

# Architectures Logicielles

## Introduction au cloud computing

### Informatique et Statistique 4<sup>ème</sup> année

Olivier Caron<sup>1</sup>

<http://ocaron.polytech-lille.net>

<sup>1</sup>École d'ingénieurs Polytech Lille  
Université de Lille

6 mars 2025



## Dave Barry, écrivain

Comme la Hongrie, le monde informatique a une langue qui lui est propre. Mais il y a une différence. Si vous restez assez longtemps avec des Hongrois, vous finirez bien par comprendre de quoi ils parlent.

# Sources

- Plusieurs sources pour établir ce cours :

# Sources

- Plusieurs sources pour établir ce cours :
  - ▶ Wikipedia : <https://fr.wikipedia.org/>

- Plusieurs sources pour établir ce cours :
  - ▶ Wikipedia : <https://fr.wikipedia.org/>
  - ▶ Le livre blanc du cloud computing  
<http://www.virtu-desk.fr/medias/files/syntecle-livre-blanc-du-cloud-computing.pdf>

- Plusieurs sources pour établir ce cours :
  - ▶ Wikipedia : <https://fr.wikipedia.org/>
  - ▶ Le livre blanc du cloud computing  
<http://www.virtu-desk.fr/medias/files/syntecle-livre-blanc-du-cloud-computing.pdf>
  - ▶ Le livre "Cloud Computing, Sécurité, gouvernance du SI hybride et panorama du marché", Guillaume Plouin, 4<sup>e</sup> édition, Dunod

# Qu'est-ce que le cloud computing ?

- Littéralement, c'est l'informatique dans les nuages

# Qu'est-ce que le cloud computing ?

- Littéralement, c'est l'informatique dans les nuages
- Cela consiste à externaliser des infrastructures informatiques vers des opérateurs spécialisés.

# Qu'est-ce que le cloud computing ?

- Littéralement, c'est l'informatique dans les nuages
- Cela consiste à externaliser des infrastructures informatiques vers des opérateurs spécialisés.
- Cette seule définition semble bien insuffisante car dans ce cas :

# Qu'est-ce que le cloud computing ?

- Littéralement, c'est l'informatique dans les nuages
- Cela consiste à externaliser des infrastructures informatiques vers des opérateurs spécialisés.
- Cette seule définition semble bien insuffisante car dans ce cas :
  - ▶ 1990 : Le Network Computer (ordinateur Java connecté sans disque dur) = Cloud Computing ?

# Qu'est-ce que le cloud computing ?

- Littéralement, c'est l'informatique dans les nuages
- Cela consiste à externaliser des infrastructures informatiques vers des opérateurs spécialisés.
- Cette seule définition semble bien insuffisante car dans ce cas :
  - ▶ 1990 : Le Network Computer (ordinateur Java connecté sans disque dur) = Cloud Computing ?
  - ▶ 1989-1990 : Web = Cloud Computing ?

# Qu'est-ce que le cloud computing ?

- Littéralement, c'est l'informatique dans les nuages
- Cela consiste à externaliser des infrastructures informatiques vers des opérateurs spécialisés.
- Cette seule définition semble bien insuffisante car dans ce cas :
  - ▶ 1990 : Le Network Computer (ordinateur Java connecté sans disque dur) = Cloud Computing ?
  - ▶ 1989-1990 : Web = Cloud Computing ?
  - ▶ 1980 : Le minitel = Cloud Computing ?

# Qu'est-ce que le cloud computing ?

- Littéralement, c'est l'informatique dans les nuages
- Cela consiste à externaliser des infrastructures informatiques vers des opérateurs spécialisés.
- Cette seule définition semble bien insuffisante car dans ce cas :
  - ▶ 1990 : Le Network Computer (ordinateur Java connecté sans disque dur) = Cloud Computing ?
  - ▶ 1989-1990 : Web = Cloud Computing ?
  - ▶ 1980 : Le minitel = Cloud Computing ?
  - ▶ 1960 : Les terminaux passifs (mainframe) = Cloud Computing ?

# Qu'est-ce que le cloud computing ?

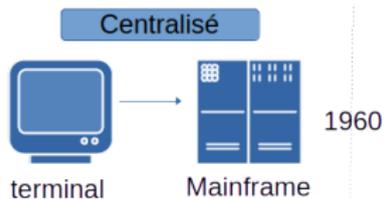
- Littéralement, c'est l'informatique dans les nuages
- Cela consiste à externaliser des infrastructures informatiques vers des opérateurs spécialisés.
- Cette seule définition semble bien insuffisante car dans ce cas :
  - ▶ 1990 : Le Network Computer (ordinateur Java connecté sans disque dur) = Cloud Computing ?
  - ▶ 1989-1990 : Web = Cloud Computing ?
  - ▶ 1980 : Le minitel = Cloud Computing ?
  - ▶ 1960 : Les terminaux passifs (mainframe) = Cloud Computing ?

## Le cloud computing, c'est ...

L'émergence / la maturité de plusieurs technologies et l'adoption de celles-ci par les utilisateurs.

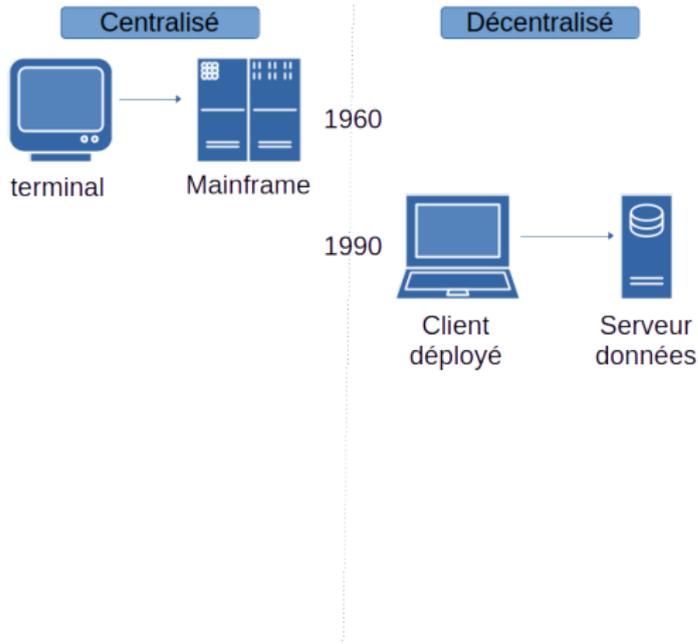
# Origine du cloud computing n°1 : HTML 5

# Origine du cloud computing n°1 : HTML 5

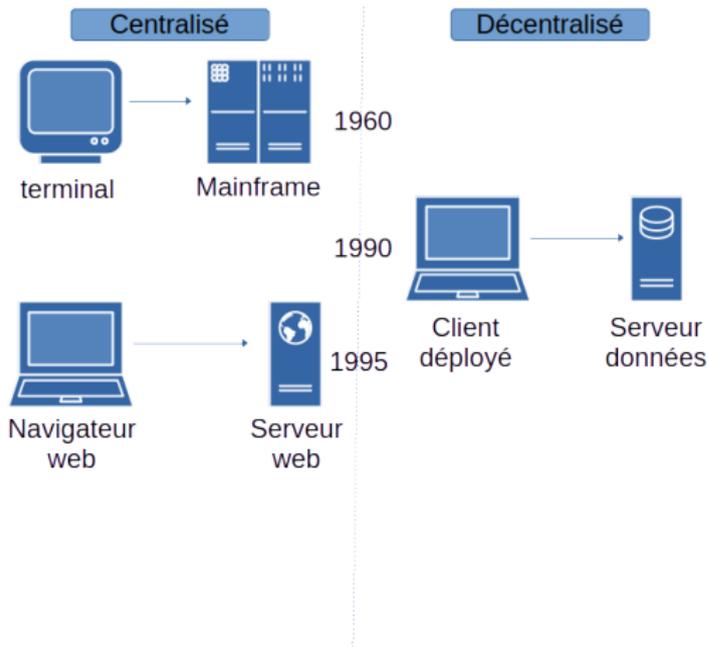


Décentralisé

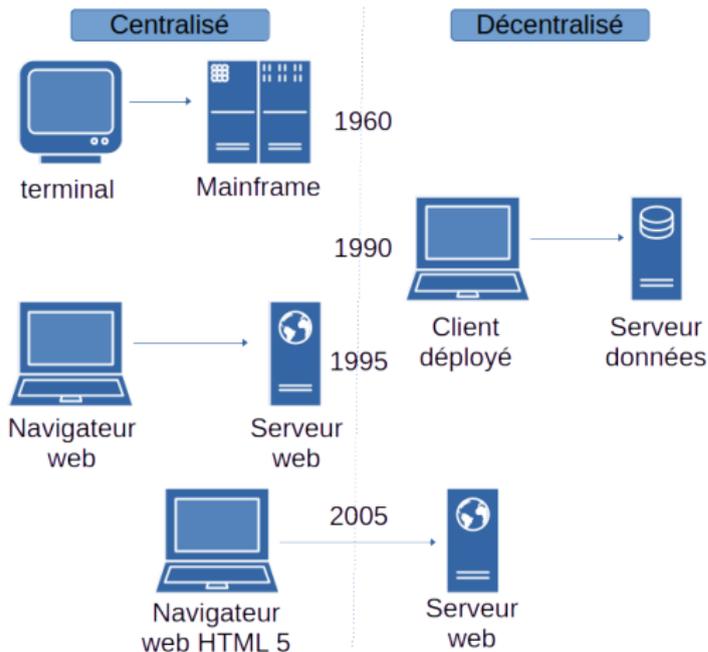
# Origine du cloud computing n°1 : HTML 5



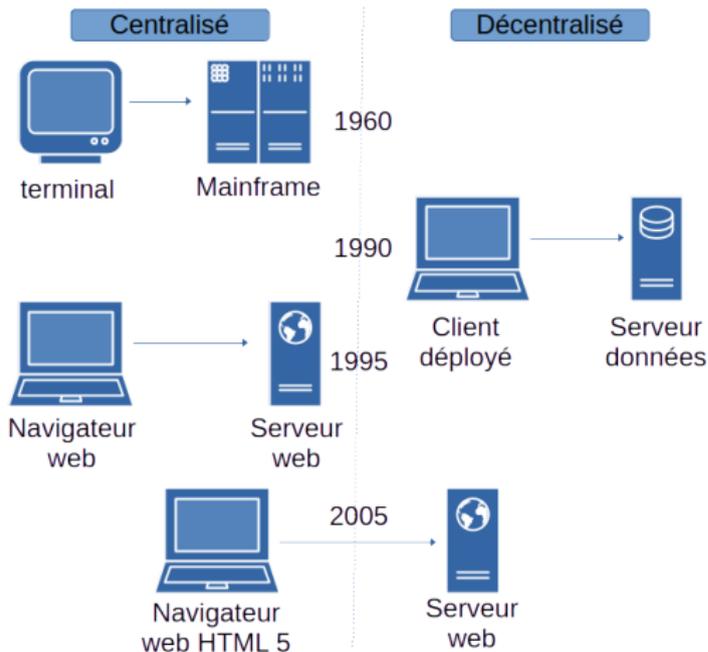
# Origine du cloud computing n°1 : HTML 5



# Origine du cloud computing n°1 : HTML 5

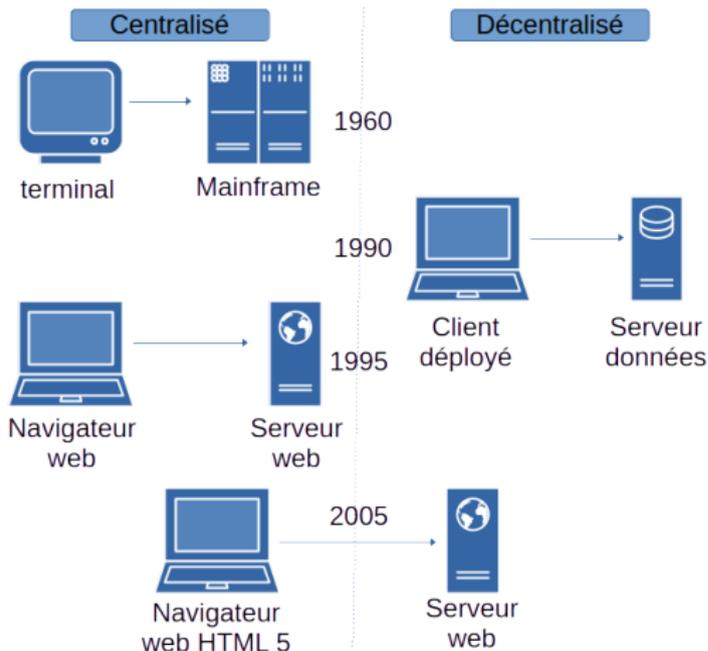


# Origine du cloud computing n°1 : HTML 5



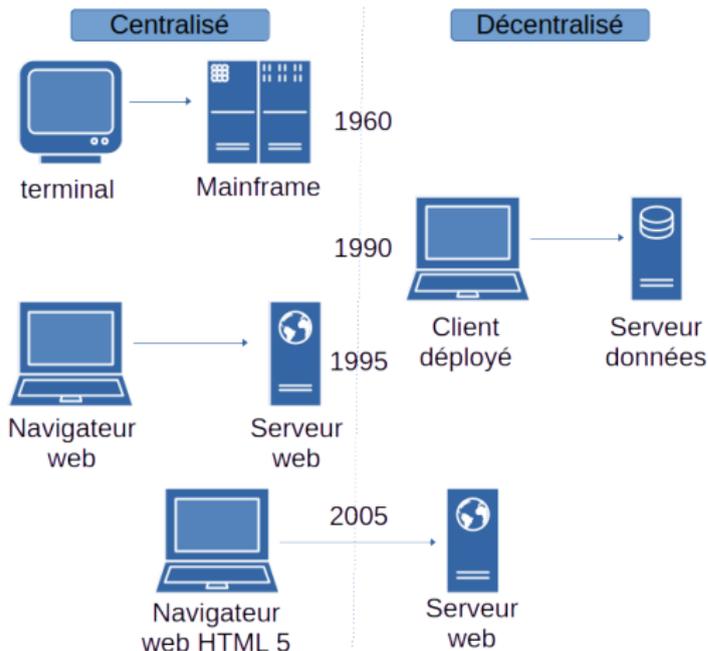
- HTML 5 = HTML + CSS + Javascript (web 2.0)

# Origine du cloud computing n°1 : HTML 5



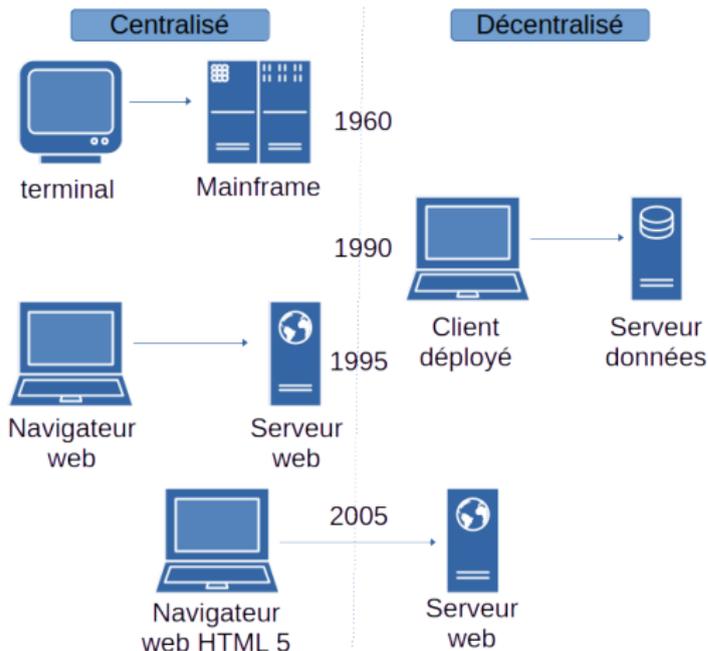
- HTML 5 = HTML + CSS + Javascript (web 2.0)
- Concept du RIA : Rich Internet Application

# Origine du cloud computing n°1 : HTML 5



- HTML 5 = HTML + CSS + Javascript (web 2.0)
- Concept du RIA : Rich Internet Application
- la bêta perpétuelle : plus de version mais une évolution constante et transparente.

# Origine du cloud computing n°1 : HTML 5



- HTML 5 = HTML + CSS + Javascript (web 2.0)
- Concept du RIA : Rich Internet Application
- la bêta perpétuelle : plus de version mais une évolution constante et transparente.

## Qualités ergonomiques

Adoption des interfaces graphiques web dites 2.0

## Origine du cloud computing n°2 : l'intelligence collective

- Adoption du web 2.0 : Wordpress, Gmail Facebook, Twitter, LinkedIn, Flickr, Youtube, Instagram, Google maps. . .

## Origine du cloud computing n°2 : l'intelligence collective

- Adoption du web 2.0 : Wordpress, Gmail Facebook, Twitter, LinkedIn, Flickr, Youtube, Instagram, Google maps. . .
- "Intelligence collective" : chacun peut contribuer à alimenter les données (wikipedia, avis utilisateurs, . . .)

## Origine du cloud computing n°2 : l'intelligence collective

- Adoption du web 2.0 : Wordpress, Gmail Facebook, Twitter, LinkedIn, Flickr, Youtube, Instagram, Google maps. . .
- "Intelligence collective" : chacun peut contribuer à alimenter les données (wikipedia, avis utilisateurs, . . .)
- Mais. . .importance de la modération. . .

# Origine du cloud computing n°2 : l'intelligence collective

- Adoption du web 2.0 : Wordpress, Gmail Facebook, Twitter, LinkedIn, Flickr, Youtube, Instagram, Google maps. . .
- "Intelligence collective" : chacun peut contribuer à alimenter les données (wikipedia, avis utilisateurs, . . .)
- Mais. . .importance de la modération. . .

## Données et applications externes

Augmentation de la confiance dans les données et application externes.

# Origine du cloud computing n°2 : l'intelligence collective

- Adoption du web 2.0 : Wordpress, Gmail Facebook, Twitter, LinkedIn, Flickr, Youtube, Instagram, Google maps. . .
- "Intelligence collective" : chacun peut contribuer à alimenter les données (wikipedia, avis utilisateurs, . . .)
- Mais. . .importance de la modération. . .

## Données et applications externes

Augmentation de la confiance dans les données et application externes.

- Attention à ne pas être trop confiant.

# Origine du cloud computing n°3 : les API ouvertes

- Mise à disposition des API ouvertes (Application Programming Interface)

## Origine du cloud computing n°3 : les API ouvertes

- Mise à disposition des API ouvertes (Application Programming Interface)
- Il est donc possible d'invoquer ces services depuis d'autres applications

## Origine du cloud computing n°3 : les API ouvertes

- Mise à disposition des API ouvertes (Application Programming Interface)
- Il est donc possible d'invoquer ces services depuis d'autres applications
- Les applications uniquement bâties sur des API sont des applications appelées **mashups**

## Origine du cloud computing n°3 : les API ouvertes

- Mise à disposition des API ouvertes (Application Programming Interface)
- Il est donc possible d'invoquer ces services depuis d'autres applications
- Les applications uniquement bâties sur des API sont des applications appelées **mashups**
- Exemple du dernier sujet de TP : application qui affiche une carte (<https://www.openstreetmap.org>) avec les infos des écoles Polytech en indiquant en outre la météo de chaque école (<https://www.openweathermap.org>)

## Origine du cloud computing n°3 : les API ouvertes

- Mise à disposition des API ouvertes (Application Programming Interface)
- Il est donc possible d'invoquer ces services depuis d'autres applications
- Les applications uniquement bâties sur des API sont des applications appelées **mashups**
- Exemple du dernier sujet de TP : application qui affiche une carte (<https://www.openstreetmap.org>) avec les infos des écoles Polytech en indiquant en outre la météo de chaque école (<https://www.openweathermap.org>)
- Un modèle économique possible (et déjà utilisé) : gratuit en cas de faible demande, payant à partir de  $x$  requêtes.

# Origine du cloud computing n°3 : les API ouvertes

- Mise à disposition des API ouvertes (Application Programming Interface)
- Il est donc possible d'invoquer ces services depuis d'autres applications
- Les applications uniquement bâties sur des API sont des applications appelées **mashups**
- Exemple du dernier sujet de TP : application qui affiche une carte (<https://www.openstreetmap.org>) avec les infos des écoles Polytech en indiquant en outre la météo de chaque école (<https://www.openweathermap.org>)
- Un modèle économique possible (et déjà utilisé) : gratuit en cas de faible demande, payant à partir de  $x$  requêtes.

## Accessibilité

Les données externes sont de plus en plus facilement accessibles.

# Origine du cloud computing n°4 : la virtualisation (1/2)

- Différents types de virtualisation :

# Origine du cloud computing n°4 : la virtualisation (1/2)

- Différents types de virtualisation :

**Virtualisation système** consiste à exécuter sur une machine hôte, dans un environnement isolé, des systèmes d'exploitation.

# Origine du cloud computing n°4 : la virtualisation (1/2)

- Différents types de virtualisation :

**Virtualisation système** consiste à exécuter sur une machine hôte, dans un environnement isolé, des systèmes d'exploitation.

**Virtualisation applicative** consiste à exécuter sur une machine hôte, dans un environnement isolé, des applications.

# Origine du cloud computing n°4 : la virtualisation (1/2)

- Différents types de virtualisation :

**Virtualisation système** consiste à exécuter sur une machine hôte, dans un environnement isolé, des systèmes d'exploitation.

**Virtualisation applicative** consiste à exécuter sur une machine hôte, dans un environnement isolé, des applications.

**Virtualisation des données** consiste à fournir une interface unifiée de données issues de plusieurs sources et occulte ainsi les détails sur ces sources (emplacement, structures de données d'origine)

# Origine du cloud computing n°4 : la virtualisation (1/2)

- Différents types de virtualisation :

**Virtualisation système** consiste à exécuter sur une machine hôte, dans un environnement isolé, des systèmes d'exploitation.

**Virtualisation applicative** consiste à exécuter sur une machine hôte, dans un environnement isolé, des applications.

**Virtualisation des données** consiste à fournir une interface unifiée de données issues de plusieurs sources et occulte ainsi les détails sur ces sources (emplacement, structures de données d'origine)

- Coût matériel. Exemple : on utilise une machine qui installe 5 serveurs virtuels exploitant chacun 20% de l'activité de la machine physique.

# Origine du cloud computing n°4 : la virtualisation (1/2)

- Différents types de virtualisation :

**Virtualisation système** consiste à exécuter sur une machine hôte, dans un environnement isolé, des systèmes d'exploitation.

**Virtualisation applicative** consiste à exécuter sur une machine hôte, dans un environnement isolé, des applications.

**Virtualisation des données** consiste à fournir une interface unifiée de données issues de plusieurs sources et occulte ainsi les détails sur ces sources (emplacement, structures de données d'origine)

- Coût matériel. Exemple : on utilise une machine qui installe 5 serveurs virtuels exploitant chacun 20% de l'activité de la machine physique.
- Élasticité : on augmente, on diminue le nombre de serveurs virtuels sur une machine selon le taux d'activité. On peut déplacer facilement un serveur virtuel sur une autre machine.

## Origine du cloud computing n°4 : la virtualisation (2/2)

- Sécurité : un serveur virtuel peut être attaqué, plus difficilement l'**hyperviseur** de la machine hôte.

## Origine du cloud computing n°4 : la virtualisation (2/2)

- Sécurité : un serveur virtuel peut être attaqué, plus difficilement **l'hyperviseur** de la machine hôte.
- Résistance aux pannes : un serveur physique tombe en panne, on installe **l'image du serveur virtuel** sur une autre machine

## Origine du cloud computing n°4 : la virtualisation (2/2)

- Sécurité : un serveur virtuel peut être attaqué, plus difficilement **l'hyperviseur** de la machine hôte.
- Résistance aux pannes : un serveur physique tombe en panne, on installe **l'image du serveur virtuel** sur une autre machine
- Performance : Au fil des années, la dégradation des performances s'est considérablement réduite, ce gain a été notamment boosté par des processeurs avec support de la virtualisation.

## Origine du cloud computing n°4 : la virtualisation (2/2)

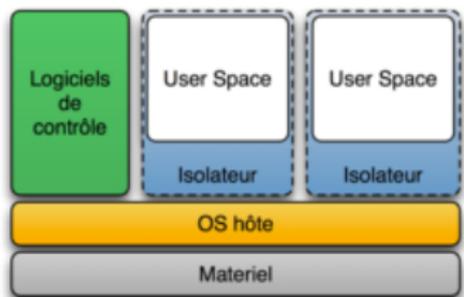
- Sécurité : un serveur virtuel peut être attaqué, plus difficilement **l'hyperviseur** de la machine hôte.
- Résistance aux pannes : un serveur physique tombe en panne, on installe **l'image du serveur virtuel** sur une autre machine
- Performance : Au fil des années, la dégradation des performances s'est considérablement réduite, ce gain a été notamment boosté par des processeurs avec support de la virtualisation.

### La virtualisation : une technologie éprouvée

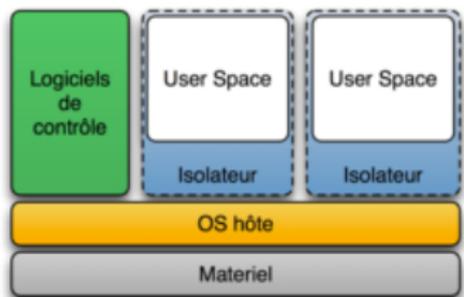
C'est une technologie qui est arrivée à maturité et appréciée par les DSI pour toutes ses qualités

# Variante 1 de la virtualisation : la conteneurisation

# Variante 1 de la virtualisation : la conteneurisation

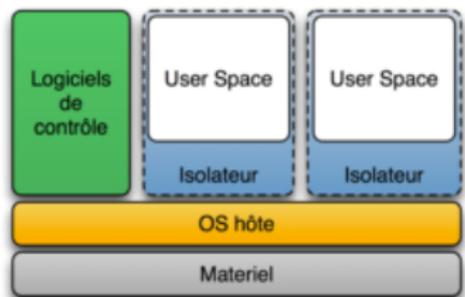


# Variante 1 de la virtualisation : la conteneurisation

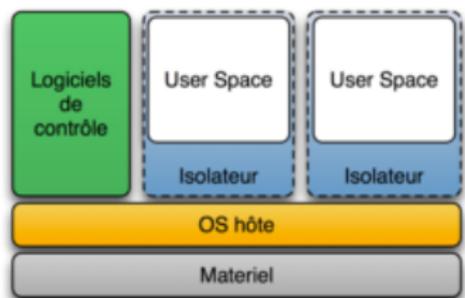


# Variante 1 de la virtualisation : la conteneurisation

- Un **isolateur** ou **conteneur** permet d'isoler l'exécution d'une application dans un contexte.

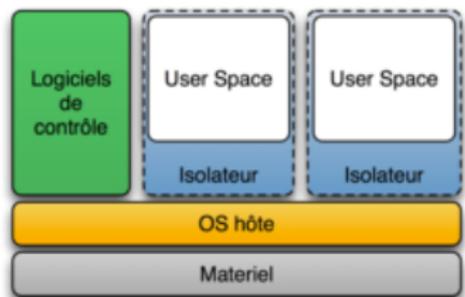


# Variante 1 de la virtualisation : la conteneurisation



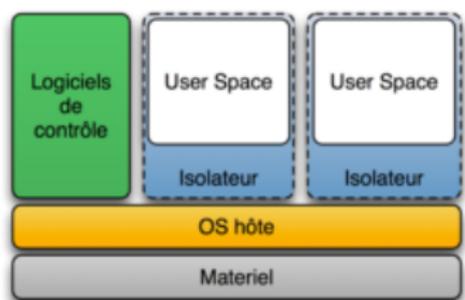
- Un **isolateur** ou **conteneur** permet d'isoler l'exécution d'une application dans un contexte.
- Pas ou très peu de dégradation des performances (même OS)

# Variante 1 de la virtualisation : la conteneurisation



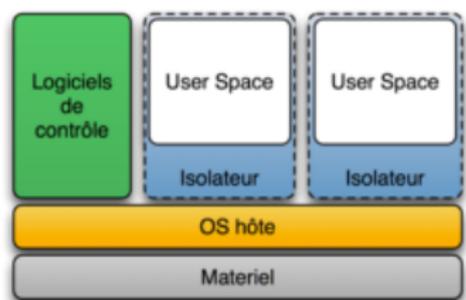
- Un **isolateur** ou **conteneur** permet d'isoler l'exécution d'une application dans un contexte.
- Pas ou très peu de dégradation des performances (même OS)
- Exemples :

# Variante 1 de la virtualisation : la conteneurisation



- Un **isolateur** ou **conteneur** permet d'isoler l'exécution d'une application dans un contexte.
- Pas ou très peu de dégradation des performances (même OS)
- Exemples :
  - ▶ Un environnement virtuel python est un mécanisme d'isolation mais ce n'est pas une isolation totale (ne protège pas contre les accès au fichiers systèmes)

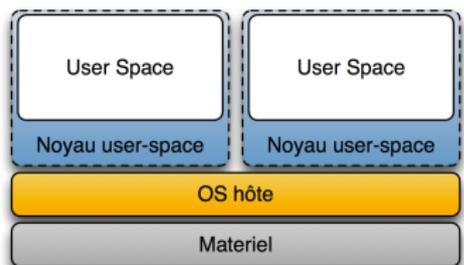
# Variante 1 de la virtualisation : la conteneurisation



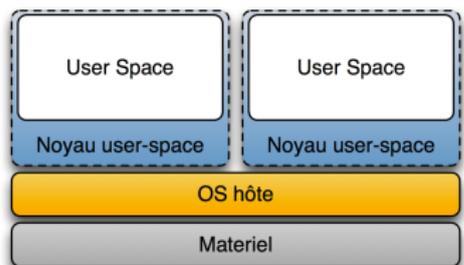
- Un **isolateur** ou **conteneur** permet d'isoler l'exécution d'une application dans un contexte.
- Pas ou très peu de dégradation des performances (même OS)
- Exemples :
  - ▶ Un environnement virtuel python est un mécanisme d'isolation mais ce n'est pas une isolation totale (ne protège pas contre les accès au fichiers systèmes)
  - ▶ Un conteneur Docker (<https://docs.docker.com/>) est un autre exemple célèbre d'isolateur avec une isolation totale. Bibliothèque d'images Docker.

# Variante 2 de la virtualisation : noyau en espace utilisateur

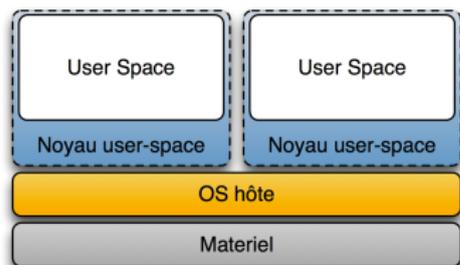
## Variante 2 de la virtualisation : noyau en espace utilisateur



## Variante 2 de la virtualisation : noyau en espace utilisateur

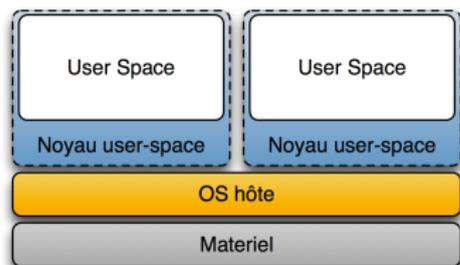


## Variante 2 de la virtualisation : noyau en espace utilisateur



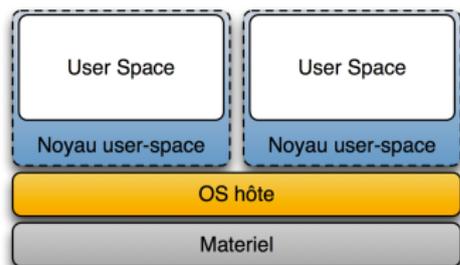
- Un **noyau en espace utilisateur** est une application qui a son propre espace utilisateur dans lequel il contrôle ses applications.

## Variante 2 de la virtualisation : noyau en espace utilisateur



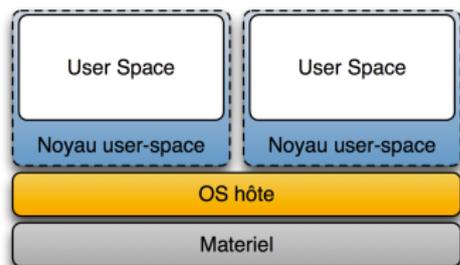
- Un **noyau en espace utilisateur** est une application qui a son propre espace utilisateur dans lequel il contrôle ses applications.
- Peu performant car empilage de deux noyaux.

## Variante 2 de la virtualisation : noyau en espace utilisateur



- Un **noyau en espace utilisateur** est une application qui a son propre espace utilisateur dans lequel il contrôle ses applications.
- Peu performant car empilage de deux noyaux.
- Utilisé pour le développement de noyaux.

## Variante 2 de la virtualisation : noyau en espace utilisateur



- Un **noyau en espace utilisateur** est une application qui a son propre espace utilisateur dans lequel il contrôle ses applications.
- Peu performant car empilage de deux noyaux.
- Utilisé pour le développement de noyaux.
- Exemples : Cooperative Linux (noyau linux avec un hôte windows), ADEOS (linux), ...

# Le cloud computing, c'est ...

Utilisation conjointe de

Virtualisation + Pay as you go + Self Service + API Ouvertes

# Le cloud computing, c'est ...

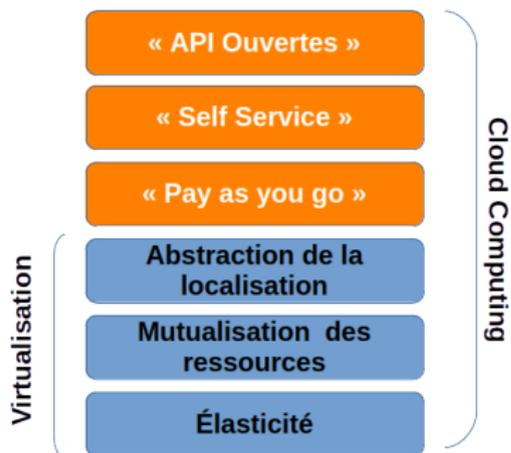
Utilisation conjointe de

Virtualisation + Pay as you go + Self Service + API Ouvertes

# Le cloud computing, c'est ...

Utilisation conjointe de

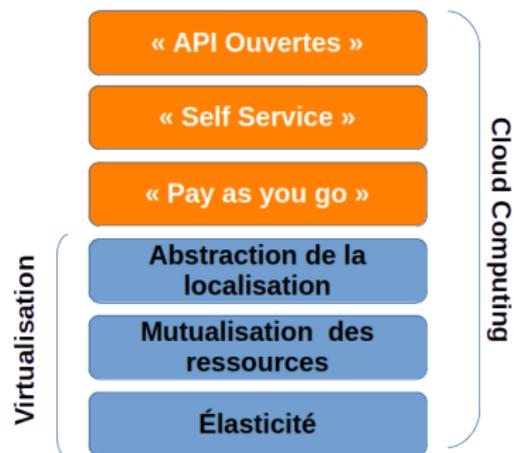
Virtualisation + Pay as you go + Self Service + API Ouvertes



# Le cloud computing, c'est ...

Utilisation conjointe de

Virtualisation + Pay as you go + Self Service + API Ouvertes



# Le cloud computing, c'est ...

Utilisation conjointe de

Virtualisation + Pay as you go + Self Service + API Ouvertes

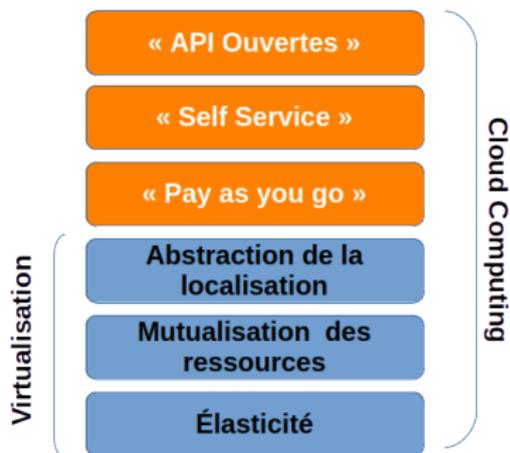


**Pay As you go** On ne paye que ce qu'on utilise en fonction de la consommation réelle.

# Le cloud computing, c'est ...

## Utilisation conjointe de

Virtualisation + Pay as you go + Self Service + API Ouvertes



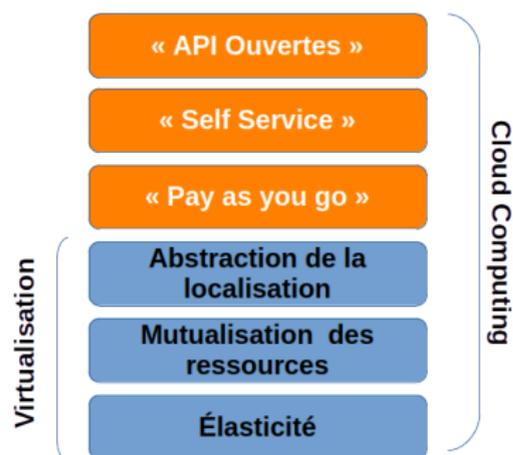
**Pay As you go** On ne paye que ce qu'on utilise en fonction de la consommation réelle.

**Self Service** On demande des ressources via un portail web en quelques minutes.

# Le cloud computing, c'est ...

## Utilisation conjointe de

Virtualisation + Pay as you go + Self Service + API Ouvertes



**Pay As you go** On ne paye que ce qu'on utilise en fonction de la consommation réelle.

**Self Service** On demande des ressources via un portail web en quelques minutes.

**API ouvertes** Interfaces techniques des plateformes cloud pour administrer.

# Les services du cloud computing

## Le service SaaS

### Définition SaaS

**Software as a Service** : location et utilisation d'un logiciel à distance

# Les services du cloud computing

## Le service SaaS

### Définition SaaS

**Software as a Service** : location et utilisation d'un logiciel à distance

- Distinctions avec un logiciel traditionnel :

# Les services du cloud computing

## Le service SaaS

### Définition SaaS

**Software as a Service** : location et utilisation d'un logiciel à distance

- Distinctions avec un logiciel traditionnel :
  - ▶ On n'achète pas, on le loue.

# Les services du cloud computing

## Le service SaaS

### Définition SaaS

**Software as a Service** : location et utilisation d'un logiciel à distance

- Distinctions avec un logiciel traditionnel :
  - ▶ On n'achète pas, on le loue.
  - ▶ On ne l'installe pas, on l'utilise à distance via un navigateur HTML 5

# Les services du cloud computing

## Le service SaaS

### Définition SaaS

**Software as a Service** : location et utilisation d'un logiciel à distance

- Distinctions avec un logiciel traditionnel :
  - ▶ On n'achète pas, on le loue.
  - ▶ On ne l'installe pas, on l'utilise à distance via un navigateur HTML 5
  - ▶ On ne le maintient pas (versions, patch, ...), c'est automatique et transparent.

# Les services du cloud computing

## Le service SaaS

### Définition SaaS

**Software as a Service** : location et utilisation d'un logiciel à distance

- Distinctions avec un logiciel traditionnel :
  - ▶ On n'achète pas, on le loue.
  - ▶ On ne l'installe pas, on l'utilise à distance via un navigateur HTML 5
  - ▶ On ne le maintient pas (versions, patch, ...), c'est automatique et transparent.
- Exemples :

# Les services du cloud computing

## Le service SaaS

### Définition SaaS

**Software as a Service** : location et utilisation d'un logiciel à distance

- Distinctions avec un logiciel traditionnel :
  - ▶ On n'achète pas, on le loue.
  - ▶ On ne l'installe pas, on l'utilise à distance via un navigateur HTML 5
  - ▶ On ne le maintient pas (versions, patch, . . .), c'est automatique et transparent.
- Exemples :
  - ▶ Gmail est une application SaaS "gratuite".

# Les services du cloud computing

## Le service SaaS

### Définition SaaS

**Software as a Service** : location et utilisation d'un logiciel à distance

- Distinctions avec un logiciel traditionnel :
  - ▶ On n'achète pas, on le loue.
  - ▶ On ne l'installe pas, on l'utilise à distance via un navigateur HTML 5
  - ▶ On ne le maintient pas (versions, patch, . . .), c'est automatique et transparent.
- Exemples :
  - ▶ Gmail est une application SaaS "gratuite".
  - ▶ Wordpress est une application de création de sites webs (de gratuit à payant selon les fonctionnalités). On peut aussi télécharger une version et le déployer sur un cloud.

# Les services du cloud computing

## Le service PaaS

### Définition PaaS

**Platform as a Service** : location d'une plateforme technique permettant l'exécution d'un code déployé pour cette plateforme

# Les services du cloud computing

## Le service PaaS

### Définition PaaS

**Platform as a Service** : location d'une plateforme technique permettant l'exécution d'un code déployé pour cette plateforme

- Les PaaS s'adressent aux **développeurs**

# Les services du cloud computing

## Le service PaaS

### Définition PaaS

**Platform as a Service** : location d'une plateforme technique permettant l'exécution d'un code déployé pour cette plateforme

- Les PaaS s'adressent aux **développeurs**
- PaaS comprend l'infrastructure, les serveurs, le stockage et les composants réseau, les IDE, les SGBD, les outils de tests, etc.

# Les services du cloud computing

## Le service PaaS

### Définition PaaS

**Platform as a Service** : location d'une plateforme technique permettant l'exécution d'un code déployé pour cette plateforme

- Les PaaS s'adressent aux **développeurs**
- PaaS comprend l'infrastructure, les serveurs, le stockage et les composants réseau, les IDE, les SGBD, les outils de tests, etc.
- Le cycle de vie complet de l'application web : conception, tests, déploiement, est prise en charge par le PaaS.

# Les services du cloud computing

## Le service PaaS

### Définition PaaS

**Platform as a Service** : location d'une plateforme technique permettant l'exécution d'un code déployé pour cette plateforme

- Les PaaS s'adressent aux **développeurs**
- PaaS comprend l'infrastructure, les serveurs, le stockage et les composants réseau, les IDE, les SGBD, les outils de tests, etc.
- Le cycle de vie complet de l'application web : conception, tests, déploiement, est prise en charge par le PaaS.
- Le PaaS occulte les dépenses liées au coût des serveurs, des licences logicielles.

# Les services du cloud computing

## Le service PaaS

### Définition PaaS

**Platform as a Service** : location d'une plateforme technique permettant l'exécution d'un code déployé pour cette plateforme

- Les PaaS s'adressent aux **développeurs**
- PaaS comprend l'infrastructure, les serveurs, le stockage et les composants réseau, les IDE, les SGBD, les outils de tests, etc.
- Le cycle de vie complet de l'application web : conception, tests, déploiement, est prise en charge par le PaaS.
- Le PaaS occulte les dépenses liées au coût des serveurs, des licences logicielles.
- Mais ce n'est pas gratuit pour autant. Exemples : Google App Engine, Microsoft Azure

# Les services du cloud computing

## Le service PaaS

### Définition PaaS

**Platform as a Service** : location d'une plateforme technique permettant l'exécution d'un code déployé pour cette plateforme

- Les PaaS s'adressent aux **développeurs**
- PaaS comprend l'infrastructure, les serveurs, le stockage et les composants réseau, les IDE, les SGBD, les outils de tests, etc.
- Le cycle de vie complet de l'application web : conception, tests, déploiement, est prise en charge par le PaaS.
- Le PaaS occulte les dépenses liées au coût des serveurs, des licences logicielles.
- Mais ce n'est pas gratuit pour autant. Exemples : Google App Engine, Microsoft Azure
- Illustration : j'ai besoin d'un serveur JEE wildfly version 26 avec un serveur de bases de données Postgres version 14, d'un IDE Java Entreprise et d'un outil de tests JUnit.

# Les services du cloud computing

## Le service IaaS

### Définition IaaS

Infrastructure as a **S**ervice : location d'une plateforme technique permettant l'exécution d'architectures applicatives complètes (SGBD, serveur d'applications, etc)

# Les services du cloud computing

## Le service IaaS

### Définition IaaS

Infrastructure as a **S**ervice : location d'une plateforme technique permettant l'exécution d'architectures applicatives complètes (SGBD, serveur d'applications, etc)

- Les IaaS s'adressent aux **ingénieurs DevOps**

# Les services du cloud computing

## Le service IaaS

### Définition IaaS

Infrastructure as a **S**ervice : location d'une plateforme technique permettant l'exécution d'architectures applicatives complètes (SGBD, serveur d'applications, etc)

- Les IaaS s'adressent aux **ingénieurs DevOps**
- Exemples : AWS EC2, Google Compute Engine

# Les services du cloud computing

## Le service IaaS

### Définition IaaS

Infrastructure as a **S**ervice : location d'une plateforme technique permettant l'exécution d'architectures applicatives complètes (SGBD, serveur d'applications, etc)

- Les IaaS s'adressent aux **ingénieurs DevOps**
- Exemples : AWS EC2, Google Compute Engine
- Illustration :

# Les services du cloud computing

## Le service IaaS

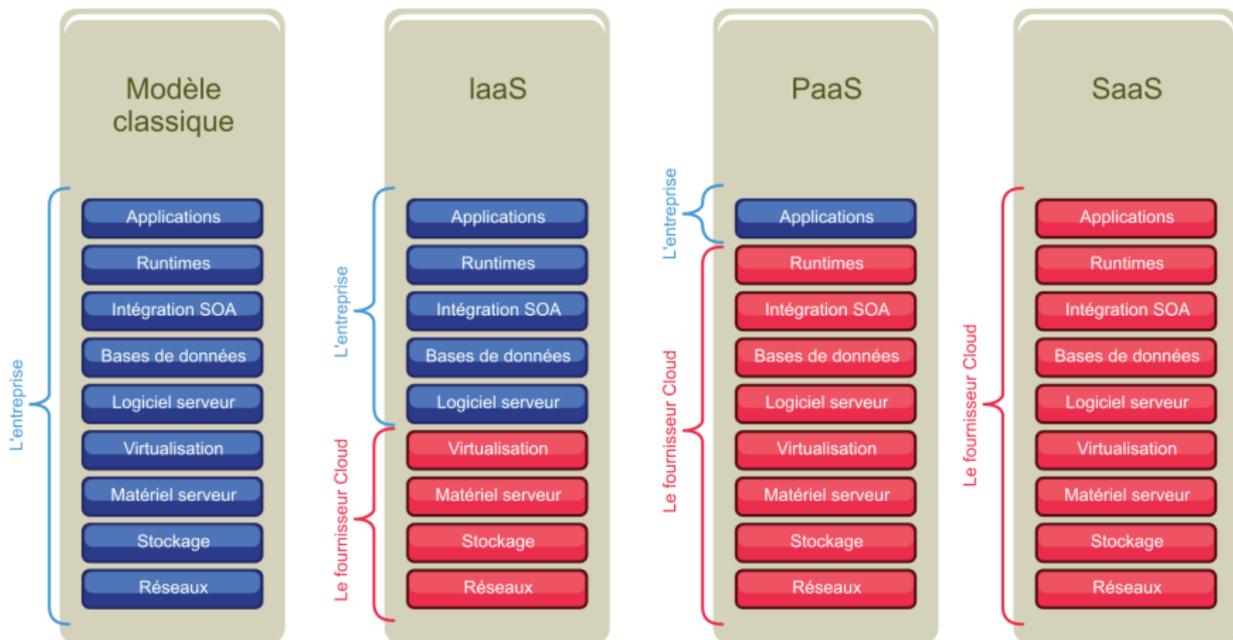
### Définition IaaS

Infrastructure as a **S**ervice : location d'une plateforme technique permettant l'exécution d'architectures applicatives complètes (SGBD, serveur d'applications, etc)

- Les IaaS s'adressent aux **ingénieurs DevOps**
- Exemples : AWS EC2, Google Compute Engine
- Illustration :
  - ▶ J'ai besoin d'une distribution linux Ubuntu version 22.04

# Les services du cloud computing

En résumé



# Les types de Cloud

- Cloud public

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud
- Cloud privé

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud
- Cloud privé
  - ▶ Dédié à une seule organisation

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud
- Cloud privé
  - ▶ Dédié à une seule organisation
  - ▶ Plus de contrôle et de sécurité

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud
- Cloud privé
  - ▶ Dédié à une seule organisation
  - ▶ Plus de contrôle et de sécurité
  - ▶ Exemple : OpenStack

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud
- Cloud privé
  - ▶ Dédié à une seule organisation
  - ▶ Plus de contrôle et de sécurité
  - ▶ Exemple : OpenStack
- Cloud hybride

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud
- Cloud privé
  - ▶ Dédié à une seule organisation
  - ▶ Plus de contrôle et de sécurité
  - ▶ Exemple : OpenStack
- Cloud hybride
  - ▶ Combinaison de cloud privé et public

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud
- Cloud privé
  - ▶ Dédié à une seule organisation
  - ▶ Plus de contrôle et de sécurité
  - ▶ Exemple : OpenStack
- Cloud hybride
  - ▶ Combinaison de cloud privé et public
  - ▶ Flexibilité accrue

# Les types de Cloud

- Cloud public
  - ▶ Hébergé par un fournisseur tiers
  - ▶ Accessible via Internet
  - ▶ Exemples : AWS, Azure, Google Cloud, OVHCloud
- Cloud privé
  - ▶ Dédié à une seule organisation
  - ▶ Plus de contrôle et de sécurité
  - ▶ Exemple : OpenStack
- Cloud hybride
  - ▶ Combinaison de cloud privé et public
  - ▶ Flexibilité accrue
  - ▶ Exemples : VMware Cloud, AWS Outposts

# Alors Cloud ou pas Cloud ?

## Cas 1 : "Pas cloud" : mode client déployé

- **Avantage** : je ne dépend pas d'un tiers pour faire tourner mes applications.

# Alors Cloud ou pas Cloud ?

## Cas 1 : "Pas cloud" : mode client déployé

- **Avantage** : je ne dépend pas d'un tiers pour faire tourner mes applications.
- **Inconvénients** :

# Alors Cloud ou pas Cloud ?

## Cas 1 : "Pas cloud" : mode client déployé

- **Avantage** : je ne dépend pas d'un tiers pour faire tourner mes applications.
- **Inconvénients** :
  - ▶ Coût du logiciel à l'achat (sauf si Open Source gratuit), coût de la maintenance (soit externe, soit interne)

# Alors Cloud ou pas Cloud ?

## Cas 1 : "Pas cloud" : mode client déployé

- **Avantage** : je ne dépend pas d'un tiers pour faire tourner mes applications.
- **Inconvénients** :
  - ▶ Coût du logiciel à l'achat (sauf si Open Source gratuit), coût de la maintenance (soit externe, soit interne)
  - ▶ Coût du matériel (serveurs + ordinateurs clients performants) pour déployer et exécuter les logiciels, coût de l'installation et maintenance des logiciels

# Alors Cloud ou pas Cloud ?

## Cas 1 : "Pas cloud" : mode client déployé

- **Avantage** : je ne dépend pas d'un tiers pour faire tourner mes applications.
- **Inconvénients** :
  - ▶ Coût du logiciel à l'achat (sauf si Open Source gratuit), coût de la maintenance (soit externe, soit interne)
  - ▶ Coût du matériel (serveurs + ordinateurs clients performants) pour déployer et exécuter les logiciels, coût de l'installation et maintenance des logiciels
  - ▶ Si bug, latence importante pour obtenir la version qui corrige le bug.

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :
  - ▶ je ne dépend pas d'un tiers pour faire tourner mes applications.

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :
  - ▶ je ne dépend pas d'un tiers pour faire tourner mes applications.
  - ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance) et annule le coût de déploiement/maintenance de ces ordinateurs clients.

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :
  - ▶ je ne dépend pas d'un tiers pour faire tourner mes applications.
  - ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance) et annule le coût de déploiement/maintenance de ces ordinateurs clients.
  - ▶ J'augmente l'élasticité

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :
  - ▶ je ne dépend pas d'un tiers pour faire tourner mes applications.
  - ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance) et annule le coût de déploiement/maintenance de ces ordinateurs clients.
  - ▶ J'augmente l'élasticité
- Inconvénients :

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :

- ▶ je ne dépend pas d'un tiers pour faire tourner mes applications.
- ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance) et annule le coût de déploiement/maintenance de ces ordinateurs clients.
- ▶ J'augmente l'élasticité

- Inconvénients :

- ▶ Coût du logiciel à l'achat (sauf si Open Source gratuit), coût de la maintenance (soit externe, soit interne)

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :

- ▶ je ne dépend pas d'un tiers pour faire tourner mes applications.
- ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance) et annule le coût de déploiement/maintenance de ces ordinateurs clients.
- ▶ J'augmente l'élasticité

- Inconvénients :

- ▶ Coût du logiciel à l'achat (sauf si Open Source gratuit), coût de la maintenance (soit externe, soit interne)
- ▶ Coût des serveurs

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :
  - ▶ je ne dépend pas d'un tiers pour faire tourner mes applications.
  - ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance) et annule le coût de déploiement/maintenance de ces ordinateurs clients.
  - ▶ J'augmente l'élasticité
- Inconvénients :
  - ▶ Coût du logiciel à l'achat (sauf si Open Source gratuit), coût de la maintenance (soit externe, soit interne)
  - ▶ Coût des serveurs
  - ▶ Si bug, latence importante pour obtenir la version qui corrige le bug.

# Alors Cloud ou pas Cloud ?

## Cas 2 : mode cloud privé interne

- Avantages :

- ▶ je ne dépend pas d'un tiers pour faire tourner mes applications.
- ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance) et annule le coût de déploiement/maintenance de ces ordinateurs clients.
- ▶ J'augmente l'élasticité

- Inconvénients :

- ▶ Coût du logiciel à l'achat (sauf si Open Source gratuit), coût de la maintenance (soit externe, soit interne)
- ▶ Coût des serveurs
- ▶ Si bug, latence importante pour obtenir la version qui corrige le bug.
- ▶ L'élasticité n'est pas totale : difficile d'acheter de nouveaux serveurs en cas de pic d'utilisation

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :
  - ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :
  - ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)
  - ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance)

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :

- ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)
- ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance)
- ▶ Je n'ai plus à maintenir le logiciel et le matériel (sauf stations clientes)

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :

- ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)
- ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance)
- ▶ Je n'ai plus à maintenir le logiciel et le matériel (sauf stations clientes)
- ▶ Les logiciels évoluent en permanence de façon transparente

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :

- ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)
- ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance)
- ▶ Je n'ai plus à maintenir le logiciel et le matériel (sauf stations clientes)
- ▶ Les logiciels évoluent en permanence de façon transparente
- ▶ J'ai une élasticité totale, je ne paye que ce que j'utilise

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :
  - ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)
  - ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance)
  - ▶ Je n'ai plus à maintenir le logiciel et le matériel (sauf stations clientes)
  - ▶ Les logiciels évoluent en permanence de façon transparente
  - ▶ J'ai une élasticité totale, je ne paye que ce que j'utilise
- Inconvénients :

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :

- ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)
- ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance)
- ▶ Je n'ai plus à maintenir le logiciel et le matériel (sauf stations clientes)
- ▶ Les logiciels évoluent en permanence de façon transparente
- ▶ J'ai une élasticité totale, je ne paye que ce que j'utilise

- Inconvénients :

- ▶ Je dépends d'un tiers

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :
  - ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)
  - ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance)
  - ▶ Je n'ai plus à maintenir le logiciel et le matériel (sauf stations clientes)
  - ▶ Les logiciels évoluent en permanence de façon transparente
  - ▶ J'ai une élasticité totale, je ne paye que ce que j'utilise
- Inconvénients :
  - ▶ Je dépends d'un tiers
  - ▶ Selon, mon activité, le cloud externe est interdit (ex : banques)

# Alors Cloud ou pas Cloud ?

## Cas 3 : mode cloud externe

- Avantages :

- ▶ Je diminue les coûts logiciels et matériels (les hébergeurs disposent de réductions car achat de serveurs à grande échelle)
- ▶ Je diminue le coût des ordinateurs clients (besoin de moins de puissance)
- ▶ Je n'ai plus à maintenir le logiciel et le matériel (sauf stations clientes)
- ▶ Les logiciels évoluent en permanence de façon transparente
- ▶ J'ai une élasticité totale, je ne paye que ce que j'utilise

- Inconvénients :

- ▶ Je dépends d'un tiers
- ▶ Selon, mon activité, le cloud externe est interdit (ex : banques)
- ▶ Cas sensible de législation (voir plus loin)

# Cloud : un cas typique

Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...

# Cloud : un cas typique

Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous

# Cloud : un cas typique

## Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous
- Je ne sais pas si cela va marcher ou pas et, si oui, à quel niveau ?

# Cloud : un cas typique

## Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous
- Je ne sais pas si cela va marcher ou pas et, si oui, à quel niveau ?
- Sans le cloud :

# Cloud : un cas typique

## Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous
- Je ne sais pas si cela va marcher ou pas et, si oui, à quel niveau ?
- Sans le cloud :
  - ▶ Je dois trouver/convaincre des investisseurs

# Cloud : un cas typique

## Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous
- Je ne sais pas si cela va marcher ou pas et, si oui, à quel niveau ?
- Sans le cloud :
  - ▶ Je dois trouver/convaincre des investisseurs
  - ▶ J'ai du mal à quantifier la charge des serveurs, la bonne architecture logicielle (ex : "Facebook")

# Cloud : un cas typique

## Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous
- Je ne sais pas si cela va marcher ou pas et, si oui, à quel niveau ?
- Sans le cloud :
  - ▶ Je dois trouver/convaincre des investisseurs
  - ▶ J'ai du mal à quantifier la charge des serveurs, la bonne architecture logicielle (ex : "Facebook")
- Avec le cloud :

# Cloud : un cas typique

## Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous
- Je ne sais pas si cela va marcher ou pas et, si oui, à quel niveau ?
- Sans le cloud :
  - ▶ Je dois trouver/convaincre des investisseurs
  - ▶ J'ai du mal à quantifier la charge des serveurs, la bonne architecture logicielle (ex : "Facebook")
- Avec le cloud :
  - ▶ J'ai besoin d'un fond de départ moins important ("pay as you go")

# Cloud : un cas typique

## Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous
- Je ne sais pas si cela va marcher ou pas et, si oui, à quel niveau ?
- Sans le cloud :
  - ▶ Je dois trouver/convaincre des investisseurs
  - ▶ J'ai du mal à quantifier la charge des serveurs, la bonne architecture logicielle (ex : "Facebook")
- Avec le cloud :
  - ▶ J'ai besoin d'un fond de départ moins important ("pay as you go")
  - ▶ Je peux adapter mon architecture en fonction de la demande.

# Cloud : un cas typique

## Je crée une startup

- J'ai une super idée de logiciel ("blablacar", "le bon coin", ...) mais...
- J'ai pas de sous
- Je ne sais pas si cela va marcher ou pas et, si oui, à quel niveau ?
- Sans le cloud :
  - ▶ Je dois trouver/convaincre des investisseurs
  - ▶ J'ai du mal à quantifier la charge des serveurs, la bonne architecture logicielle (ex : "Facebook")
- Avec le cloud :
  - ▶ J'ai besoin d'un fond de départ moins important ("pay as you go")
  - ▶ Je peux adapter mon architecture en fonction de la demande.
  - ▶ Je peux arrêter les frais plus facilement en cas d'échec.

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)
  - ▶ Avec le cloud, on ne sait pas où sont véritablement stockées les données, c'est la loi du pays hébergeur qui prédomine

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)
  - ▶ Avec le cloud, on ne sait pas où sont véritablement stockées les données, c'est la loi du pays hébergeur qui prédomine
  - ▶ Exemple : Le **Patriot Act** est une loi qui permet à l'administration américaine de demander l'ouverture des données à toute société ayant son siège aux US (cf Edward Snowden, espionnage NSA).

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)
  - ▶ Avec le cloud, on ne sait pas où sont véritablement stockées les données, c'est la loi du pays hébergeur qui prédomine
  - ▶ Exemple : Le **Patriot Act** est une loi qui permet à l'administration américaine de demander l'ouverture des données à toute société ayant son siège aux US (cf Edward Snowden, espionnage NSA).
  - ▶ Exemple : le gouvernement chinois a exigé le filtrage de données aux sociétés Google, Microsoft et Yahoo qui ont obtempéré (cf Amnesty International 2006)

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)
  - ▶ Avec le cloud, on ne sait pas où sont véritablement stockées les données, c'est la loi du pays hébergeur qui prédomine
  - ▶ Exemple : Le **Patriot Act** est une loi qui permet à l'administration américaine de demander l'ouverture des données à toute société ayant son siège aux US (cf Edward Snowden, espionnage NSA).
  - ▶ Exemple : le gouvernement chinois a exigé le filtrage de données aux sociétés Google, Microsoft et Yahoo qui ont obtempéré (cf Amnesty International 2006)
  - ▶ S'assurer que les acteurs du cloud respectent des certifications de sécurité (ex : ISO 27001)

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)
  - ▶ Avec le cloud, on ne sait pas où sont véritablement stockées les données, c'est la loi du pays hébergeur qui prédomine
  - ▶ Exemple : Le **Patriot Act** est une loi qui permet à l'administration américaine de demander l'ouverture des données à toute société ayant son siège aux US (cf Edward Snowden, espionnage NSA).
  - ▶ Exemple : le gouvernement chinois a exigé le filtrage de données aux sociétés Google, Microsoft et Yahoo qui ont obtempéré (cf Amnesty International 2006)
  - ▶ S'assurer que les acteurs du cloud respectent des certifications de sécurité (ex : ISO 27001)
- Respecter l'**intégrité des données** lors d'une altération accidentelle ou mal intentionnée.

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)
  - ▶ Avec le cloud, on ne sait pas où sont véritablement stockées les données, c'est la loi du pays hébergeur qui prédomine
  - ▶ Exemple : Le **Patriot Act** est une loi qui permet à l'administration américaine de demander l'ouverture des données à toute société ayant son siège aux US (cf Edward Snowden, espionnage NSA).
  - ▶ Exemple : le gouvernement chinois a exigé le filtrage de données aux sociétés Google, Microsoft et Yahoo qui ont obtempéré (cf Amnesty International 2006)
  - ▶ S'assurer que les acteurs du cloud respectent des certifications de sécurité (ex : ISO 27001)
- Respecter l'**intégrité des données** lors d'une altération accidentelle ou mal intentionnée.
  - ▶ Avec le cloud, c'est plus facile de disposer de datacenters suffisamment éloignés (ex : tremblement de terre)

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)
  - ▶ Avec le cloud, on ne sait pas où sont véritablement stockées les données, c'est la loi du pays hébergeur qui prédomine
  - ▶ Exemple : Le **Patriot Act** est une loi qui permet à l'administration américaine de demander l'ouverture des données à toute société ayant son siège aux US (cf Edward Snowden, espionnage NSA).
  - ▶ Exemple : le gouvernement chinois a exigé le filtrage de données aux sociétés Google, Microsoft et Yahoo qui ont obtempéré (cf Amnesty International 2006)
  - ▶ S'assurer que les acteurs du cloud respectent des certifications de sécurité (ex : ISO 27001)
- Respecter l'**intégrité des données** lors d'une altération accidentelle ou mal intentionnée.
  - ▶ Avec le cloud, c'est plus facile de disposer de datacenters suffisamment éloignés (ex : tremblement de terre)
- **Disponibilité** des données :

# Cloud computing et la sécurité des données

- Respecter les **réglementations** des secteurs d'activité (ex : secteur bancaire interdit le cloud)
- Respecter/protéger les **données personnelles** (RGPD)
  - ▶ Avec le cloud, on ne sait pas où sont véritablement stockées les données, c'est la loi du pays hébergeur qui prédomine
  - ▶ Exemple : Le **Patriot Act** est une loi qui permet à l'administration américaine de demander l'ouverture des données à toute société ayant son siège aux US (cf Edward Snowden, espionnage NSA).
  - ▶ Exemple : le gouvernement chinois a exigé le filtrage de données aux sociétés Google, Microsoft et Yahoo qui ont obtempéré (cf Amnesty International 2006)
  - ▶ S'assurer que les acteurs du cloud respectent des certifications de sécurité (ex : ISO 27001)
- Respecter l'**intégrité des données** lors d'une altération accidentelle ou mal intentionnée.
  - ▶ Avec le cloud, c'est plus facile de disposer de datacenters suffisamment éloignés (ex : tremblement de terre)
- **Disponibilité** des données :
  - ▶ En cas de défaillance d'un serveur, disposer rapidement d'un nouveau serveur est facile et rapide

# Conclusion

- La décision d'opter ou pas pour le cloud computing est devenu incontournable.

# Conclusion

- La décision d'opter ou pas pour le cloud computing est devenu incontournable.
- Nécessite de prendre en compte plusieurs paramètres (sécurité, innovation, réglementations, coûts, ressources humaines, etc)

# Conclusion

- La décision d'opter ou pas pour le cloud computing est devenu incontournable.
- Nécessite de prendre en compte plusieurs paramètres (sécurité, innovation, réglementations, coûts, ressources humaines, etc)
- Un nouveau paramètre : l'impact écologique (et pour cela aussi, le choix n'est pas évident)

# Conclusion

- La décision d'opter ou pas pour le cloud computing est devenu incontournable.
- Nécessite de prendre en compte plusieurs paramètres (sécurité, innovation, réglementations, coûts, ressources humaines, etc)
- Un nouveau paramètre : l'impact écologique (et pour cela aussi, le choix n'est pas évident)
- Dépend aussi du sentiment de confiance

# Conclusion

- La décision d'opter ou pas pour le cloud computing est devenu incontournable.
- Nécessite de prendre en compte plusieurs paramètres (sécurité, innovation, réglementations, coûts, ressources humaines, etc)
- Un nouveau paramètre : l'impact écologique (et pour cela aussi, le choix n'est pas évident)
- Dépend aussi du sentiment de confiance
- Le plus souvent, la décision se tourne vers un mode hybride.

## Edward Snowden

Je prends conscience aujourd'hui que pour mon gouvernement, j'étais un homme transparent. Le portable qui me permettait de m'orienter et me corrigeait quand je me trompais de direction, qui me traduisait les panneaux indicateurs et me donnait les horaires des bus et des trains, veillait également à ce que mes patrons connaissent mes moindres faits et gestes. Mon téléphone leur indiquait quand je m'étais trouvé à tel ou tel endroit sans même que j'ai eu besoin de le toucher ou de le sortir de ma poche.