



# Ops

DSIA-5102A - ESIEE Paris

[nicolas.vo@edu.esiee.fr](mailto:nicolas.vo@edu.esiee.fr)  
[nicolasvo95@gmail.com](mailto:nicolasvo95@gmail.com)

# Problématique

Quelles technologies et méthodologies de développement et de déploiement sont utilisées de nos jours?

# Table des matières

Evolution du déploiement

Evolution du développement

Infrastructures mutable et immuable

Scaling vertical et horizontal

Démo  

# Déploiement

# Evolution du déploiement

## Bare-metal

- Machine physique
- Pas de partage des ressources; pas de concurrence
- Meilleure sécurité
- Pas de partage des ressources; gâchis
- Coûts élevés
- Temps de mise en service
- Sensibilité aux pannes

# Evolution du déploiement

Machine virtuelle

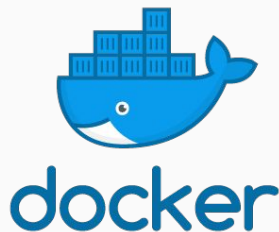
- Hyperviseur, VMM
- Meilleure utilisation des ressources
- Configuration flexible
- Résilience aux pannes
- Coûts assez élevés
- Besoins en ressources des VM
- Temps de mise en marche



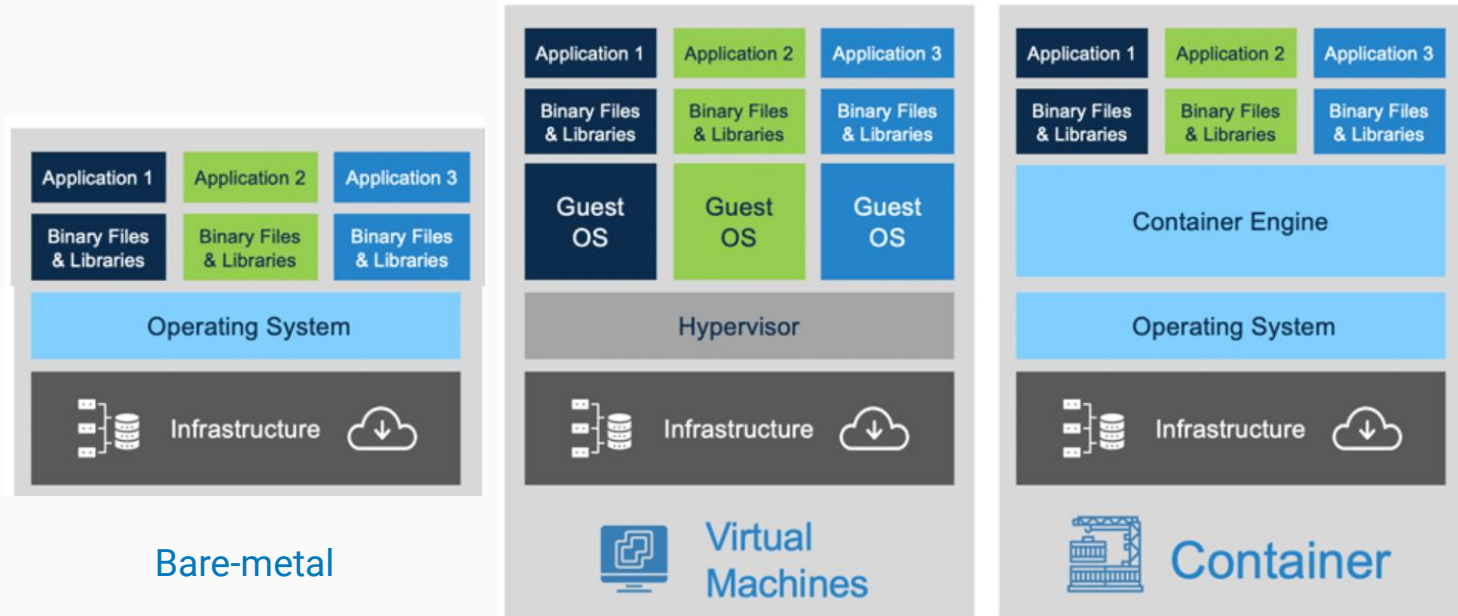
# Evolution du déploiement

## Conteneur

- Container runtime
- Environnement isolé, self-contained app
- Léger et peu cher en ressource
- Rapide à lancer et à jeter
- Tourne partout
- Rapide avancée technologique
- Difficulté d'implémentation



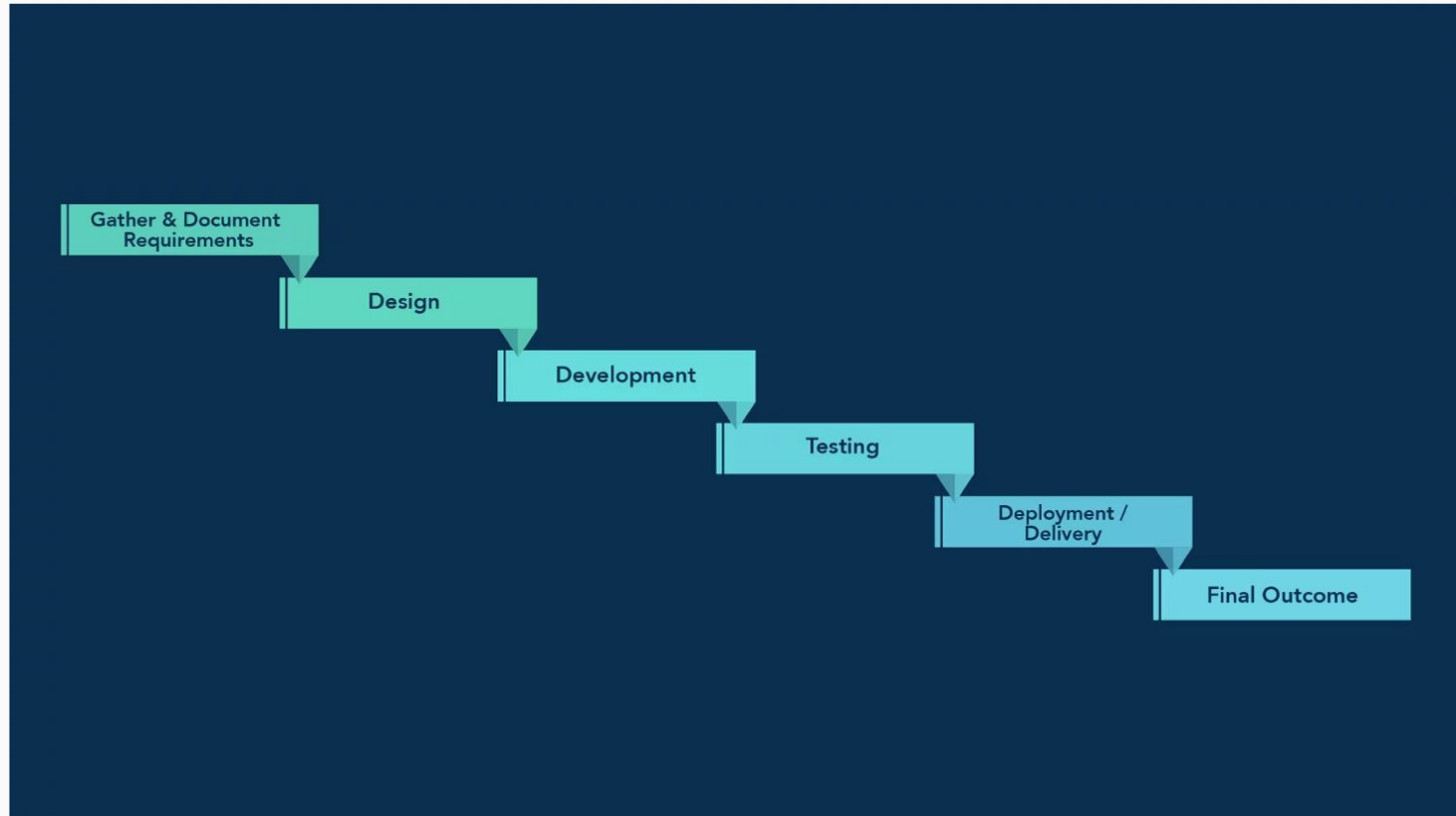
# Comparaison des trois types de déploiements



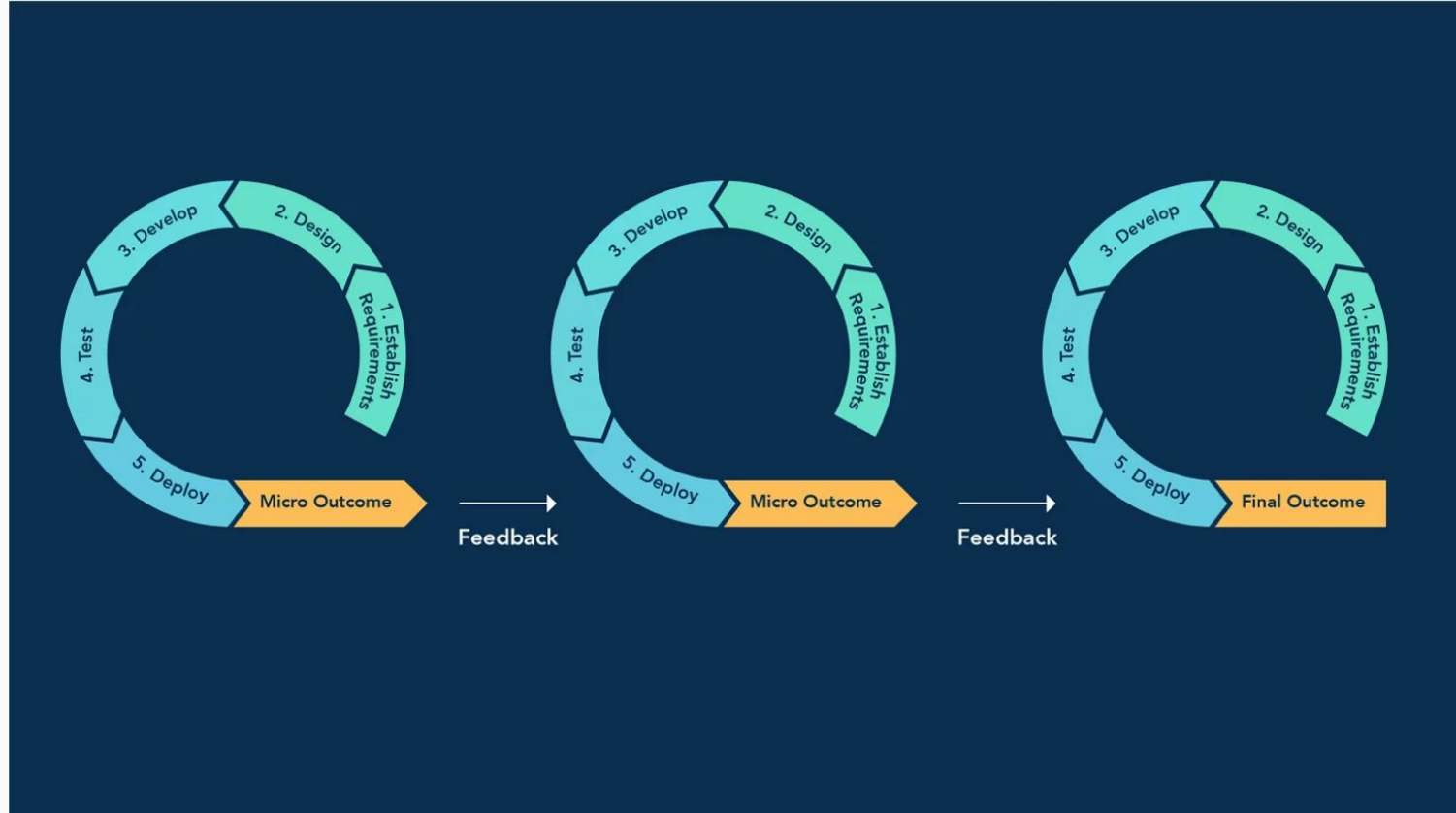


# Développement

# Evolution du développement - Approche cascade



# Evolution du développement - Méthode Agile



# Evolution du développement

- CI/CD - continuous integration, delivery, deployment
- Cycle court
- Collaborateurs et départements nombreux
- Environnements (quasi) identiques
- Rapide à lancer et jeter

★ Méthode Agile + conteneurisation ★



**Jenkins**



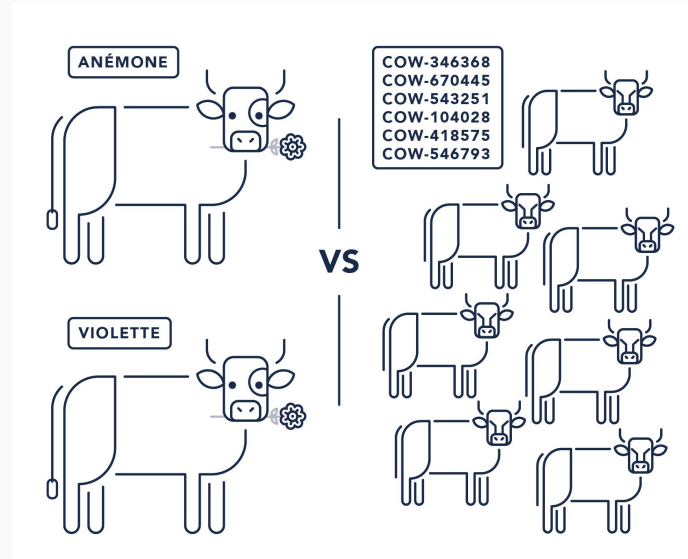
**DRONE**  
by harness



**CI/CD**

# Mutable and immutable infrastructures

# Infrastructures mutable et immuable

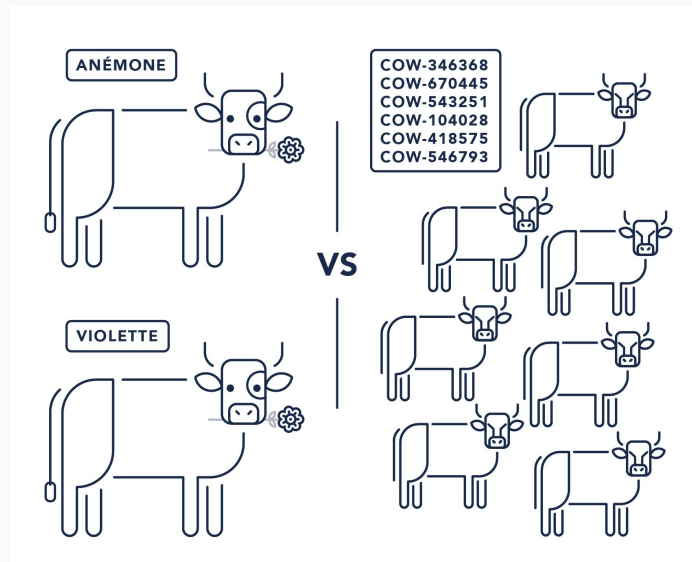


*Pets vs Cattle*

# Infrastructures mutable et immuable

## Mutable

- Maintenances complexes
- Upgrades risquées
- Configuration drift



*Pets vs Cattle*

## Immuable

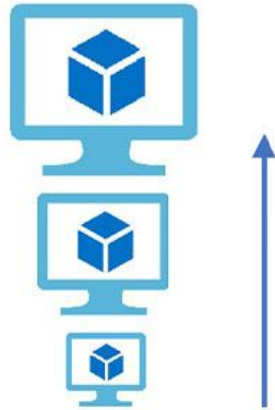
- Déploiements consistants
- Pas de configuration drift
- Réarchitecture des apps

# Vertical and horizontal scaling



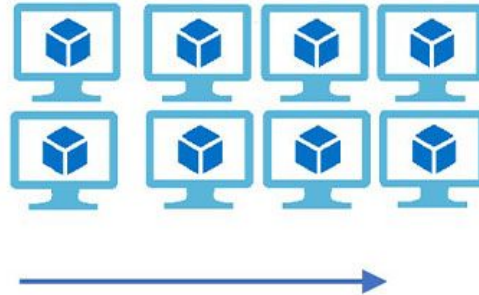
## Vertical Scaling

( Increase size of instance (RAM , CPU etc.) )



## Horizontal Scaling

( Add more instances )



Demo time



## Annexe 1 - VMs et conteneurs dans un bare-metal

