



principes architecturaux du "cloud computing"

PaaS



Google App Engine

Google fournit des solutions SaaS et PaaS dans le cloud

Co;puting . Quelques exemples de solutions SaaS, y compris

Google Apps, compris Gmail, Doc, etc. et PaaS

inclut Google App Engine.



Google App Engine

Dans l'espace Platform as a Service (**PaaS**), Google est un
Joueur important . App Engine est une plate-forme pour créer, stocker et
exécuter des applications sur les serveurs de Google en utilisant les
Langues de programmation comme java et python. App Engine inclut des
outils pour gérer le data store, surveiller le site et sa ressource consommées
et le débogage et la journalisation. Un utilisateur peut servir
l'application de son propre nom de domaine (tel qu



Google App Engine

<http://www.example.com/>) à l'aide de Google Apps. Ou, il peut

servir son application en utilisant un nom libre sur le domaine appspot.com.

Un utilisateur peut partager son application avec le monde, ou limiter

accès aux membres de l'organisation. App Engine ne coûte rien

pour commencer. Toutes les applications peuvent utiliser jusqu'à 1 Go de stockage et assez de CPU et de bande passante pour supporter une application efficace desservant environ 5 millions de pages vues par mois, absolument gratuit.

Les applications nécessitant plus de stockage ou de bande passante peuvent

achat qui est divisé en cinq seaux: temps CPU,

la bande passante, la bande passante, le stockage et les e-mails sortants.



Google App Engine: architecture

L'architecture de Google App Engine fournit une nouvelle approche sans traiter les serveurs Web et équilibres de charge, mais au lieu de déployer le applications sur le cloud Google App Engine en fournissant accès aux instances et évolutivité



Google App Engine: architecture

le kit de développement logiciel (SDK) Google App Engine fournit des langages de programmation Java et Python. les langues ont leur propre application de serveur web qui contient tous les services Google App Engine sur un ordinateur local.

Le serveur simule également un environnement sécurisé. le

Le Google App Engine SDK comporte des API et des bibliothèques, y compris

outils pour télécharger des applications. L'architecture définit le

structure des applications exécutées sur Google App Engine



Google App Engine: architecture

Python: Google App Engine permet la mise en œuvre des applications utilisant le langage de programmation Python et l'exécute eux sur son interprète. Le moteur Google App Engine est riche en terme des API et outils pour la conception d'applications Web, la modélisation de données, gestion, accès aux données des applications, prise en charge des bibliothèques matures

Les principales caractéristiques de Google App Engine sont ses DataStore, fichier de configuration app.yaml et comment il sert une application.



Google App Engine: architecture

Bigtable: Le DataStore est construit au-dessus de Bigtable qui est construit sur le système de fichiers Google. Le DataStore est le seul base de données que Google App Engine prend en charge pour la journalisation et le stockage données, y compris les données de session. Il utilise légèrement différent terminologie héritée du BigTable. Le Bigtable peut être défini comme un énorme tableur avec un nombre illimité de colonnes et sous la forme d'une chaîne. Le BigTable est un tableau Base de données NoSQL conçue pour évoluer de manière fiable des pétaoctets de données et des milliers de machines. C'est un sparse, persistante distribuée, multidimensionnelle map de stockage . C'est généralement appelé "map" indexée avec la clé de ligne, la clé de colonne et un horodatage. ⁸



Google App Engine: architecture

Un "map" est un type de données abstrait composé de collection de clés, et une collection de valeurs où chaque clé est associé à une valeur. Le Bigtable a un serveur maître qui coordonne les grands segments d'une table logique appelée "tablets". Les tablets sont répartis sur une rangée avec une taille optimale de 200 Mo pour chaque tablette à des fins d'optimisation. la table contient des lignes et des colonnes et chaque cellule a un time stamp. Google App Engine permet l'utilisation du BigTable dans les applications via l'API DataStore

Google App Engine: architecture

L'API DataStore: le DataStore est responsable de la l'évolutivité des applications Google App Engine. la structure des applications leur permet de distribuer les demandes à travers les serveurs qui devraient être compromises

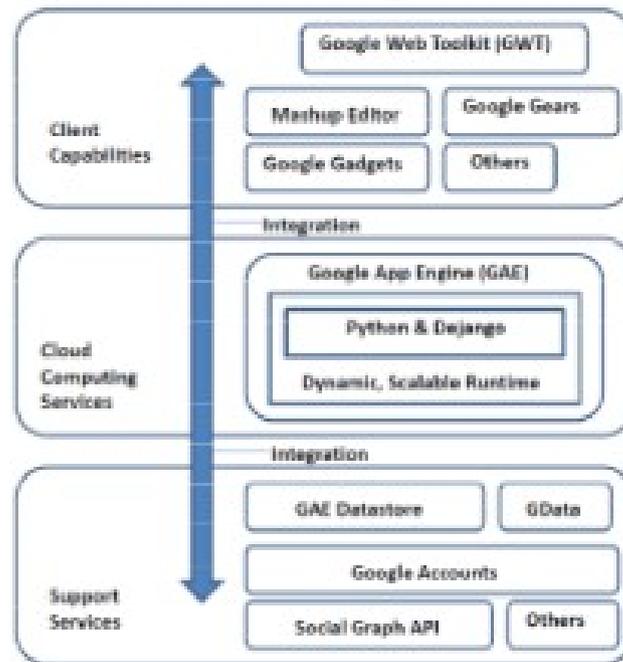


Fig. 1. Architecture of Google App Engine

Google App Engine: architecture

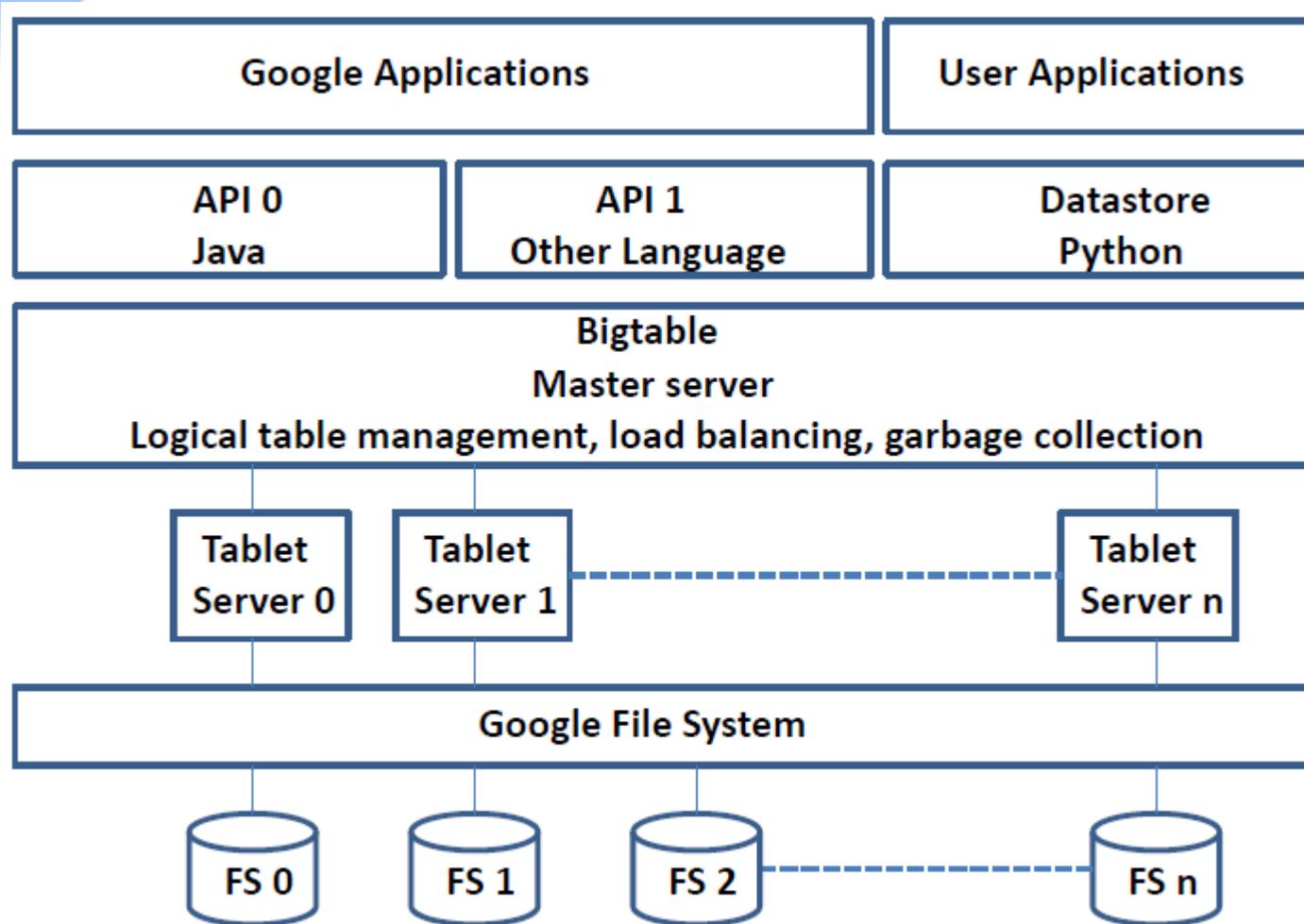


Fig. 2. Google DataStore Architecture



Google App Engine: architecture

avec des bases de données relationnelles. Contrairement à toute base de données relationnelle, Le DataStore Google App Engine peut créer un nombre infini des lignes et des colonnes qui passe à l'échelles en ajoutant des serveurs aux clusters. Dans le DataStore, les tables sont appelées "modèles" et sont représenté dans les classes. Les enregistrements sont appelés "entités" et sont les instances du modèle; les colonnes sont appelées "propriétés" et sont des attributs de modèles ou d'entités. Pour accéder au DataStore nous devons définir une classe de modèle avec des propriétés, puis créer des entités et les stocker dans la base de données. Les requêtes ultérieures peuvent être exécuter pour récupérer les entités. La classe de modèle peut être créée par sous-classe `db.model`.



Google App Engine: architecture

Google App Engine fournit une variété de types de propriétés de chaînes et entiers à Boolean, objets date / heure, liste d'objets, numéros de téléphone, les adresses email, les points géographiques comme latitude, la longitude, etc.

Google App Engine permet d'effectuer des requêtes à l'aide de Bigtable en tant que base de données par ses services utilisant Google App Engine Query Language (GQL)



Google App Engine: architecture

Toutes les données sont stockées dans le cloud

qui pourrait être à n'importe quel endroit sur les serveurs de Google. Si les données sont à stocker sur une base de données externe installée localement sur notre machine Google impose des contraintes strictes en raison de

problèmes de sécurité qui peuvent être un problème potentiel si

les organisations mettent leurs données sécurisées sur des serveurs situés dans emplacements éloignés. La figure 3 montre l'Architecture de tout les DataStore

Google App Engine: architecture

Fichier de configuration "app.yaml": Le fichier app.yaml est un

plate-forme neutre et fichier lisible par l'homme pour représenter

Les données. Il est créé comme une alternative à XML pour représenter

structures dans les langages de programmation comme des listes et

dictionnaires La syntaxe "Key: value" représente les éléments d'un

dictionary et "-" représentent des éléments dans une liste. Le fichier

représente un dictionnaire avec 5 éléments clés. Elles sont

application, version, runtime, api_version et gestionnaires. le

La structure du fichier app.yaml . La première

clé est l'application, il peut être n'importe quel nom lorsqu'il est exécuté

sur un serveur local

Google App Engine: architecture

Mais s'il est téléchargé sur le serveur de Google, alors La valeur de l'application clé doit correspondre à la valeur de l'ID d'application. la deuxième clé est "version" qui est utilisée pour spécifier la version nombre d'applications. Google utilise "MAJOR.MINOR" format pour représenter les numéros de demande. La version MAJOR est le nombre de jeux d'utilisateurs et la version MINOR est le nième téléchargement de cette version. Google App Engine enregistre le dernier téléchargement pour chaque MAJOR version, et l'un d'eux peut être choisi comme l'actuel. Pour les troisième et quatrième clés d'exécution et api_version sont spécifiés en tant que Python. Nouvelles versions de l'API sera disponible dans le futur. Les gestionnaires spécifient le mappage de l'URL modèles Les gestionnaires sont des valeurs clés différentes qui peuvent être un fichier statique, fichier de script ou un répertoire statique.



Google App Engine: architecture

Comment App Engine sert les applications. Chaque application a un fichier `app.yaml` qui indique comment gérer les demandes d'URL. Google App Engine fournit un cadre simple appelé `webapp` qui aide à organiser le code. Quand un navigateur web envoie une requête au cloud de Google, il choisit un serveur à proximité de l'emplacement des utilisateurs, instancie l'application si elle n'est pas en cours d'exécution et traite la demande des utilisateurs. Par conséquent, le nuage répond aux demandes en créant les instances si nécessaire et les supprime lorsqu'ils ne sont pas utilisés.



Google App Engine: architecture

Java: Google App Engine fournit des outils et des API requis pour le développement d'applications Web fonctionnant sur l'heure d'exécution Java de Google App Engine. L'application interagit avec l'environnement en utilisant des servlets et web technologies telles que Java Server Pages (JSP) qui peuvent être développé en utilisant Java6. L'environnement GAE utilise Java SE Runtime JRE plateforme 6 et les bibliothèques dont les applications peut accéder en utilisant des API.

Google App Engine: architecture

Java: Google App Engine fournit des outils et des API

requis pour le développement d'applications Web fonctionnant sur l'heure d'exécution Java de Google App Engine. L'application interagit avec l'environnement en utilisant des servlets et web technologies telles que Java Server Pages (JSP) qui peuvent être développé en utilisant Java6. L'environnement GAE utilise Java SE Runtime JRE plateforme 6 et les bibliothèques dont les applications peut accéder en utilisant des API.



Java SDK a des implémentations pour Java Data Objects (JDO) et Java Persistence (JPA)

interfaces. Pour échanger des e-mails avec Google App Moteur, il fournit le service de messagerie Google App Engine via l'API Java Mail. Support pour d'autres langages comme JavaScript, Ruby ou Scala est également fourni par Google App Moteur avec l'utilisation de compilateurs compatibles JVM et interpréteurs. Lorsque Google App Engine reçoit une demande Web cela correspond à l'URL mentionnée dans les applications descripteur de déploiement (c'est-à-dire, fichier web.xml dans WEB-INF

directory) il appelle une servlet correspondant à cette requête et utilise Java API Servlets pour fournir des données demandées et accepte les données de réponse.

Google App Engine

Google App Engine facilite la création d'une application

qui fonctionne de manière fiable, même sous forte charge et avec de grandes quantités de données. App Engine inclut les fonctionnalités suivantes:

- Service web **dynamique**, avec un support complet pour le commun technologies web
- **Stockage persistant** avec requêtes, tri et transactions
- **Mise à l'échelle** automatique et **équilibrage de charge**
- **API** pour **l'authentification** des utilisateurs et l'envoi d'e-mails utiliser des comptes Google
- **Tâches planifiées** pour déclencher des événements dans des temps spécifiés

Et des intervalles réguliers



Créé initialement pour exploiter (en vendant) les ressources non utilisées de leur infrastructure.

- ♦ Plate-forme de cloud public
- ♦ Initialement de type IaaS
- ♦ Propose actuellement du PaaS et SaaS

Objectifs

- Proposer un portail web où l'on peut acquérir des ressources de nature et de taille variable



Le cloud Amazon Web Services (**AWS**) fournit une infrastructure hautement fiable et évolutive pour le déploiement des solutions web-scale , avec un minimum de support et de coûts d'administration, et plus de flexibilité que ce que vous attendez de votre propre infrastructure, sur site ou dans un centre de données.



AWS offre une variété de services d'infrastructure aujourd'hui.
Le diagramme ci-dessous vous présentera la terminologie
AWS et vous aidera
vous comprenez comment votre application peut interagir avec
différents Amazon Web Services et comment différents
services interagir les uns avec les autres.

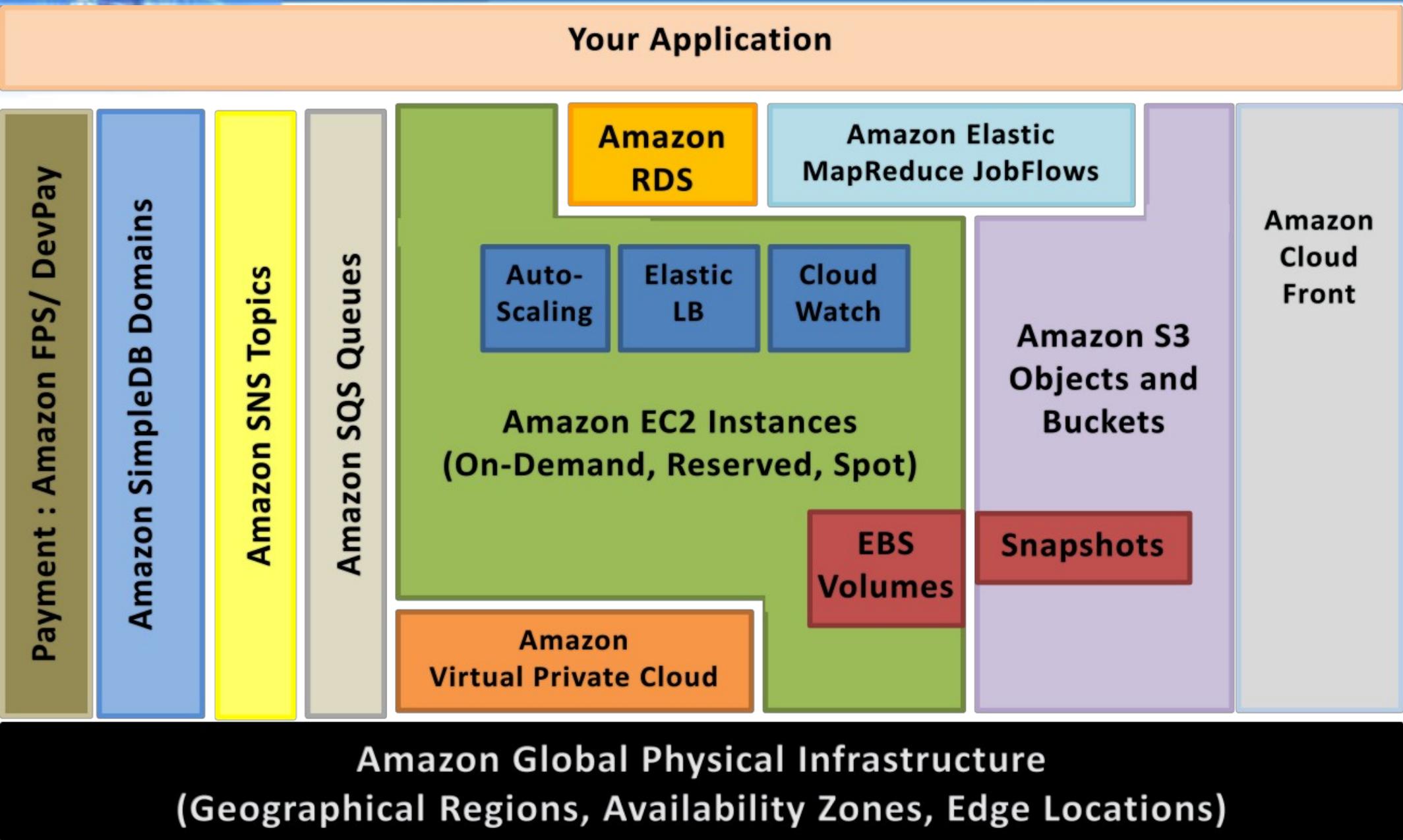


Figure 1: Amazon Web Services

Amazon EC2

Amazon Elastic Compute Cloud (**Amazon EC2**) 1 est un service Web qui fournit une capacité de calcul redimensionnable dans le nuage.

Vous pouvez regrouper le système d'exploitation, le logiciel d'application et les paramètres de configuration associés dans une Amazon Image de la machine (AMI). Vous pouvez ensuite utiliser ces AMI pour provisionner plusieurs instances virtualisées ainsi que la mise hors service

en utilisant des appels de service Web simples pour augmenter et réduire la capacité rapidement, à mesure que vos besoins en capacité changent.

Amazon EC2

Vous pouvez acheter des instances à la demande dans lesquelles vous payez les instances à l'heure ou des instances réservées dans lesquelles vous payer un paiement unique et faible et recevoir un taux d'utilisation inférieur pour exécuter l'instance qu'avec une instance à la demande ou Instances ponctuelles où vous pouvez enchérir sur la capacité inutilisée et réduire davantage vos coûts. Les instances peuvent être lancées dans un ou plus de régions géographiques. Chaque région a plusieurs zones de disponibilité. Les zones de disponibilité sont des emplacements distincts qui sont conçu pour être isolé des défaillances dans d'autres zones de disponibilité et fournir un réseau peu coûteux et à faible latence connectivité à d'autres zones de disponibilité dans la même région.

Les adresses IP élastiques vous permettent d'allouer une adresse IP statique et de l'assigner par programme à une instance. Vous pouvez

activer la surveillance sur une instance Amazon EC2 à l'aide **d'Amazon CloudWatch** afin de gagner en visibilité sur

L'utilisation des ressources, les performances opérationnelles et les modèles de demande globale (y compris les métriques telles que l'utilisation du processeur, les lectures de disque

et écrit, et le trafic réseau).



Amazon EC2

Vous pouvez créer un groupe Auto- à l'aide de la fonction Auto-scaling qui automatiquement

Évaluez votre capacité sur certaines conditions en fonction de la mesure collectée par Amazon CloudWatch. Vous pouvez également distribuer le trafic entrant en créant un équilibreur de charge élastique en utilisant le service Elastic Load Balancing . Amazon Elastic Block

Stockage (**EBS**) volumes fournissent un stockage persistant lié au réseau aux instances Amazon EC2.



Amazon EC2

- Amazon EC2 présente un environnement virtuel , vous permettez de:
 - Utiliser une interface Web pour lancer des instances avec une variété de systèmes d'exploitation
 - qui sont regroupés dans **Amazon Machine Images** (AMI).
 - Charger vos instances avec votre coutume d'environnement d'application.
 - Gérer l'authorisations d'accès de votre réseau
 - Amazon EC2 réduit le temps requis pour obtenir et démarrer de nouvelles instances de serveur aux minutes.
 - Cela vous permet d'augmenter rapidement la capacité
- lorsque vos besoins informatiques changent.

propose trois modèles d'instance qui diffèrent dans leur disponibilité et leur structure de prix

- Instances à la demande Pay-by-the hour
- Commencez et arrêtez comme vous le souhaitez
- Instances réservées
- Payer des frais initiaux annuels et recevoir un rabais sur la charge horaire
- Commencez et arrêtez comme vous le souhaitez
-

- Instances ponctuelles
- Enchère pour la capacité EC2 inutilisée
- Mentionnez votre prix au comptant et si le taux du marché est moins que votre enchère, vous obtenez votre instance
- L'instance se termine automatiquement si votre **SpotAmazon**

Le prix devient inférieur au taux du marché actuel

Amazon Web Services

Virtual machines and virtual disks

Une machine virtuelle peut en avoir une ou

plus de disques virtuels attachés à celui-ci

Serveur de calcul

Il existe deux types de disques virtuels:

Disque virtuel géré par le stockage

Disque virtuel local

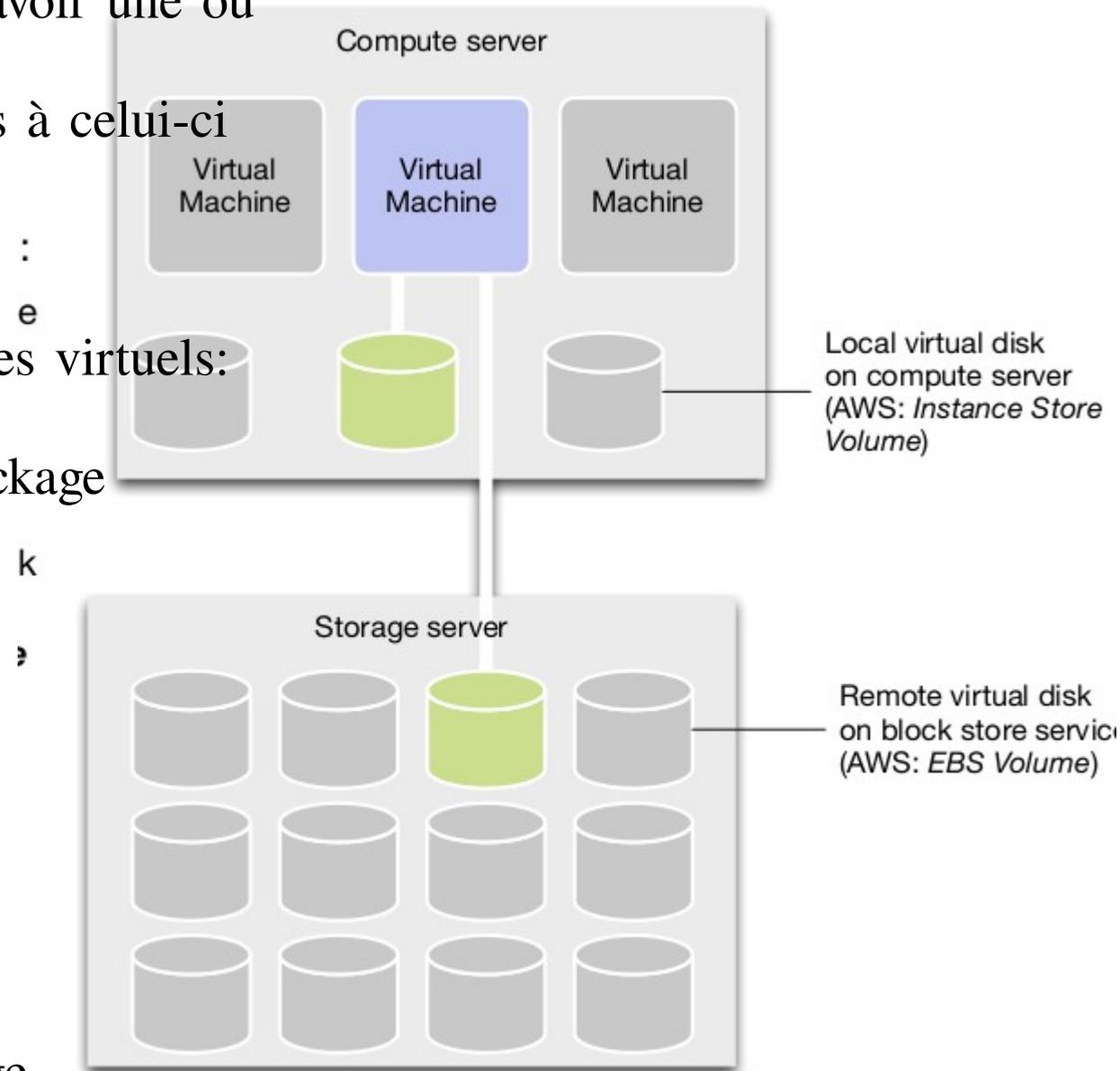
sur le serveur de calcul

(AWS: magasin d'instance

de volume)

service et affecté sur un stockage

serveur: volume EBS Cycle de vie indépendant du Machine virtuelle



Amazon Web Services

Virtual machines and virtual disks

- (Rarement utilisé de nos jours :)

alloué sur le même serveur que

héberge la machine virtuelle: Instance

Volume du magasin

Serveur de stockage

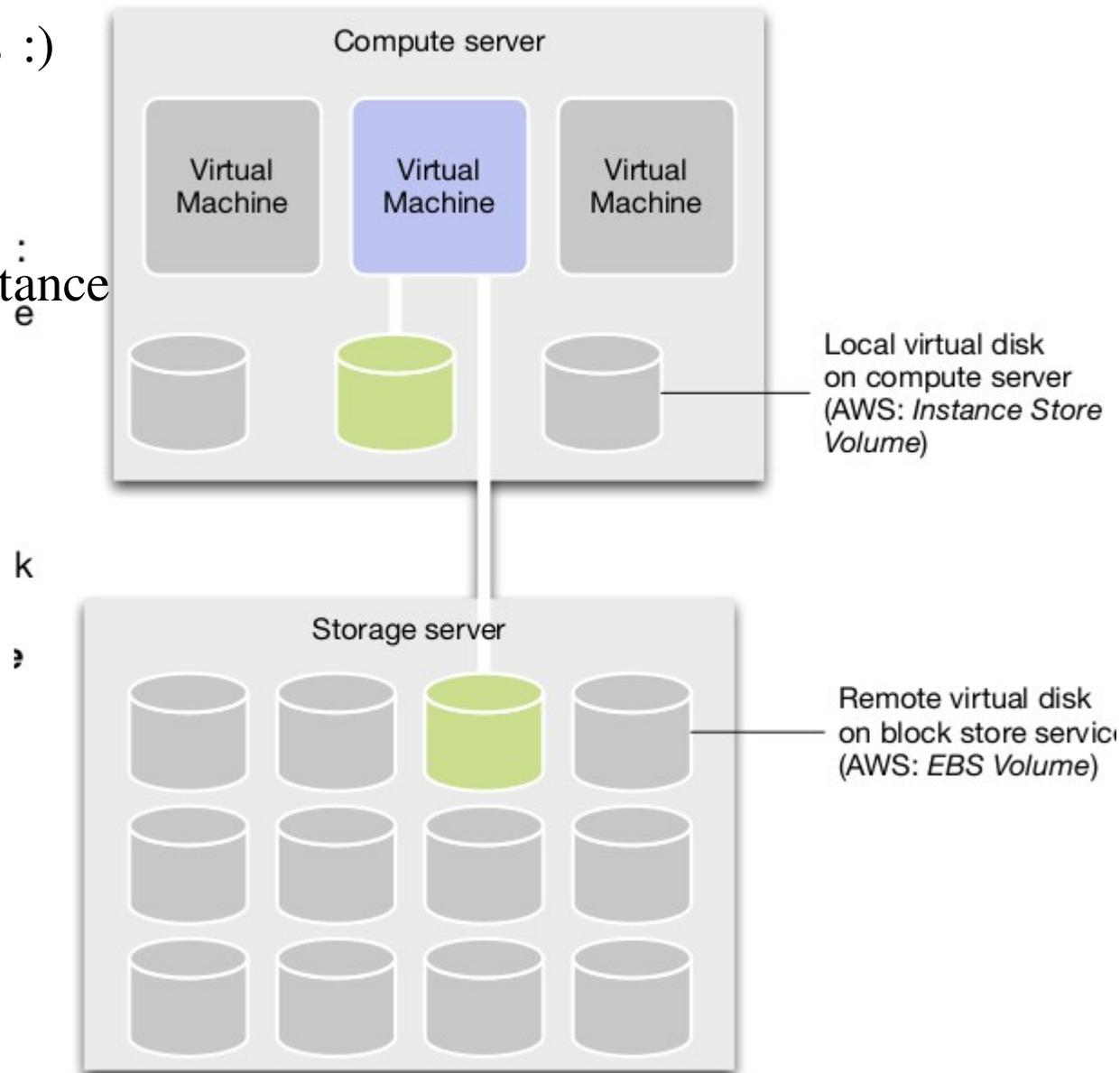
- Le disque est désalloué le

Disque virtuel distant

on block store service

(AWS: volume EBS)

la machine virtuelle est désallouée



Paramètres d'instance EC2

■ Puissance du processeur

- Mesuré dans Elastic Compute Unit (ECU) - Défini par Amazon comme la capacité CPU équivalente d'un

Processeur Opteron / Zeon de 1,0 à 1,2 GHz 2007

■ Mémoire

- Mesuré en GB

■ Performances d'E / S

Trois niveaux: Faible / Modéré / Élevé

■ Les instances haut de gamme ont 10 Gigabit Ethernet

EC2

EC2 instance types

Instance Name	Mem (GB)	CPU Capacity	Disk (GB)	Platform	On-Demand Pricing / hour (Linux)
Micro	0.59	Upto 2 ECUs	--	32/64	\$0.02
Small	1.7	1 core - 1 ECU	160	32	\$0.085
Large	7.5	2 cores, 2 ECUs each	850	64	\$0.34
Extra Large	15	4 cores, 2 ECUs each	1690	64	\$0.68
High-Mem Extra Large	17.1	2 cores, 3.25 ECUs each	420	64	\$0.50
High-Mem Double Extra Large	34.2	4 cores, 3.25 ECUs each	850	64	\$1.00
High-Mem Quad Extra Large	68.4	8 cores, 3.25 ECUs each	1690	64	\$2.00
High CPU Medium	1.7	2 cores, 2.5 ECUs each	350	32	\$0.17
High CPU Extra Large	7	8 cores, 2.5 ECUs each	1690	64	\$0.68
Cluster Compute Quad XL	23	33.5 ECUs	1690	64	\$1.30
Cluster GPU Quad XL	22	33.5 ECUS + 2x GPUs	1690	64	\$2.10



Amazon **S3** est un data store hautement durable et distribué. Avec une interface de services Web simple, vous pouvez stocker et récupérer de grandes quantités de données en tant qu'objets dans des compartiments (seaux) à tout moment, de n'importe où sur le Web en utilisant HTTP standard.

Des copies d'objets peuvent être distribuées et mises en cache dans 14 emplacements périphériques à travers le monde en créant une distribution

Utilisation du service **Amazon CloudFront** - un service Web pour la diffusion de contenu (contenu statique ou en continu).

Amazon SimpleDB

est un service Web qui fournit la fonctionnalité de base d'une base de données - recherche en temps réel et simple requête de données structurées

- sans la complexité opérationnelle. Vous pouvez organiser l'ensemble de données en domaines et exécuter des requêtes sur l'ensemble des

données stockées dans un domaine particulier. Les domaines sont des collections d'éléments décrits par des paires attribut-valeur.

S3

Amazon Simple Storage Service (**S3**) est un serveur Web hautement performant et hautement disponible

service de stockage qui prend en charge les fichiers très volumineux.

- Vous pouvez écrire, lire et supprimer des objets (fichiers) dans S3 contenant de 1 octet à 5 To de données. le

le nombre d'objets pouvant être stockés est illimité.

- Chaque objet est stocké dans un bucket et un bucket peut être stocké dans l'une de plusieurs régions.
- Les objets stockés dans une région ne quittent jamais la région sauf si vous les transférez.



Amazon RDS

Amazon Rational Database Service (**Amazon RDS**) fournit un moyen facile de configurer, d'exploiter et de mettre à l'échelle une base de données relationnelle dans le nuage. Vous pouvez lancer une instance DB et accéder à une base de données MySQL complète et ne pas s'inquiéter à propos des tâches d'administration de base de données courantes comme les sauvegardes, la gestion des correctifs, etc.



Amazon SQS

Le service Amazon Simple Queue (**Amazon SQS**) est une file d'attente distribuée fiable et hautement évolutive pour stocker les messages lorsqu'ils se déplacent entre les ordinateurs et les composants de l'application.

Amazon Simple Notifications Service (**Amazon SNS**) fournit un moyen simple de notifier les applications ou les personnes du cloud en créant des rubriques et en utilisant un protocole publish-subscribe.



Amazon Elastic MapReduce

Amazon Elastic MapReduce fournit un framework Hadoop hébergé s'exécutant sur l'infrastructure Web d'Amazon

Elastic Compute Cloud (Amazon EC2) et Amazon Simple Storage Service (Amazon S3) et vous permet de créer

JobFlows personnalisés. JobFlow est une séquence d'étapes MapReduce.



Amazon VPC

Amazon Virtual Private Cloud (**Amazon VPC**) vous permet d'étendre votre réseau d'entreprise dans un cloud privé

contenu dans AWS. Amazon VPC utilise le mode de tunnel IPSec qui vous permet de créer une connexion sécurisée entre un

passerelle dans votre centre de données et une passerelle dans AWS.



Amazon Route53

Amazon Route53 est un service DNS hautement évolutif qui vous permet de gérer vos enregistrements DNS en créant un HostedZone pour chaque domaine que vous souhaitez gérer.



AWS Identity and Access Management (IAM) vous permet de créer plusieurs utilisateurs avec des informations d'identification de sécurité uniques et gérer les autorisations pour chacun de ces utilisateurs dans votre compte AWS. IAM est intégré de manière native dans AWS Services.

Aucune API de service n'a été modifiée pour prendre en charge IAM et les applications et outils sortants créés au-dessus des API de service AWS continuer à travailler lors de l'utilisation d'IAM.













