

CLOUD COMPUTING

Romain ODDONE

Qui suis-je ?

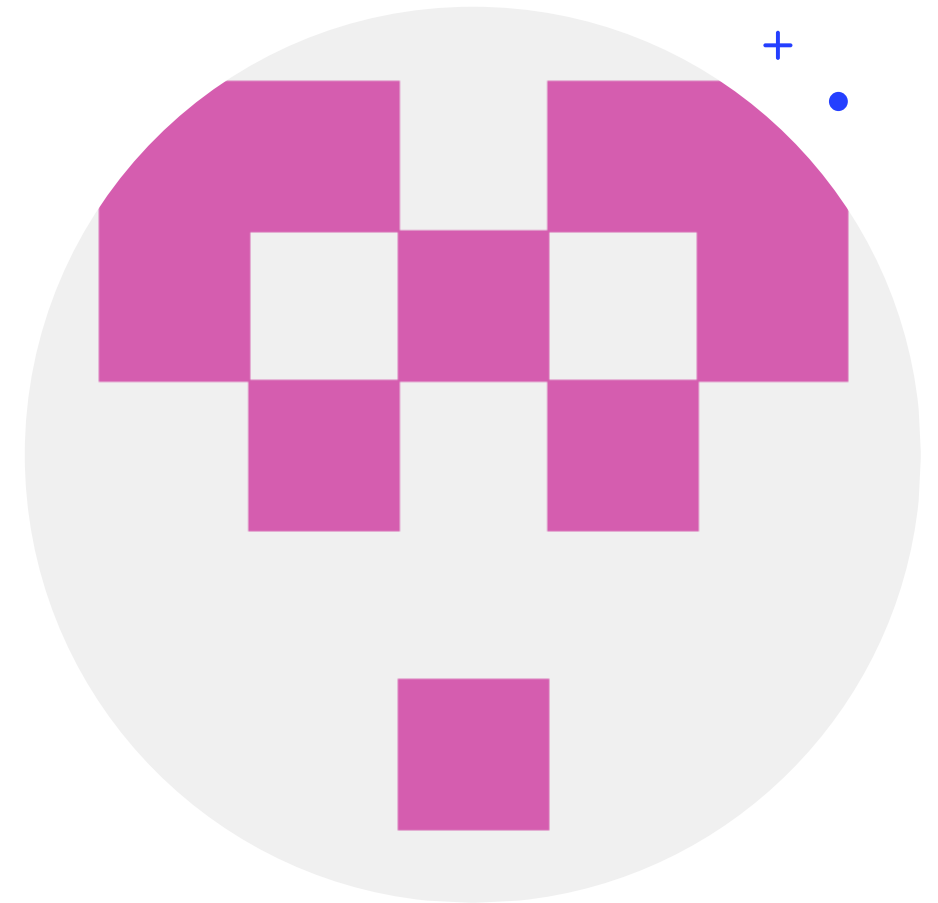
Architecte logiciel, 12 ans d'XP.

Parcours : Stagiaire > Développeur > Lead Développeur >
Tech lead > Architecte

Entreprise : Access IT

♥ Technos : .Net, Vue.js, Azure

<https://github.com/roddone/>





Qu'est-ce que le cloud computing ?

Composantes du Cloud

DevOps

+



Ressources / Services

Réseau

Composantes du Cloud : le DevOps

Né de la problématique suivante :

Dev

Bonjour l'équipe infra, je viens de terminer mon projet et il faudrait le déployer en production. Le code est sur github, bisous

7 minutes ago

Infra

T'es gentil mais je ne connais pas ce projet, j'ai besoin de savoir comment ça se déploie, en quelle technologie c'est développé etc.

4 minutes ago

Dev

c'est pourtant simple, c'est un projet node avec une API .Net et une BDD mongoDB, il suffit de builder le projet web, l'API et de modifier toutes les configurations pour que les services communiquent entre eux.

7 minutes ago

configurations pour que les services communiquent entre eux.

7 minutes ago

Infra

tu sais quoi, je ne suis pas développeur donc je n'y comprends rien, je te file un serveur vierge et tu vas t'en occuper toi-même.

4 minutes ago

Dev

oui mais moi je suis incapable d'installer un serveur de zéro, ce n'est pas mon métier ...

7 minutes ago

Composantes du Cloud : le devops

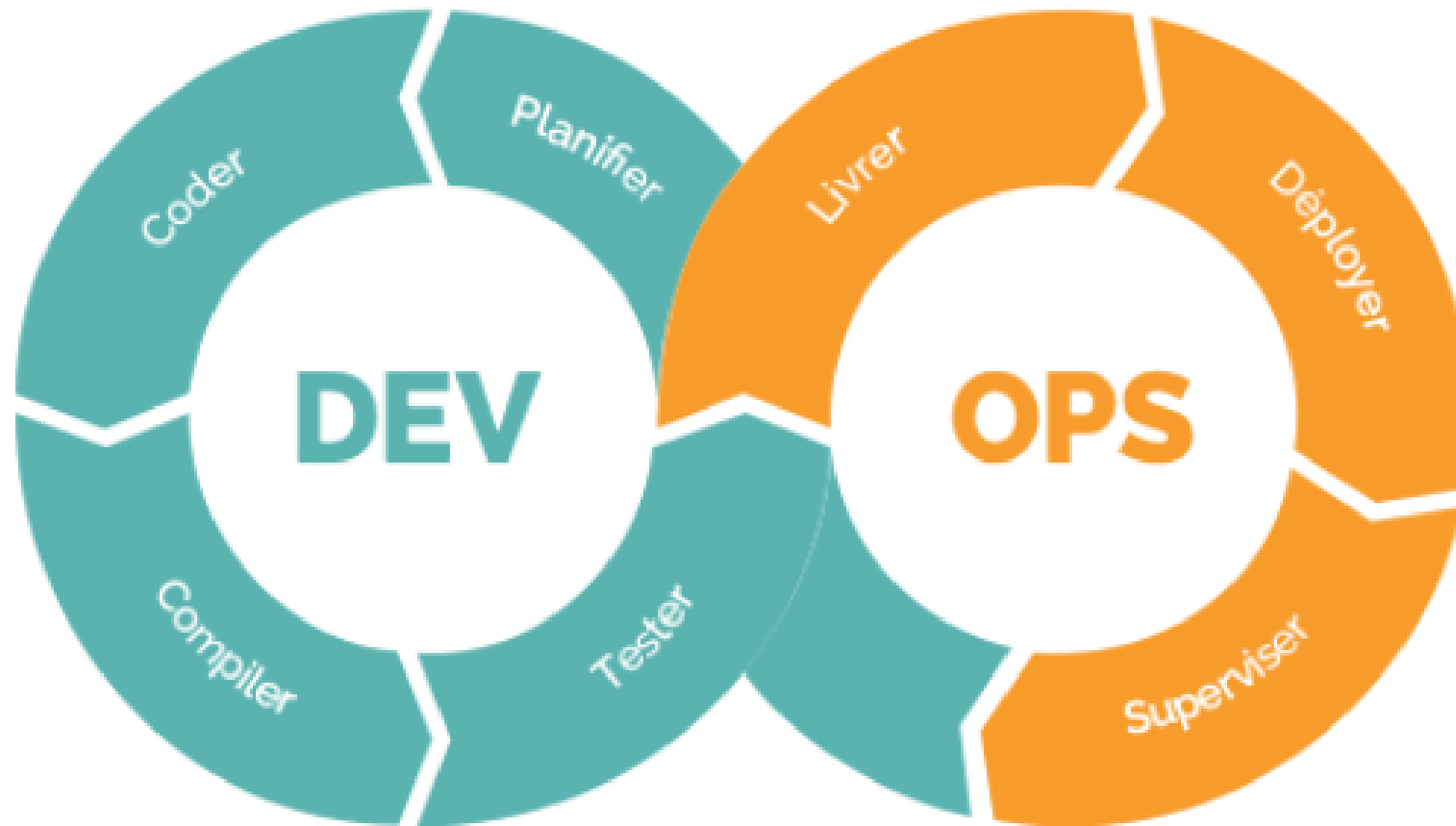
Pratiques de collaboration entre les développeurs de logiciels et les équipes en charge de l'infrastructure

+



Méthodes de travail, technologies permettant à des équipes de métiers différentes (dev et infra le plus souvent) de collaborer pour le déploiement de ressources.

Composantes du Cloud : le devops



Composantes du Cloud : le devops

CI/CD : **I**ntégration **C**ontinue / **D**éploiement **C**ontinu

+



Plutôt que de livrer le code manuellement et en un seul bloc aux équipes Infra, le code est **compilé, testé, packagé** et **livré** par une plateforme unique.

Différentes plateformes proposent cela :

Github

Gitlab

Sourceforge

Azure Devops

Azure Devops



Azure Boards

Outils de planification Agile



Azure Pipelines

CI/CD pour n'importe quelle plateforme



Azure Repos

Dépôts gratuits illimités



Offres de Test Azure

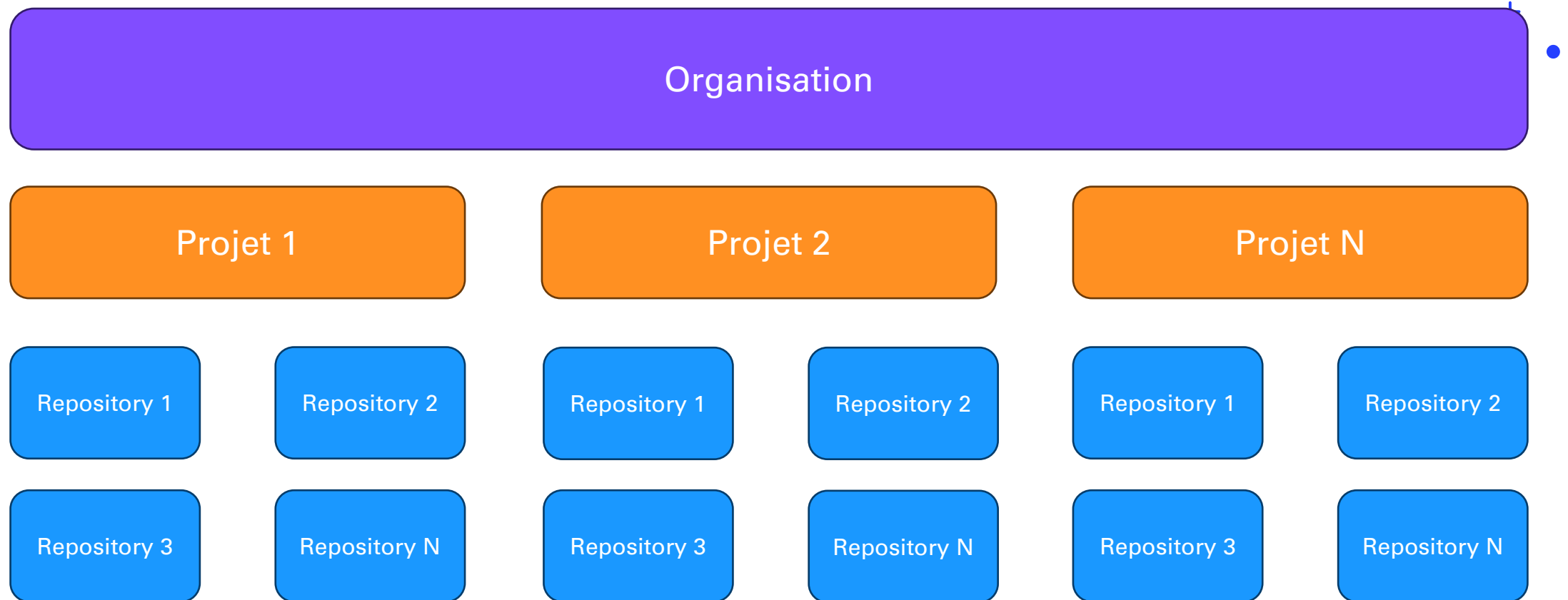
Tests exploratoires et manuels



Azure Artifacts

Dépôt de package universel

Azure Devops : Structure



Azure Devops



Azure Boards

Outils de planification Agile

Permet de définir le backlog du projet, le découpage en epics, US, tâches, bugs

Permet d'appliquer une méthode Agile (Scrum, Kanban)

Apporte les métriques permettant de suivre l'avancement du projet

Azure Devops



Azure Repos

Dépôts gratuits illimités

Dépôt de stockage du code source, basé sur git

+

•

Offre une vue permettant de voir l'état du code par fichier, par push, par commit, par branche etc.

Emplacement à partir duquel les Pull-Requests sont réalisés par les devs, et validés par le Lead/Archi.

Azure Devops



Azure Pipelines

CI/CD pour n'importe quelle plateforme

Gestion des « pipelines », les tâches permettant de tester et compiler un projet afin d'en produire un livrable. +

« Release » (déploiement) des livrables sur les différents environnements composant le projet (Dev/Staging/Préprod/Prod)

Stockage sécurisé de fichiers utilisés lors des déploiements (configurations, credentials, etc.).

Azure Devops



Offres de Test Azure

Tests exploratoires et manuels

Permet de créer des scénarios de tests manuels pour s'assurer du bon fonctionnement de l'application.

Offre une interface « côte à côte » avec le navigateur pour jouer les tests et reporter des bugs.

Azure Devops



Azure Artifacts

Dépôt de package universel

Offre un dépôt **sécurisé** et **privé** compatible avec de multiples providers :

- nuget (.Net)
- npm (Javascript)
- Cargo (Rust)
- Pip(Python)
- Maven (Java)
- Universal packages

Permet d'y stocker des dépendances privées pour les projets.

Azure Devops



Overview

Offre un dashboard personnalisable via des widgets, pour par exemple suivre les tâches qui me sont assignées, l'état des builds, le burndown du projet etc.



Offre aussi un Wiki pour le projet afin d'y mettre les informations utiles à l'équipe et aux nouveaux arrivants.

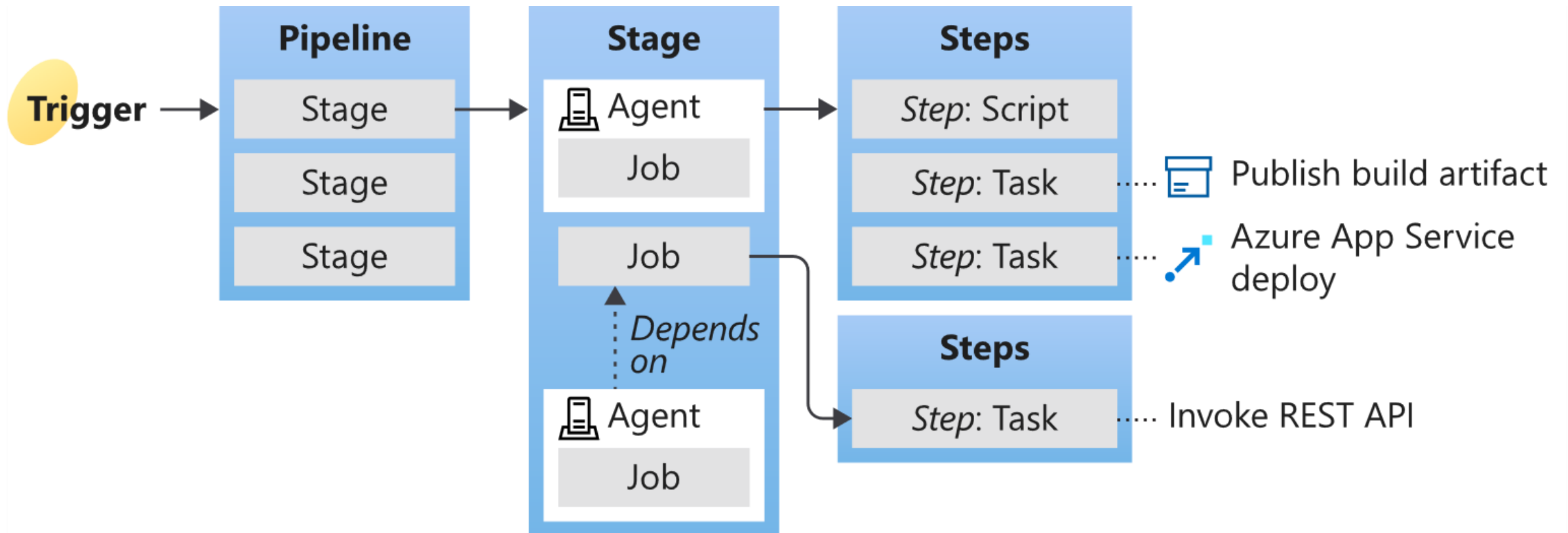
Pipelines

RTFM : <https://learn.microsoft.com/en-us/azure/devops/pipelines/?view=azure-devops>

+

•

Pipelines



Pipelines

Pipeline : ensemble d'actions permettant de préparer un projet en vue de son déploiement

Composé d'une ou plusieurs "phases" (stages)

Déclenché par un "trigger"

Fichier Yaml

Pipelines

```
1  trigger:
2    - lot-2
3
4  pool: ait-global
5
6  variables:
7    - name: tag
8      value: 'latest'
9
10 jobs:
11   - job: Back
12     steps:
13 >   - script: mkdir $(Build.ArtifactStagingDirectory)/images ...
14
15
16 >   - task: UseDotNet@2 ...
17
22
23 >   - template: _publish-api-project.yml ...
24
26
27 >   - job: Front ...
28
75
76 >   - job: Final ...
```

Pipelines

Trigger : évènement qui va provoquer l'exécution d'un pipeline.

Différents types de triggers :

- Commits sur une branche particulière
- Pull request
- Manuel
- Exécution planifiée
- Complétion d'un autre pipeline

YAML

```
trigger:  
- main  
- releases/*
```

YAML

```
# A pipeline with no CI trigger  
trigger: none
```

Pipelines

Stage: permet de regrouper plusieurs jobs

+

●

Job : regroupement de tâches/étapes (Steps) nécessaires pour publier un projet

Exécuté par un Agent

Agent : service exécutant les jobs

Peut être un agent cloud fourni par microsoft, ou un agent "hosted" installé sur un serveur physique, une machine virtuelle ou un container docker

Pipelines

Step: unité logique permettant d'effectuer une action

Peut être de différents types :

- Script (powershell ou Bash)
- Tâche disponible dans le catalogue

+

●

Cloud Computing

Le **Cloud Computing** consiste à fournir des services informatiques sur⁺ Internet. •

Services de base : machines virtuelles, stockage, base de données, réseau, sites webs

Services avancés : IoT, Machine Learning, IA

Cloud Computing vs 'On Premise'

On Premise (Cloud Privé) : serveurs physiques présents dans les locaux de l'entreprise

+++ **Coûts de setup élevés**, nécessité d'acheter des machines qui coûtent cher (plusieurs dizaines de milliers d'euros)

+ **Coûts réguliers de maintenance/infogérance** (mises à jour, dépoussiérage, remplacement de matériel, électricité)

++ **Coûts 'humains'**, nécessité de payer des employés spécialisés dédiés à l'exploitation des machines.

--- **pas de coûts** d'abonnements à un service Cloud externe

Cloud Computing vs 'On Premise'

On Premise : serveurs physiques présents dans les locaux de l'entreprise

Responsabilité des données : l'entreprise doit gérer elle-même la sécurité des données : droits d'accès, pare-feu etc.

Responsabilité de la SLA (Service Level Agreement) : temps de réponse, % de disponibilité

Cloud Computing vs 'On Premise'

Cloud Computing (Cloud publique) : pas de serveurs physiques, infogérance confiée à l'hébergeur Cloud

- **Coûts de setup faibles**, aucune machine à acheter
- **Aucun coûts réguliers** de maintenance/infogérance
- **Coûts 'humains' réduits**, nécessité moins de personnel car les tâches de maintenance sont réduites
- +(+++) **Coûts d'abonnements** réguliers à un hébergeur

Cloud Computing vs 'On Premise'

Cloud Computing : pas de serveurs physiques, infogérance confiée à l'hébergeur Cloud

Responsabilité des données : partagées entre l'entreprise et l'hébergeur cloud

Responsabilité de la SLA (Service Level Agreement) : confiée à l'hébergeur cloud

Cloud Computing vs 'On Premise'

Deux modes de facturation principaux : plans dédiés, et plans à la consommation.

Les plans dédiés ont un prix fixe et des ressources dédiées, tandis que les plans à la consommation ont un prix variable en fonction de l'utilisation

Cloud Computing vs 'On Premise'

Cloud Hybride : mix entre cloud privé et cloud publique

+

●

Utilisation partagée entre différents services issus d'un Cloud
Publique et privé

Exemple : hébergement d'un site web sur Azure, mais stockage des données sur des disques durs en interne

Cloud Computing : Une responsabilité partagée

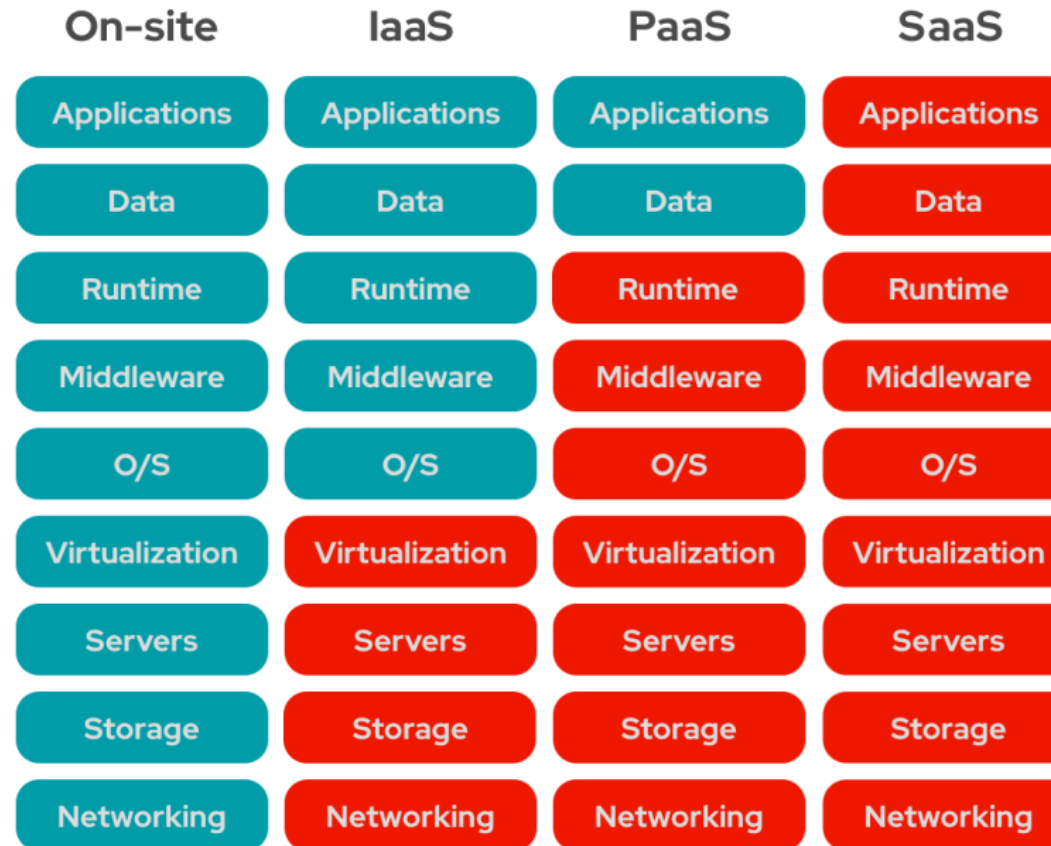
As A Service ?

Signifie qu'un tiers est responsable de vous fournir un service Cloud, afin qu'on puisse se concentrer sur l'essentiel : le code et les relations client.

Différents niveaux "As a Service" :



IaaS, PaaS, SaaS

Cloud Computing : Une responsabilité partagée



The diagram illustrates the shared responsibility model for cloud computing. It is structured as a 4x9 grid. The columns represent different service models: On-site, IaaS, PaaS, and SaaS. The rows represent different layers of the technology stack: Applications, Data, Runtime, Middleware, O/S, Virtualization, Servers, Storage, and Networking. Each cell in the grid contains a rounded rectangle with the layer name. The color of the rectangle indicates who is responsible for managing that layer: teal for 'You manage' and red for 'Service provider manages'. In the On-site column, all layers are teal. In the IaaS column, the top three layers (Applications, Data, Runtime) are teal, while the bottom six (Middleware, O/S, Virtualization, Servers, Storage, Networking) are red. In the PaaS column, the top two layers (Applications, Data) are teal, and the bottom seven are red. In the SaaS column, all layers are red.

On-site	IaaS	PaaS	SaaS
Applications	Applications	Applications	Applications
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
O/S	O/S	O/S	O/S
Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking

-  You manage
-  Service provider manages

Cloud Computing : Une responsabilité partagée

IaaS : Infrastructure as a Service

Solution la plus proche du "On Premise", offre un service sur des serveurs préconfigurés permettant à l'utilisateur de gérer lui-même ses ressources, sans devoir s'occuper de la maintenance de serveurs.

L'utilisateur est uniquement responsable de la maintenance de ses ressources.

Services IaaS connus : **Azure, AWS, Google Cloud**

IaaS

Applications

Data

Runtime

Middleware

O/S

Virtualization

Servers

Storage

Networking

Cloud Computing : Une responsabilité partagée

PaaS : Platform as a Service

Permet à l'utilisateur de créer ses propres applications, avec un cadre imposé (OS, logiciels et versions)

L'utilisateur n'a pas à se soucier des mises à jour/maintenance/infogérance, mais n'a pas non plus la possibilité de sortir du cadre qu'on lui impose. Risque d'être bloqué en cas de besoin spécifique

Services PaaS connus : **Heroku, Netlify, Salesforce**

PaaS

Applications

Data

Runtime

Middleware

O/S

Virtualization

Servers

Storage

Networking

Cloud Computing : Une responsabilité partagée

SaaS : Software as a Service

Le fournisseur Cloud s'occupe de tout, et fournit un logiciel 'clé en main' permettant à l'utilisateur de s'occuper de toutes les tâches nécessaires.

L'utilisateur n'a plus rien à gérer, mais est entièrement dépendant du logiciel fourni, de ses fonctionnalités et performances. Les évolutions/corrections sont au bon vouloir de l'hébergeur

Services SaaS connus : Wordpress, Dropbox, Outlook, Gmail

SaaS

Applications

Data

Runtime

Middleware

O/S

Virtualization

Servers

Storage

Networking

FinOps

Contraction de "Financial" et "DevOps"

+



Ensemble de pratiques permettant aux entreprises de maîtriser et réduire leurs coûts liés au Cloud

=> Permet d'économiser de l'argent pour vous et vos clients

FinOps

- Monitoring en temps réel pour étudier la charge de vos ressources⁺ •
- Scaling automatique en cas de forte charge
- Architectures Serverless
- Mutualisation de ressources
- Allumage/extinction automatique la nuit/week-ends

GreenOps

Contraction de "Green" et "DevOps"

+

●

Méthode d'optimisation de l'impact écologique des entreprises.

Équivalent du FinOps, pour l'environnement au lieu de l'argent

GreenOps et finOps peuvent être similaires, mais le Cloud "Vert" peut engendrer des coûts supplémentaires

GreenOps

- Réécriture de code, optimisation des algorithmes pour consommer⁺ moins de CPU et de RAM
- Bien choisir le langage de programmation : les langages bas niveau comme C ou Rust (ou même C# 😊) consomment beaucoup moins de ressources que Python ou Java
- Choisir un hébergeur qui utilise des énergies renouvelables (Azure ❤️)

GreenOps

- Mutualisation des ressources
- Suppression des ressources non utilisées
- Architectures Serverless
- Scaling automatique
- Extinction automatique des ressources

+



Azure

Démonstration :

Ressources Group, App Service Plan, App Service, Databases, Blob Storage, Virtual Machines, Azure Functions, Container Registry, IA

+

•