



Administration

Infrastructure

Séance 5

Sommaire

- Questions sur la séance labos ?
- Wooclap virtualisation
- Présentation Chapitre 12 Devops & Ansible
- Démo ansible
- Exercice papier ansible première version nodejs exoplanets



Questions sur la séance labos ?

- Modifier le Dockerfile -> souvent il faut recréer une image
- + Bien tout supprimer
 - `docker stop $(docker ps -aq)`
 - `docker system prune`

Wooclap

Comment participer ?




- 1 Allez sur wooclap.com
- 2 Entrez le code d'événement dans le bandeau supérieur

Code d'événement
ADMININFRA5



- 1 Envoyez **@ADMININFRA5** au **0460 200 711**
- 2 Vous pouvez participer

 Copier le lien de participation

Virtualisation - Conteneuration

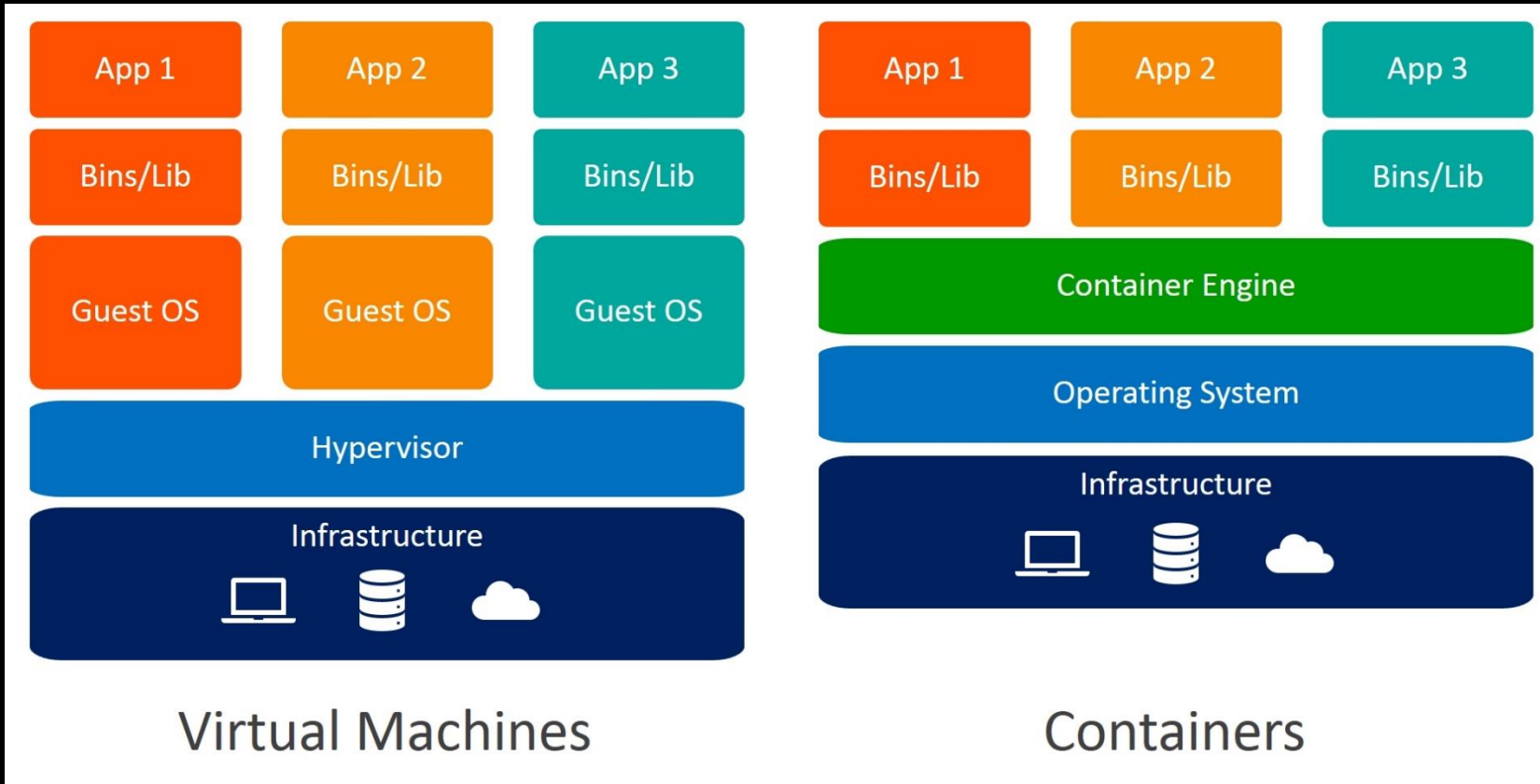
- Virtualisation -> s'abstraire du physique -> Pq ?
 - + Élément + configurable : changer disque physique vs changer taille VM Azure
 - + Élément + transportable : une VM Linux est facile à déplacer vs un serveur Linux physique
 - + Élément + optimisé : allocation dynamique des ressources
 - 1 serveur physique avec 16GB RAM vs un hyperviseur avec 16 GB RAM et 3 serveurs virtuels
 - + Élément + disponible : facile de déployer une Debian à partir d'une image
 - + Élément + sécurisé : isolation VM /conteneurs (NAT - port forwarding)

Le but de la virtualisation n'est pas forcément de rencontrer TOUS ses objectifs !

Types Virtualisation

- Serveurs
 - + Ex: VM
- Applications
 - + Ex: ThinApp, Xen, Docker
- Stockage
 - + Ex: SAN
- Réseau
 - + Ex: VLAN

Virtualisation vs Conteneurisation



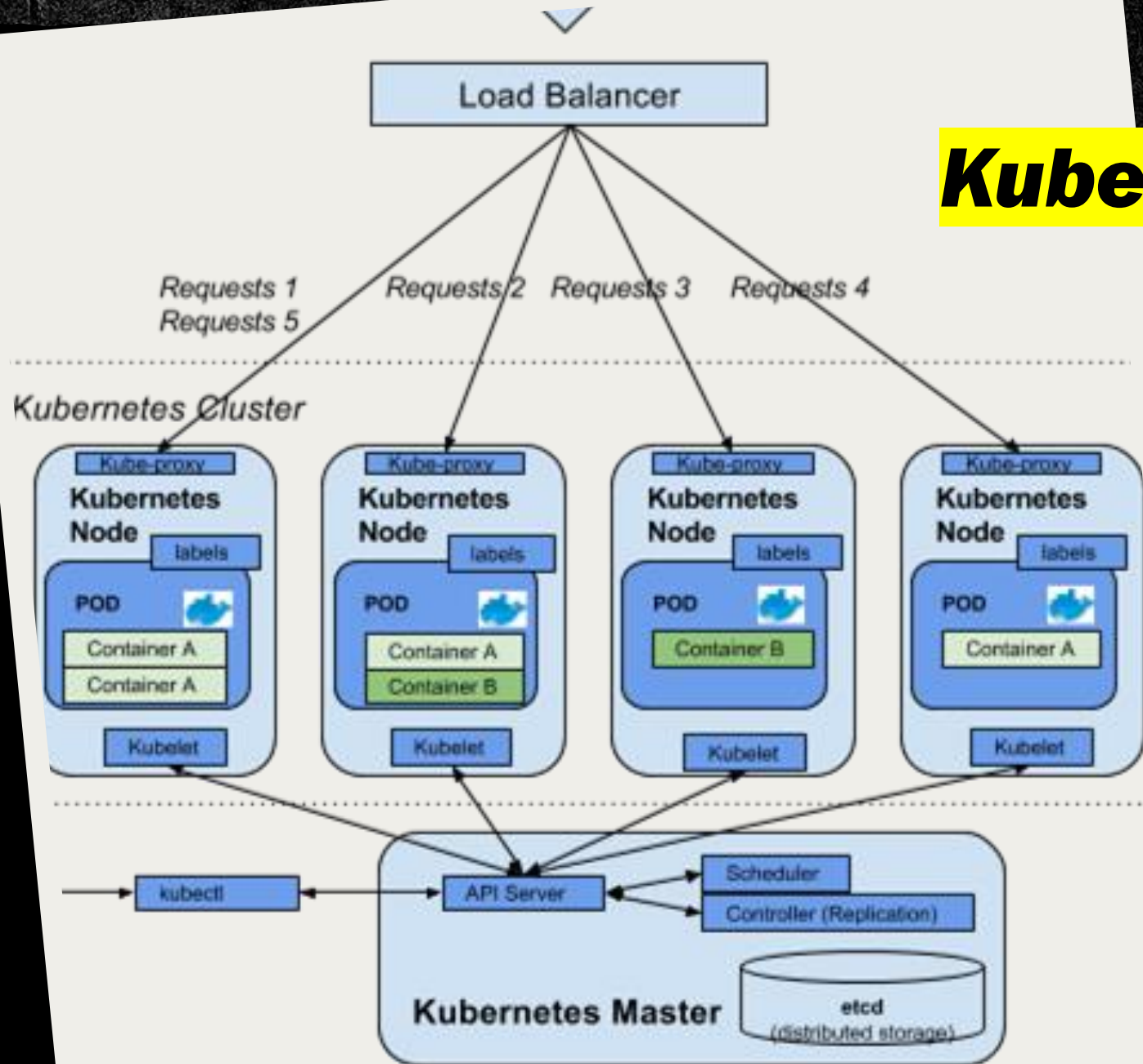
Virtualisation vs Conteneurisation

Différences		Docker	Machine Virtuelle
Optimisation		léger (taille et empreinte mémoire)	+ lourd (taille et empreinte mémoire)
Transport		Transport facile (taille légère)	Taille élevée
Disponibilité		Dockerhub et registry privée	Images sur Azure, ...
Configurable		moins d'options configurables	+ options configurables
Sécurité		moins de sécurité (partage du noyau par les conteneurs)	isolation peut être totale

Conteneurisation – 12 facteurs

- <https://12factor.net/fr/>

Kubernetes (K8s)

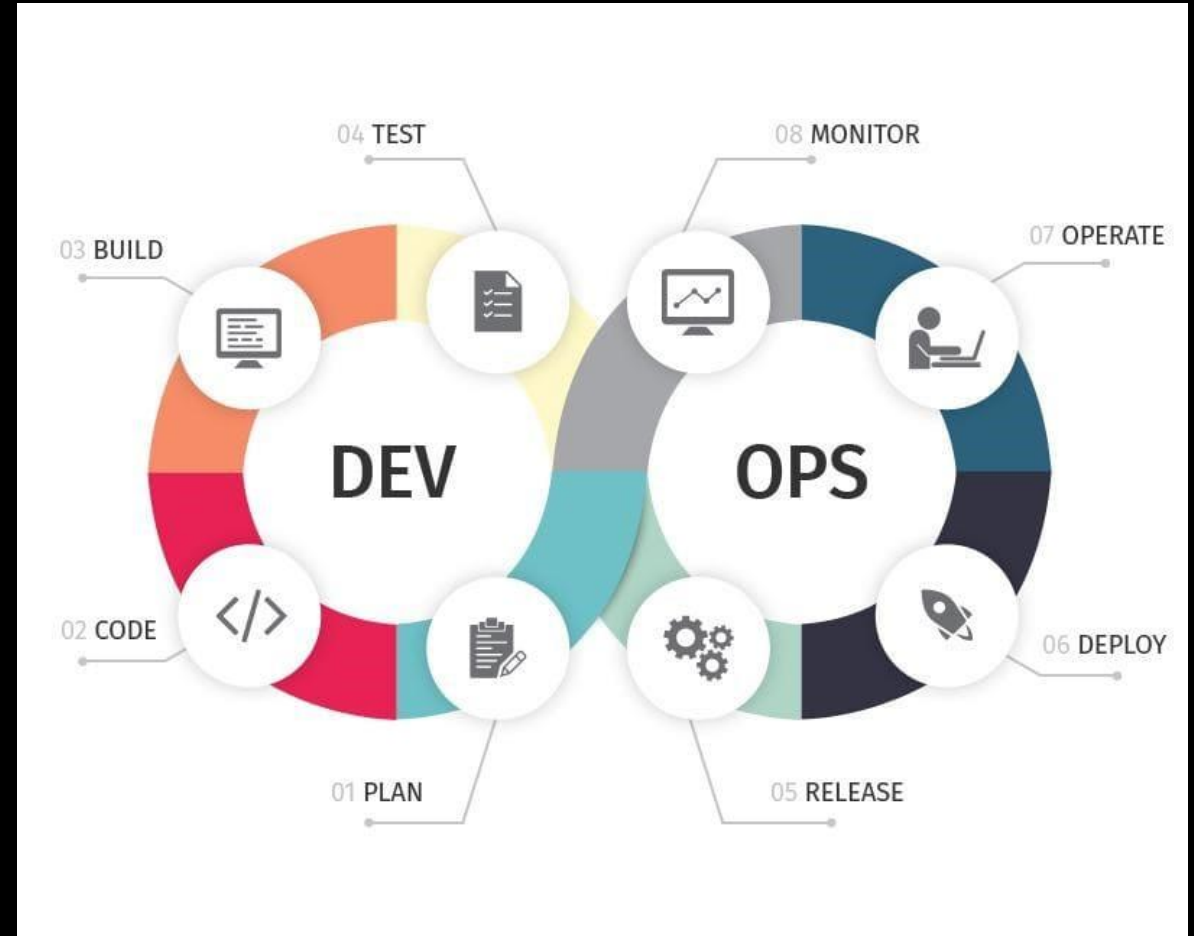


- Orchestrateur de conteneurs
- POD
- Nodes
- Load balancer

Chapitre 12 :

DevOps

- Quels outils sont utilisés pour le DevOps ?
+ Docker ou solution de conteneurs présente un peu partout



DevOps : pipelines et outils

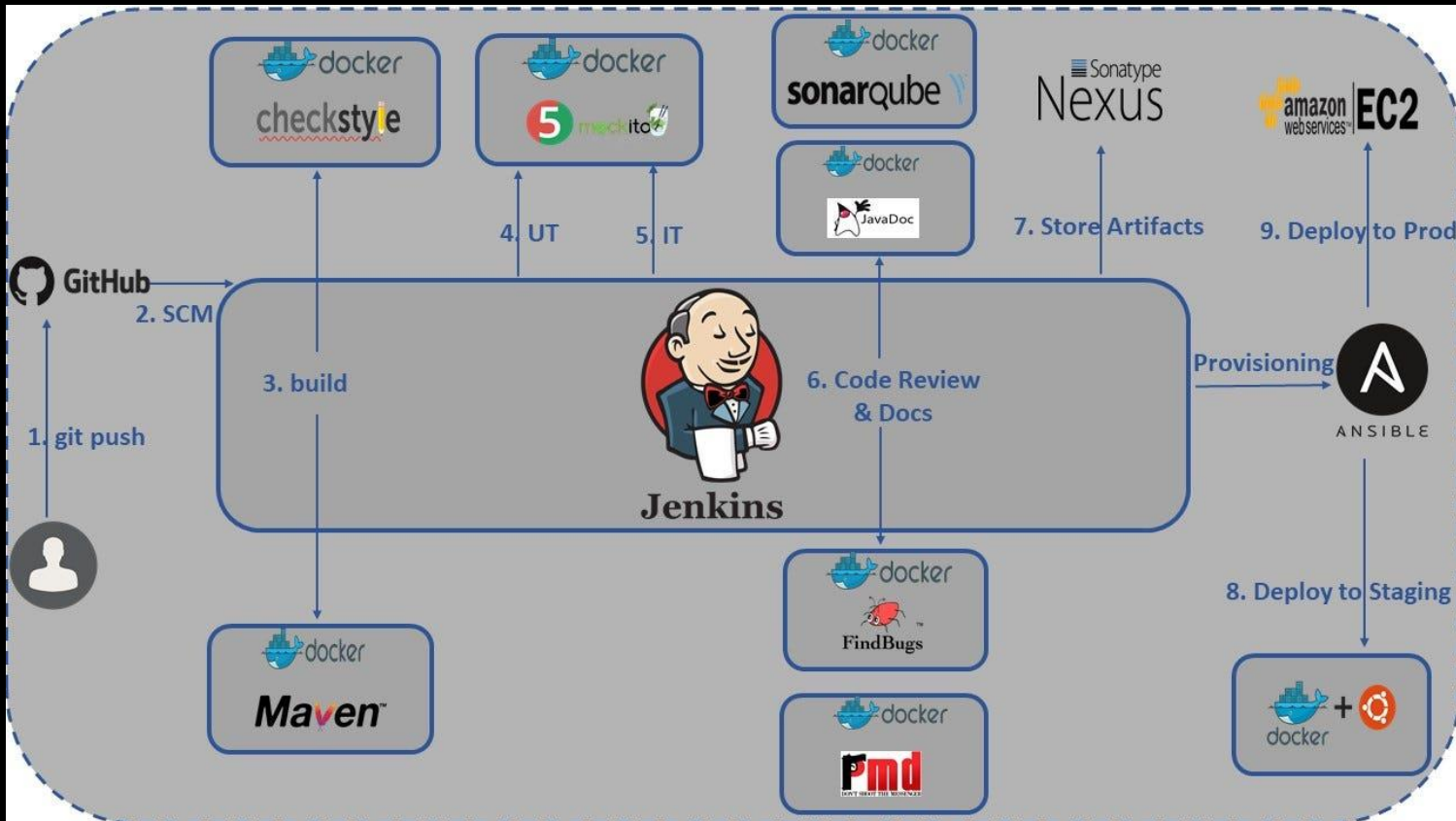


Image issue de : <https://medium.com/@ahmed24khaled/devops-ci-cd-pipelinee-e5a067786c26>

DevOps : pipelines & outils

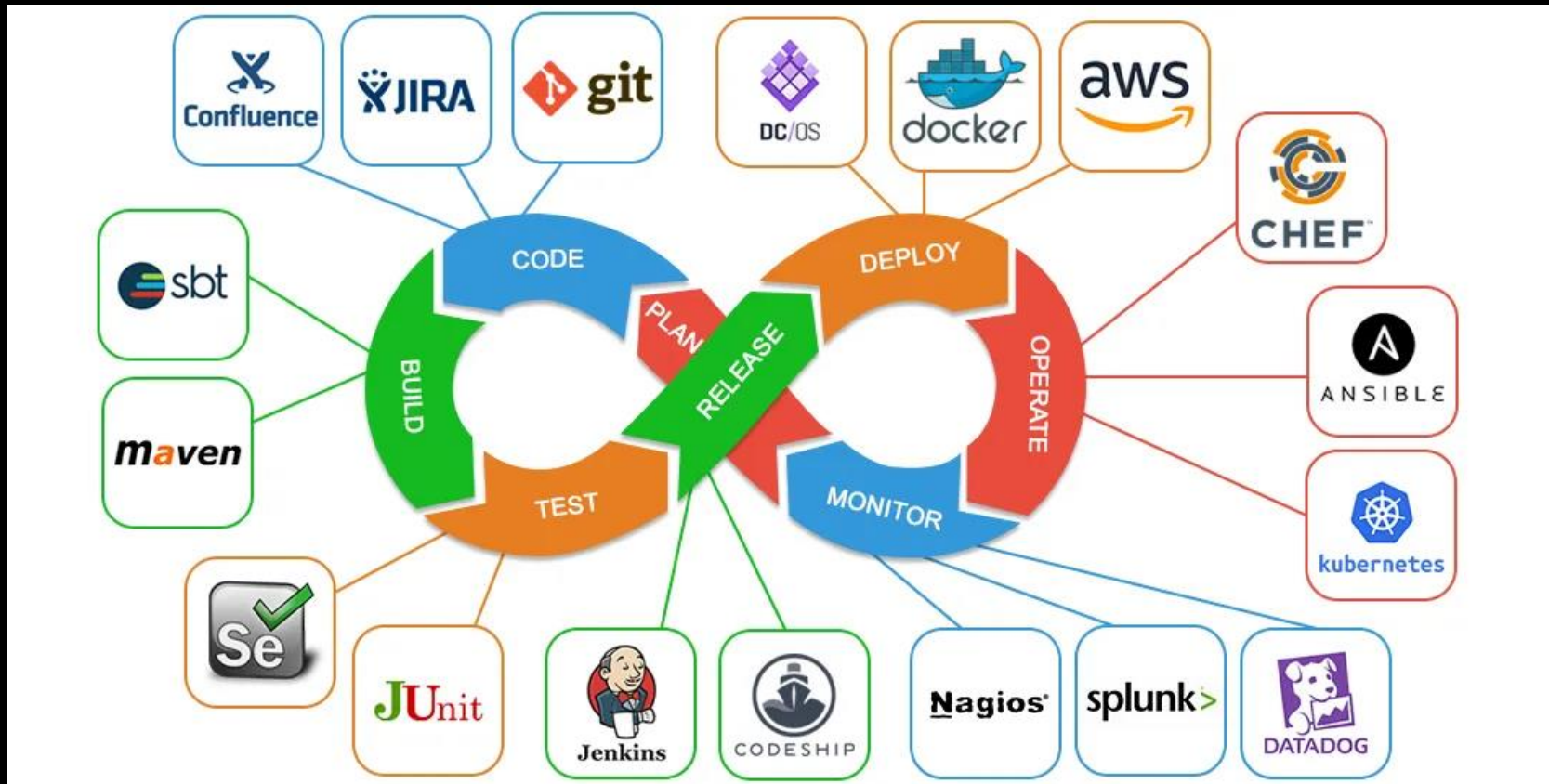
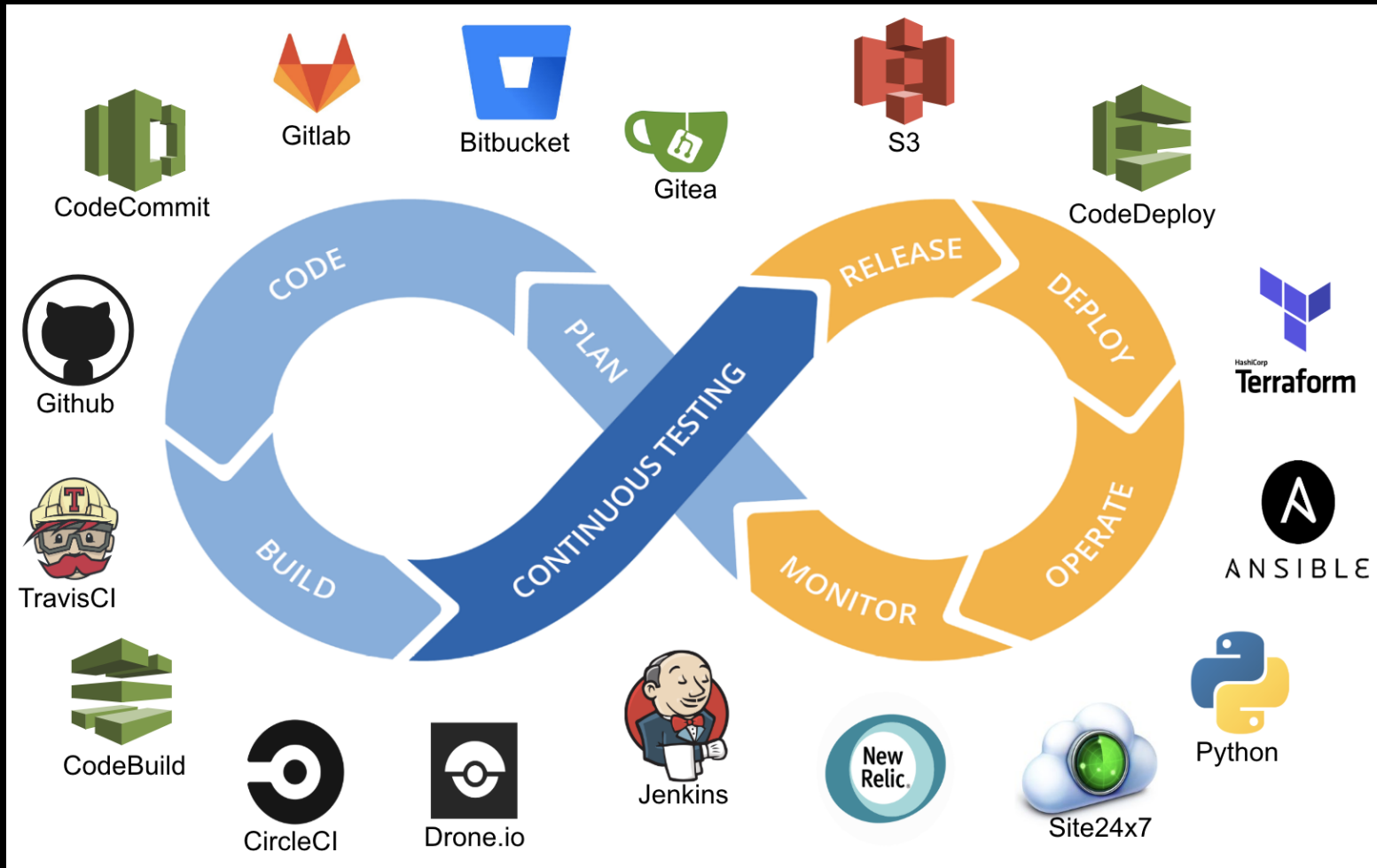


Image issue de : <https://medium.com/@ahmed24khaled/devops-ci-cd-pipelinee-e5a067786c26>

DevOps : pipelines & outils



DevOps : pipelines & outils

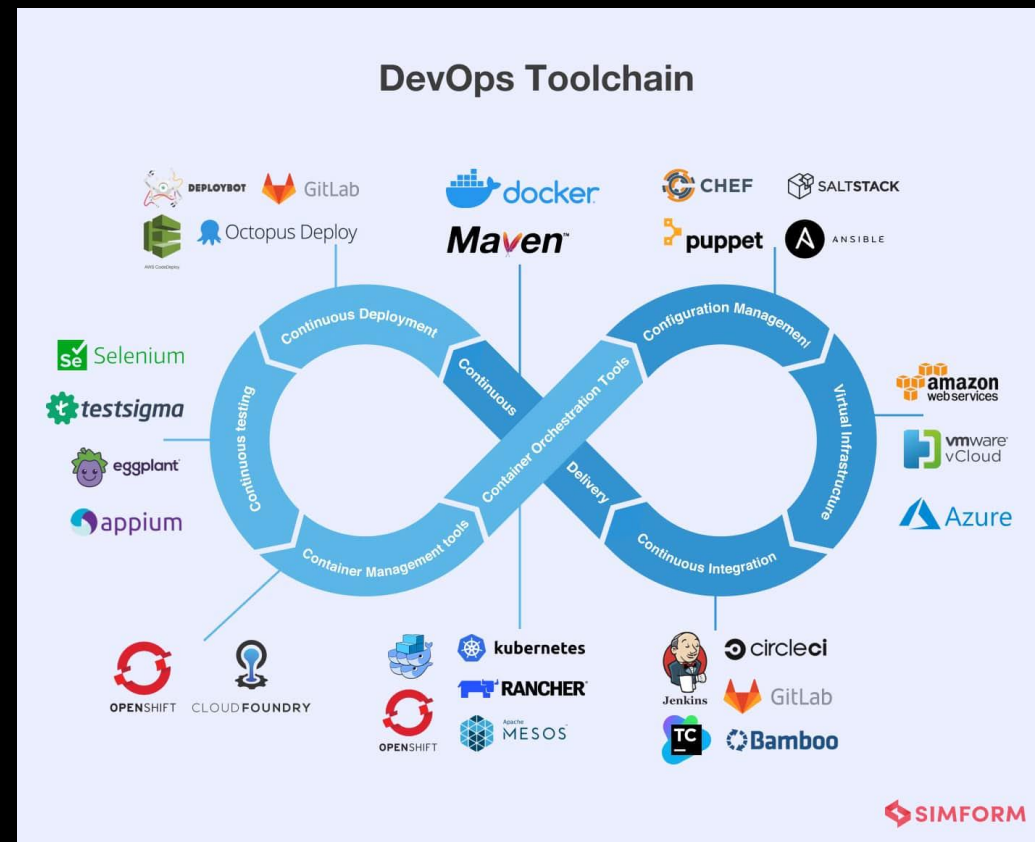
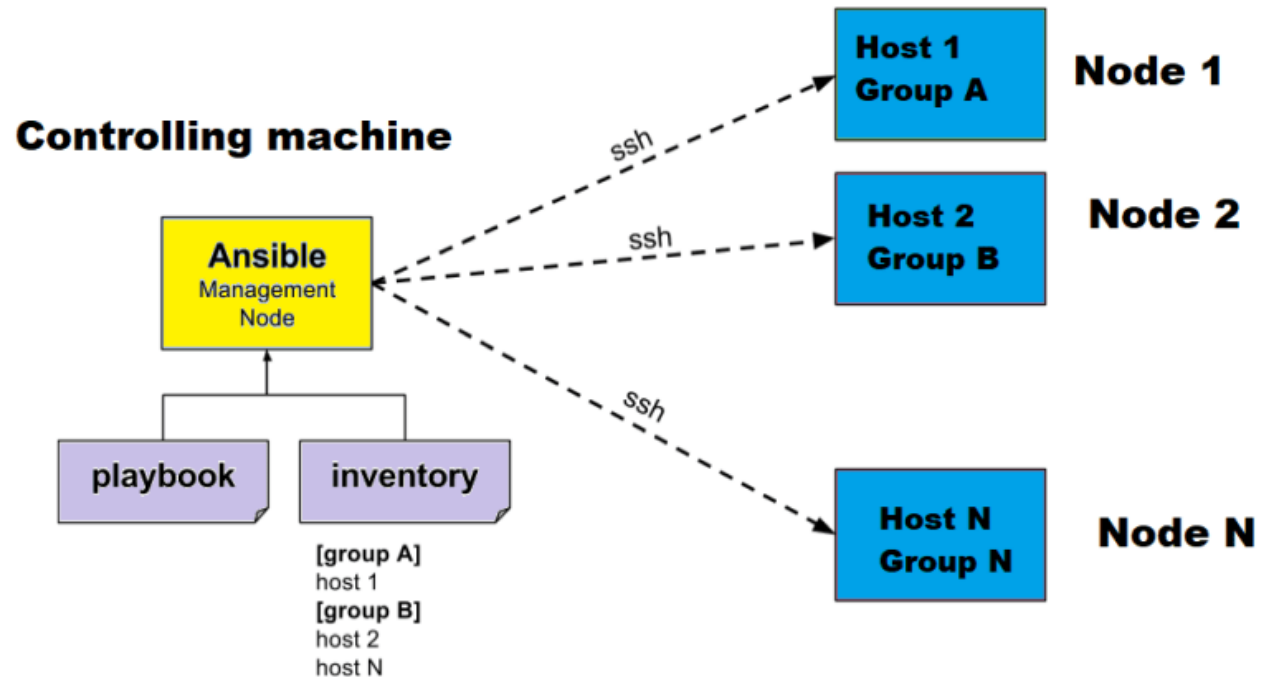


Image issue de : <https://www.simform.com/blog/devops-strategy/>

Devops : Ansible



DevOps : Ansible

- Liste des modules Ansible

+ https://docs.ansible.com/ansible/latest/collections/index_module.html

DevOps : Ansible

- Modules importants

- + apt

- + git

- + command

- + Template

- + <https://docs.ansible.com/ansible/latest/collections/ansible/builtin/index.html#plugins-in-ansible-builtin>

DevOps : Ansible

- Rôles

```
Ansible/
├── play.yml
├── roles
│   └── nginx
│       ├── files
│       │   ├── index.html
│       │   └── index.php
│       ├── handlers
│       │   └── main.yml
│       ├── meta
│       ├── tasks
│       │   └── main.yml
│       ├── templates
│       └── vars
```

DevOps : Ansible

- Rôle
 - + Répertoire
 - + Découpage structuré d'un playbook
 - defaults : valeurs par défaut des variables. celles-ci peuvent être surchargées par définition explicite ou vars
 - files : fichiers statiques à déployer
 - handlers : tâches pouvant être appelées par notification
 - tasks : tâches à effectuer
 - templates: modèles de fichier de configuration
 - vars : variables
 - README.md : description générale du rôle

DevOps : Ansible

```
---  
# gather facts, notify/handlers  
- hosts: localhost  
  gather_facts: true  
  tasks:  
    - name: Install Nginx  
      package:  
        name: nginx  
        state: latest  
      when: ansible_distribution == 'CentOS'  
      notify: restart nginx  
  
  handlers:  
    - name: restart nginx  
      service:  
        name: nginx  
        state: restarted
```

DevOps : Ansible

```
---  
# loop ands item  
- hosts: localhost  
  - name: Create new users  
    user:  
      name: '{{ item }}'  
      state: present  
    loop:  
      - john  
      - mike  
      - andrew
```


DevOps : Ansible

```
---
# become, vars , loop, gather facts
# gather facts true by default
- name: Install software
  become: yes
  hosts: all
  vars:
    packages:
      - name: neofetch
        required: True

      - name: cpu-checker
        required: True

      - name: screenfetch
        required: False
  tasks:
    - name: Install "{{ item.name }}" on Ubuntu
      apt:
        name: "{{ item.name }}"
        state: present

      when:
        - item.required == True
        - ansible_facts['distribution'] == "Ubuntu"

    loop: "{{ packages }}"
```

Démo : Ansible

- Déployer un site HTML avec Apache

Préparation labo

- Déploiement du site des Exoplanets - Première version
 - Voici les étapes pour déployer ce site :
 - + nodejs doit être installé sur la machine
 - + cloner le repository public
`https://gitlab.vinci.be/olivier.choquet/exoplanets_infra.git`
 - + lancer « npm install » dans le repo local
 - + lancer « npm start » dans le repo local