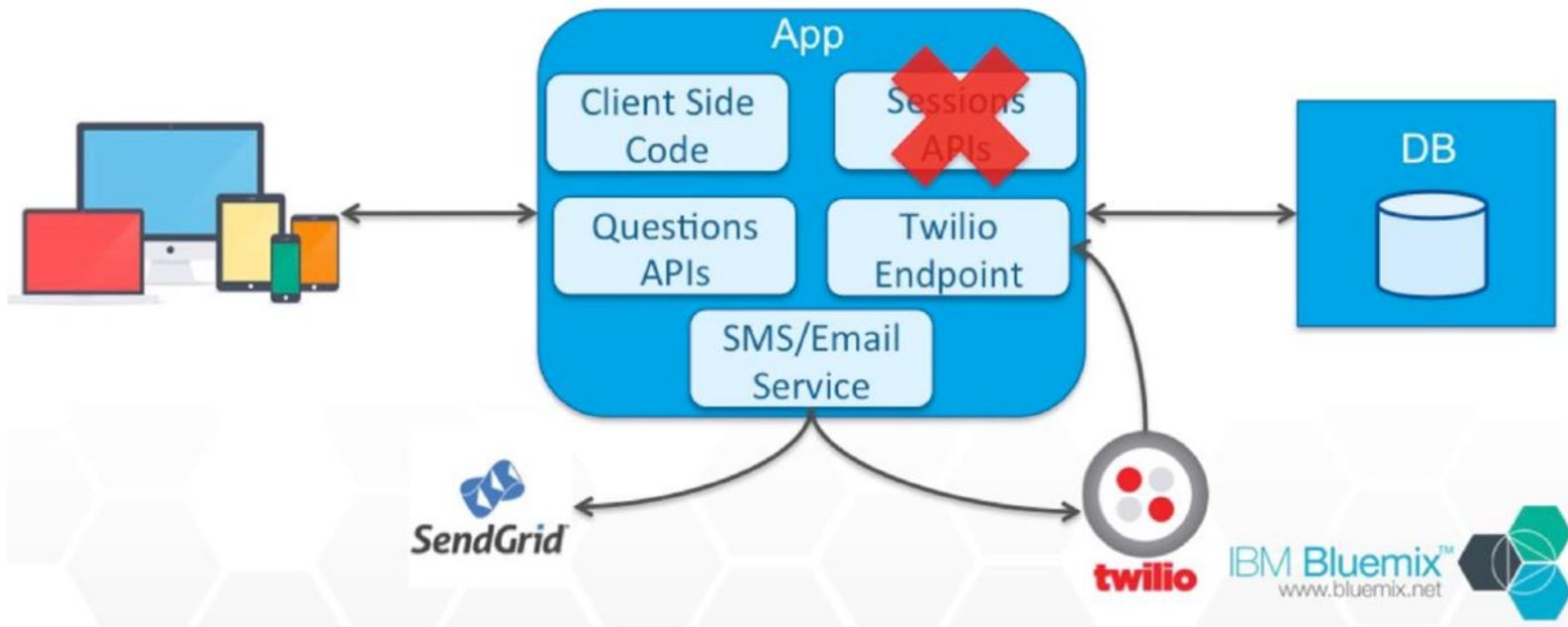
antoine.balliet@dauphine.psl.eu

Isolation des applications



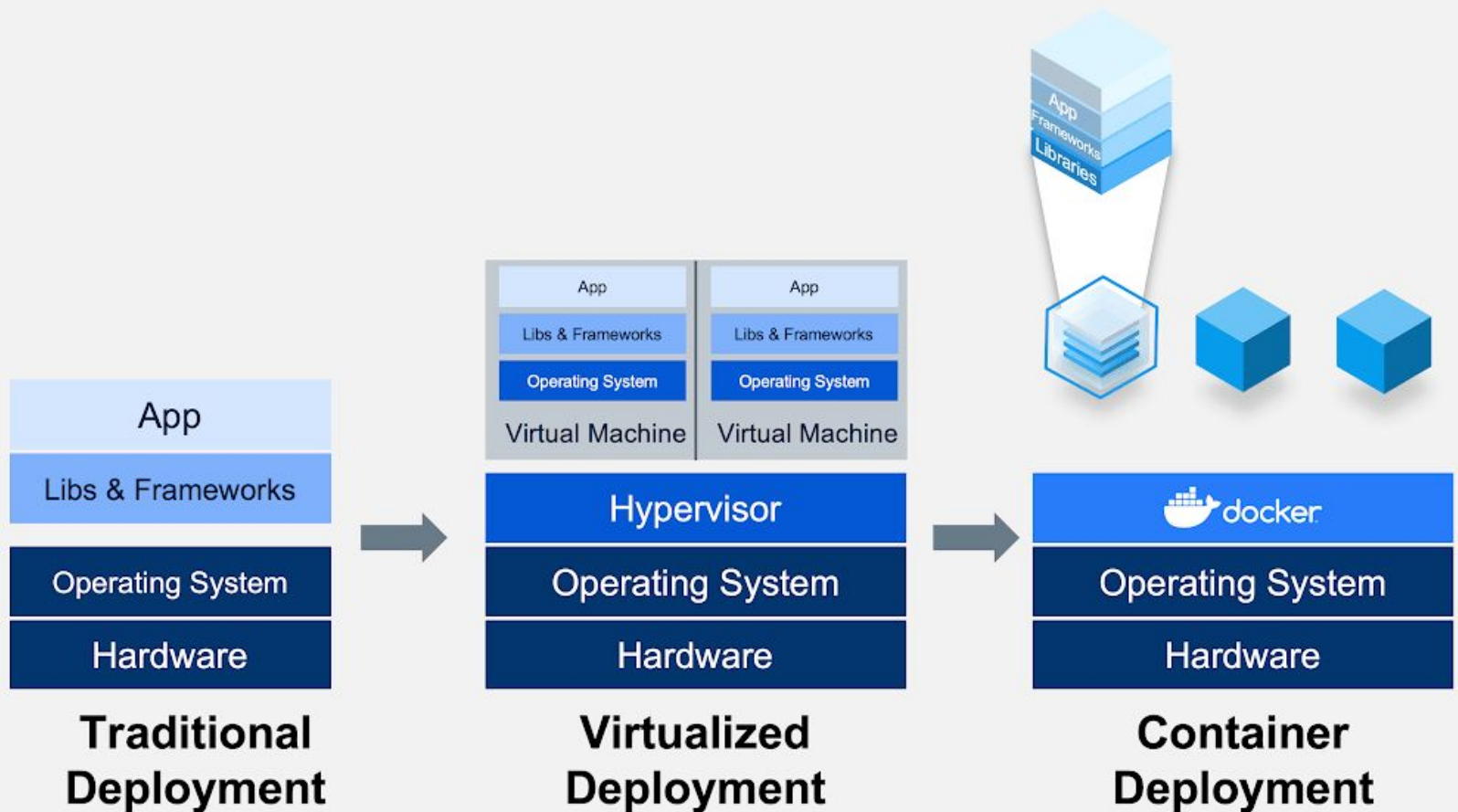
Isolation des applications

Failure in Monolith !!



[illegible]

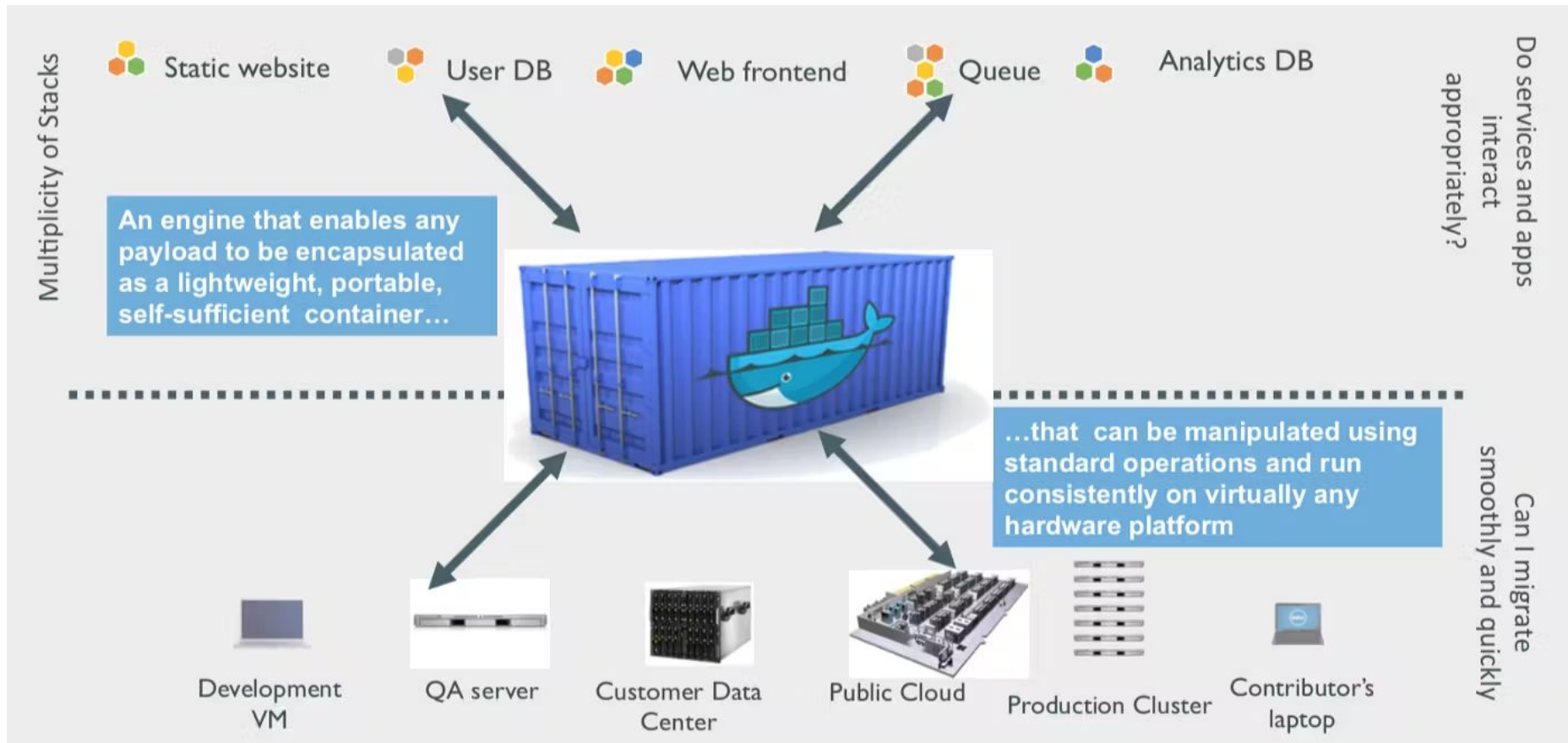
Les déploiements



Conteneurisation



Conteneurisation

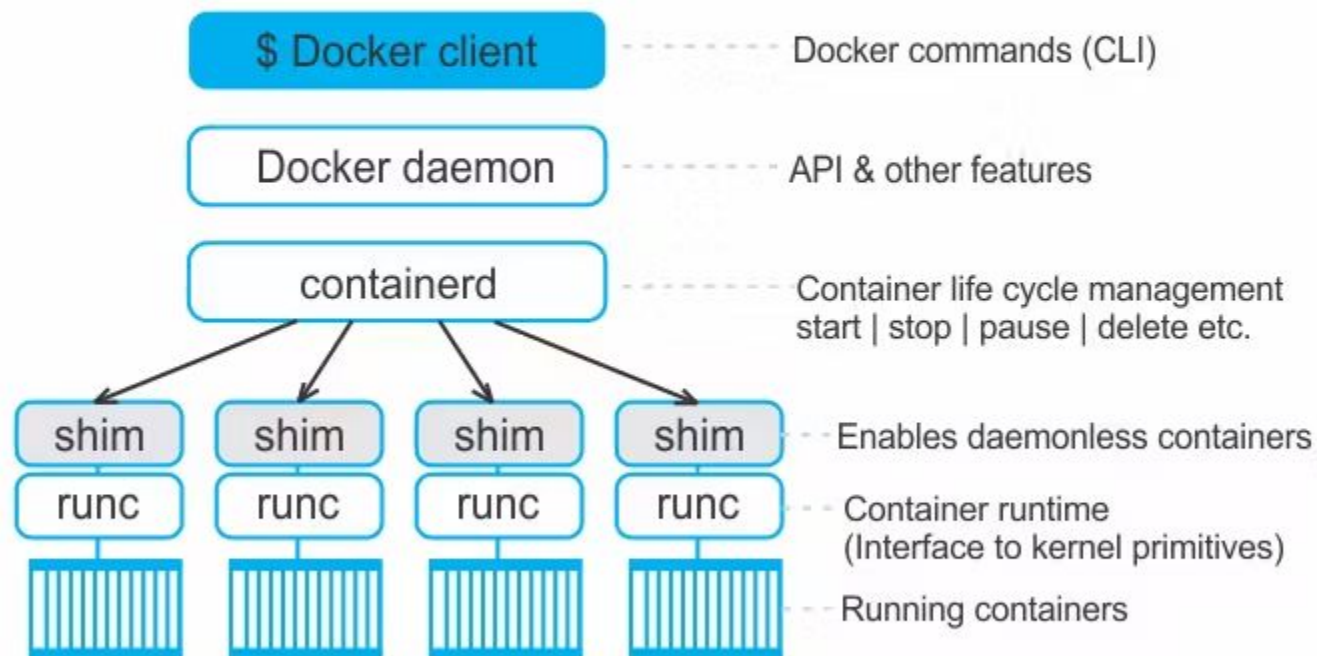


Docker



- **open source** (Apache 2.0)
- crée par la startup DotCloud (renommée depuis Docker) fondée par deux français anciens de l'Epitech.

Docker



Docker

- Uniformiser les environnements : 10 devs travaillent sur multiples OS (Ubuntu, Windows, MacOS, ...)
- Création des environnements locaux
- CI/CD: Création des espaces isolés pour valider les chaînes de tests/déploiements

Docker

- Stateful vs Stateless
- Immutabilité: un conteneur n'a pas pour vocation de stocker des données.
- Définition des ressources

“Application instances should be treated more as cattle”.
Twelve factor apps.

Pour aller plus loin

- <https://docs.docker.com/get-started/>
- <https://12factor.net/> : twelve factor apps. Principes généraux d'architecture pour les applications distribuées