

# Université Kasdi Merbah de Ouargla

Faculté des Sciences, de la Technologie et des Sciences de la Matière

Département des Mathématiques et d'Informatique



Mémoire

## MASTER ACADEMIQUE

Domaine : Mathématique et Informatique

Filière : Informatique Académique

Spécialité : Informatique Industrielle

Présenté par : **AOUAMEUR Zakia**

**TAHRINE Halima**

Thème

### Comparaison et mise en place des plateformes de Cloud Computing : OpenStack et Eucalyptus

Soutenu publiquement

Le :.... /06 /2013

Devant le Jury :

Président : Bachir Said, Maitre de conférences

Examinatrice : Adel Zga, Maitre assistant

Rapporteur : Hmida Djedjai, Maitre assistant

UKM Ouargla

UKM Ouargla

UKM Ouargla

Année Universitaire : 2012/2013

# Remerciements

C'est avec l'aide de Dieu qu'a vu le jour ce présent travail.

Ensuite, il n'aurait pas pu être achevé sans le soutien, les conseils et les encouragements de certaines personnes auxquelles nous tenons ici à exprimer nos sincères remerciements.

En premier lieu, nous exprimons toute notre gratitude pour Notre Promoteur, Mr: *hmida djedjai*

pour ces précieux conseils, sa disponibilité, la confiance qu'il nous a toujours témoigné et la

sollicitude dont il nous a entouré, et ce tout au long de l'élaboration du présent travail.

Nous n'oublions pas non plus Nos Enseignants, qui tout au long du cycle d'études à à UNIVERSITE

KASDI MERBAH OUARGLA, nous ont transmis leur savoir.

Nous adressons une pensée particulièrement affective à Nos Amis qui ont rendu agréables nos longues années d'études.

Nous tenons enfin à remercier tous ceux qui ont collaborés de près ou de loin à l'élaboration de ce

travail. Qu'ils acceptent nos humbles remerciements.

# Dédicaces

*Je dédie ce modeste A mon père et à ma mère qui ont veillé nuit et jour pour que je puisse arriver à ce stade de mes études*

*Je dédie ce modeste effort à l'esprit pour de mon père, et je souhaite de mon Dieu de l'introduit le paradis.*

*Et je le dédie aussi à toutes ces personnes qui me sont très chères :*

*Mes chères sœurs*

*Mes chers frères*

*A mes chères nièces **Hebat et Sara***

*A mon encadreur, Mr **Imida.djedjai** sans qui ce modeste travail n'aurait pas vu le jour.*

*Et toute la famille **Aouameur et Hamaza***

*Et je n'oublier pas mes chère amies, en souvenir des agréables moments qu'ont à passé ensemble, en particulier: **Nasria, Hadjer, Halima et Rannda, Assia, Saliha, Houda, Amina, Khoula, Nesiba, Fajra, Jamila, Amina, Kholoud. ....***

*A tous mes enseignants qui ont grandement contribué à l'enrichissement de mes connaissances.*

*A tous les membres de la promotion de Master et licence LMD de l'université Kasdi Merbeh de OUARGLA  
Et à tous ceux qui me connaissent*

**ZAKIA**

# Dédicaces



*Je dédie ce modeste travail à ma très chers parents: **Abed Rahmane** et **Fadila**,  
qu'ils n'ont pas cessés de ma encourager durant toutes mon étude, Je demande  
que protégés par Dieu, à me soutenir dans l'avenir.*

*A mon encadreur, Mr **hmida.djedjai** sans qui ce modeste travail n'aurait pas vu  
le jour.*



*Aussi à mes frères et mes sœurs et leurs enfants.*

*Et à toutes les personnes des familles **Tahrine** et **Naceur***

*À mes chères amies **Zakia, Hadjer, Nasria, Randa, Saliha, Assia, Khaoula,**  
**Amina, Kholoud.***

*Ainsi qu'à toutes les personnes qui m'ont aidée à réaliser ce travail.*



*À tous mes enseignants, à tous mes collègues 2 Master informatique 12/13 et à  
tous mes amis.*

*Et À tous les responsables et les travailleurs du Département **Mathématique** et  
**informatique.***

***Halima***

# Table des matières

Liste des figures .....	IV
Liste des tableaux.....	V
Introduction générale .....	VI

## *Partie I. État de l'art sur le Cloud Computing*

### *Chapitre 1 : Cloud Computing -Concepts et Terminologie*

1.1 Introduction .....	1
1.2 Historique .....	1
1.3 Définition .....	2
1.4 Les Caractéristique de Cloud Computing .....	2
1.5 Les services de Cloud Computing.....	3
1.5.1 IaaS : Infrastructure as a service.....	3
1.5.2 PaaS : Platform as a service.....	4
1.5.3 SaaS : Software as a Service .....	5
1.6 Les Modèles de déploiement de Cloud Computing .....	7
1.6.1 Cloud privé: .....	7
1.6.2 Cloud public: .....	7
1.6.3 Cloud communautaire : .....	8
1.6.4 Cloud hybride : .....	8
1.7 Les éléments constitutifs du Cloud Computing .....	9
1.7.1 Infrastructure .....	9
1.7.2 Virtualisation .....	9
1.7.3 Interfaces de service .....	9
1.7.4 Centre de données (Datacenter) .....	10
1.8 Quelle est la différence entre la virtualisation et le Cloud Computing ? .....	10
1.9 Quelle est la différence entre le Grid et le Cloud Computing ? .....	11
1.10 Les avantages du Cloud Computing .....	11
1.11 Les inconvénients de Cloud Computing .....	12
1.12 Les principaux fournisseurs de Cloud Computing.....	12
1.12.1 Amazon Web Services (IaaS): .....	12
1.12.2 Amazon Web Services (PaaS):.....	12
1.12.3 Microsoft Azure (PaaS):.....	12
1.12.4 Google AppEngine (PaaS): .....	13
1.12.5 force.com (PaaS): .....	13
1.12.6 salesforce.com (SaaS): .....	13
1.12.7 Google Apps (SaaS): .....	13
1.13 Sécurité dans le Cloud.....	13
1.13.1 La confidentialité.....	13
1.13.2 L'Intégrité.....	13



1.13.3 La Disponibilité .....	14
1.14 La technologie de stockage en réseau .....	14
1.14.1 Stockage de SAN.....	14
1.14.2 Stockage de NAS.....	16
1.14.3 Stockage de DAS.....	17
1.15 Conclusion.....	17
<b><i>Chapitre 2 : Les plateformes de Cloud Computing Open Source et Commercial</i></b>	
2.1 Introduction .....	18
2.2 Les Open Source .....	18
2.2.1 Eucalyptus .....	18
2.2.2 Plateforme OpenStack .....	21
2.4 Plateforme propriétaires de Cloud Computing .....	26
2.3.1 Plateforme Microsoft Windows Azure.....	26
2.4 Conclusion.....	34
<b><i>Partie II. Nos contributions</i></b>	
<b><i>Chapitre 3 : Mise en place d'OpenStack et Eucalyptus</i></b>	
3.1 Introduction .....	35
3.2 Enivrement d'installations.....	35
3.2.1 Serveur .....	35
3.2.2 VMware vSphere.....	35
3.2.3 Les machine virtuelles.....	36
3.3 Installation d'OpenStack .....	36
3.3.1 L'installation de Controller Node: .....	38
I.8.1. L'installation de Network Node: .....	45
I.8.2. Compute Node .....	47
Figure 3.14: login de machine compute node .....	48
3.4 Installation plateforme Eucalyptus.....	49
3.5 Conclusion.....	49
<b><i>Chapitre 4 : Un Framework Comparatif entre OpenStack et Eucalyptus</i></b>	
4.1 Introduction .....	50
4.2 Étude comparative entre deux plate-formes.....	50
4.2.1 Comparaison générale .....	50
4.2.2 Comparaison pendant l'installation .....	52
4.3 Conclusion.....	58
Conclusion générale.....	59
Glossaire.....	64
Annexes .....	70

## Liste des figures

Figure	Page
Figure1.1: Les différents attributs de Cloud Computing .....	2
Figure 1.2: Le contrôle des services dans IaaS .....	4
Figure 1.3:Le contrôle des services dans PaaS .....	5
Figure 1.4: Le contrôle des services dans SaaS .....	6
Figure 1.5:Service du Cloud Computing .....	6
Figure1.6: Cloud Privé .....	7
Figure 1.7: Cloud public .....	8
Figure 1.8: Cloud Community .....	8
Figure 1.9: Cloud Hybride .....	9
Figure 1.10:Exemple d'un Centre des données .....	10
Figure 1.11:Architecture générale de SAN .....	15
Figure 1.12:Architecture générale de NAS .....	16
Figure 2.1: Architecture d'Eucalyptus .....	20
Figure 2.2: Architecture d'OpenStack .....	21
Figure 2.3: Architecture de Swift.....	22
Figure 2.4: Architecture Nova.....	23
Figure2.5: Architecture de Glance .....	25
Figure 2.6: Architecture OpenStack Horizon .....	25
Figure 2.7: Architecture OpenStack Keystone.....	26
Figure 2.8: Architecture Windows Azure AppFabric .....	28
Figure 2.9: Les composants au cœur de Windows Azure .....	29
Figure 2.10: Architecture SQL Azure .....	30
Figure 2.11: Les composants de VMware vSphere.....	31
Figure 3.1: L'architecture OpenStack pour l'installation.....	36
Figure 3.2: login de machine controller .....	39
Figure 3.3 :Table de Keystone .....	40
Figure 3. 4: List de Tenant .....	41
Figure 3.5: List de Role.....	41
Figure 3.6: List de User.....	41
Figure 3.7: Table de Glance .....	42
Figure 3.9: Vérification d'image Glance.....	42
Figure 3.10: Table de Nova.....	43
Figure 3.11: Table de Cinder.....	44
Figure 3.12: Login plateforme OpenStack Horizon.....	45
Figure 3.13: Interface plateforme OpenStack Horizon .....	45
Figure 3.14: login de machine network node.....	46
Figure 3.15: login de machine compute node .....	48

## Liste des tableaux

Table		Page
Table 1.1	les différents entre SAN et NAS .....	17
Table 3.1	Architecture et informations de nœud .....	38
Table 4.1	Framwork01 : caractéristiques générales .....	51
Table 4.2	Framwork02 : caractéristiques techniques.....	57



---

## Introduction générale

**E**n 1969, Leonard Kleinrock l'un des principaux scientifiques du projet ARPANET, a déclaré: "A partir de maintenant, les réseaux informatiques sont encore à leurs balbutiements, mais à mesure qu'ils grandissent et deviennent sophistiqués, nous allons probablement voir la propagation de services informatiques qui devient comme les services d'électricité et de téléphone, desservira à tous les habitations et les bureaux à travers le pays". La vision de Leonard Kleinrock est de fournir les services de l'informatique comme un service public, qui est aujourd'hui une réalité grâce au cloud computing.

Cloud Computing est une méthode qui propose de différents types de services sur Internet. Il peut être une application logicielle (par exemple, Google Drive) livrée sur l'Internet. Il peut être une plate-forme de développement (par exemple, Google App Engine) sur laquelle les utilisateurs peuvent créer leur application. Il peut être un système d'exploitation (par exemple, Amazon EC2) où les utilisateurs peuvent créer leur propre plate-forme et logiciel. Tous ces services sont accessibles via une connexion Internet [26].

Le Cloud Computing permet de résoudre les problématiques liés à la rareté des ressources informatiques comme CPU, RAM, Network, etc. Surtout ces dernières années où la demande pour ces ressources est augmentée d'une manière très élevée.

Infrastructure-as-a-Service (IaaS) Cloud est un des modèles du Cloud Computing utilisés pour livrer des ressources informatiques, et ceci est accompli en provisionnant les ressources virtuelles aux clients. Par exemple, si une personne désire exploiter une entreprise en ligne, il n'y a pas besoin de mettre en place sa propre infrastructure de TI. Au lieu de cela, il peut louer une machine virtuelle d'Amazon et fournir le service [28].

En plus de fournir les ressources virtuelles, IaaS apporte un autre avantage de "pay-per-use". Le modèle pay-per-use peut être facilement compris par comparaison avec un régime de paiement d'électricité, payer seulement la quantité d'électricité consommée. De même, IaaS permet à un client de payer uniquement les ressources utilisés. Cela permet à l'entreprise de n'est

plus investir des capitaux dans l'infrastructure informatique et la permet de concentrer sur leur QoS a fourni à ses clients au lieu de penser sur l'approvisionnement et le sous-dimensionnement des ressources informatiques [26].

Les acteurs impliqués dans le Cloud Computing liés à IaaS sont appelés des intervenants de Cloud (Cloud stakeholders). Les intervenants de Cloud sont les fournisseurs de Cloud et les utilisateurs de Cloud. Les fournisseurs de Cloud sont responsables de la mise en place de l'infrastructure informatique et les utilisateurs de Cloud utilisent l'infrastructure fournie pour effectuer leurs tâches.

La plate-forme de gestion de Cloud est un logiciel qui convertit l'infrastructure informatique physique existante dans une infrastructure de Cloud. Le nombre de plateformes commerciales de gestion de Cloud augmente avec la popularité de IaaS-Cloud, tels que VMware vCloud, CloudForms, etc [29]. Nombreuses contributions ont été faites par la communauté open source associée à IaaS-Cloud. Certains des plates-formes open source sont : Eucalyptus, OpenStack, Nimbus, XCP et OpenNebula [27, 28]. [5,6] citer une liste non exhaustive de certaines plates-formes open source d'IaaS-Cloud.

Le nombre de plates-formes de Cloud augmente jour après jour. Un des problèmes majeurs pour les consommateurs de Cloud est choisi la plate-forme, par-ce-que les caractéristiques du ce dernier varient considérablement. Il est donc difficile pour les consommateurs de Cloud de choisir la solution basé sur ses propres besoins. Par exemple, si un client pense que la gestion de l'alimentation est importante pour son entreprise, donc ils doivent lire les informations fournies dans le site de différentes solutions pour déterminer si la gestion de l'alimentation est fournie avec cette plate-forme ou non.

Pour résoudre le problème de choix une plate-forme de gestion de Cloud, un Framework doit être développée. Le Framework regroupe les fonctionnalités offertes par la plate-forme de gestion de différents types de Cloud, que se soit open-source ou commerciale afin de faciliter la comparaison. Ce Framework offre trois avantages. Tout d'abord, il permet aux consommateurs de Cloud de choisir la solution (plate-forme) adapté à leurs besoins opérationnels. Deuxièmement, les éditeurs de ces logiciels de gestion (les entreprises qui vendent leurs plates-formes, par exemple, Eucalyptus, OpenStack, etc.), peut inclure des fonctionnalités

---

supplémentaires à leur logiciel. Enfin, nouveaux développeurs de logiciels de gestion de Cloud (celui qui commence la création d'un nouveau logiciel de gestion de Cloud) peuvent commencer à construire leur logiciel en incluant les fonctionnalités répertoriées dans le Framework.

Les deux plates-formes de gestion de Cloud OpenStack et Eucalyptus sont l'un des plates-formes populaires de Cloud. En particulier, le nombre de consommateurs de Cloud qui téléchargent ces derniers est plusieurs milliers par mois. Certaines des travaux existantes focaliser sur la performance de ces plates-formes (OpenStack et Eucalyptus). Cependant, il y a peu d'attention sur la comparaison des caractéristiques de ces deux logiciels de gestion de Cloud entre eux ou avec d'autres logiciels open source.

Cette mémoire porte sur un grand objectif. Qui est d'élaborer un Framework permettant de comparer les fonctionnalités, l'architectures, ressources pré-requis, caractéristiques, capacités et d'autre critères pour deux plates-formes de gestion de Cloud Computing plus utilisés sur le marché open source exhaustivement OpenStack et Eucalyptus .

Le reste de ce mémoire est organisé comme suit : un état de l'art sur le Cloud Computing est discuté dans le 1<sup>er</sup> partie qui est composé par de chapitre, un discute les différents terminologies et concept liés au Cloud Computing et l'autre citer la définition et les caractéristiques de quatre(04) plates-formes open sources et commerciales. Le deuxième partie de ce mémoire représente notre contribution en forme de Framework de comparaison entre les deux plates-formes de gestion de Cloud (OpenStack et Eucalyptus), ce dernier est réaliser via deux chapitre où le premier montre l'installation et la mise en place de deux plates-formes et le deuxième discute la comparaison entre eux, à la fin nous résumons notre mémoire avec une conclusion général.

*Partie I. État de l'art sur le Cloud  
Computing*

# *Chapitre 1 : Cloud Computing -Concepts et Terminologie*



## 1.1 Introduction

La technologie de l'internet se développe plus rapide et en manière progressivité. Dans se dernier année il y a une nouvelle destination, Son but est d'améliorer les services dans le domaine TIC (Technologies de l'information et de la communication), il s'agit du « Cloud Computing ».Ce dernier est un concept qui regroupe plusieurs technologies servant à délivrer différents services. Il peut être schématisé par un ensemble indéterminé de ressources informatiques interconnectées et une coopération dans un réseau bien défini.

L'accès aux services se fait à la demande par une application standard facilement disponible à savoir, un navigateur Web.

Dans ce chapitre, nous allons présenter les notions et les concepts fondamentales du Cloud Computing.

## 1.2 Historique

La faible performance des processeurs ; capacité de stockage et d'autre ressources employés dans les systèmes classique sont limités et coûteuse, où ils ne permettent pas l'accès à des services et des ressources de façon plus rapide.

Avec les développements technologiques actuels ont ouvert la voie à l'émergence du concept de Cloud Computing, qui permet l'utilisation des serveurs plus efficaces et moins coûteux, aussi il augmente la performance des connexions d'internet, lorsqu'ils deviennent plus de vitesse à la demande des utilisateurs.

La notion de ce service a été proposée pour la première fois en 1961 lors d'une conférence au MIT (Massachusetts Institute of Technology) par John McCarthy.

Il est également l'inventeur du langage Lisp dans lequel un groupe d'ordinateur se relie pour former un ordinateur virtuel unique permettant le calcul de haute performance.

**Salesforce.com** fut le premier hébergeur de Cloud en 1999, suivi en 2002 par Amazon qui proposa un ensemble d'hébergement d'application et de stockage.

Amazon développa ses services en 2005 (Amazon Web Services) et en 2006 (Elastic Compute Cloud ou EC2).

En 2007, Google, IBM et des universités lancèrent un projet de recherche sur le Cloud qui permit de lui faire gagner en popularité et en consistance.

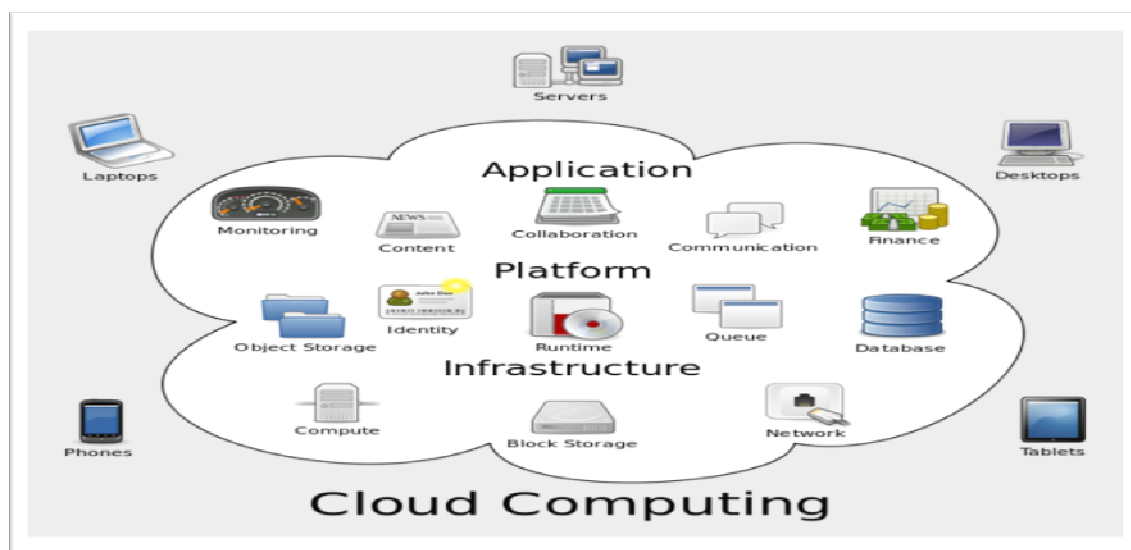
C'est en 2009 que la réelle explosion du Cloud survint avec l'arrivée sur le marché de sociétés comme Google (Google App Engine), Microsoft (Microsoft Azure), IBM (IBM Smart Business Service), Sun (Sun Cloud) et Canonical Ltd (Ubuntu Enterprise Cloud). [9]

### 1.3 Définition

NIST : (National Institute of Standards and Technology) en français " Institut national des normes et de la technologie " définit le Cloud Computing comme suit : « Le Cloud Computing est l'ensemble des disciplines, pratiques, technologies et modèles commerciaux utilisés pour délivrer comme un service à la demande et par le réseau des capacités informatiques (logiciels, plateformes, matériels) ». [33]

Dans la littérature il y a d'autres définitions de Cloud Computing nous choisissons les suivantes :

- ✓ est un modèle qui offre aux utilisateurs du réseau un accès à la demande, à un ensemble de ressources informatiques partagées et configurables [2], aussi avec le Cloud, n'importe qui peut entrer et à partir de n'importe quel endroit et à tout moment.
- ✓ C'est un concept abstrait, son but est de pousser les entreprises à externaliser les ressources numériques qu'elles stockent. Ces ressources offrant des capacités de stockage et de calcul, des logiciels de gestion de messagerie, et d'autres services sont mises à disposition par des sociétés tierces et accessibles, grâce à un système d'identification, via un PC et une connexion à Internet. [3]



**Figure1.1:** Les différents attributs de Cloud Computing [7]

### 1.4 Les Caractéristique de Cloud Computing

- ✓ **Elasticité** : pour les utilisateurs de Cloud Computing a des capacités illimitées, qui peuvent être augmentées ou réduites selon l'usage.
- ✓ **pay-as-you-use** : les clients du Cloud vont donc recevoir des tarifications, les plus précises possibles, pour les ressources qu'ils utilisent. Par exemple la tarification

d'Amazon, pour le micro machine virtuelle à 0.02\$ par heure et 3.10\$ par heure pour les grosses machine virtuelles. [12]

- ✓ **Self-service (à la demande)** : capacité à fournir une ressource informatique automatiquement, sans requérir d'interaction humaine côté fournisseur. [5]
- ✓ **Mesure de la qualité de services** : évaluer et garantir un niveau de performance et de disponibilité adapté aux besoins spécifiques des clients. [1]
- ✓ **Accès réseau universel**: L'accès à les ressources est très rapide et à laide d'un réseau, par des protocoles standards en manière très élasticité.
- ✓ **Mise en commun de ressources** : Datacenter fournissant les ressources (machines, stockage, etc.) pour les différentes clients en monde partagé. [13]
- ✓ **Multi-tenancy** : La capacité de fournir un service simultanément à plusieurs clients, cela permet également d'augmenter l'utilisation des ressources IT déployée.
- ✓ **Disponibilité** :La haute disponibilité de la plate-forme est obtenue par redondance et par la capacité de se remettre rapidement en cas des problèmes.
- ✓ **Green IT** :Green IT est un concept qui désigne l'ensemble des nouvelles technologies à faible impact environnemental. Le Cloud Computing, basé sur la virtualisation, qu'il réduise les risques des machines physiques sur l'environnement et la consommation énergétique.

## 1.5 Les services de Cloud Computing

Le Cloud Computing propose trois types de services principaux :

### 1.5.1 IaaS : Infrastructure as a service

IaaS (Infrastructure as a service) en Français "L'infrastructure en tant que service", est un service principal dans le Cloud, ce service fourni l'entreprise par différente composants informatiques comme des espaces de stockages, équipements réseaux, des unités centrales, etc.

Les utilisateurs peuvent accédés à ces services à la demande via l'internet sans restriction, comme le travaille dans une matériel local.

Les services sont présentés par l'IaaS, stockent dans des serveurs virtuels situés dans des Centres des données ; parmi ses avantages sont :

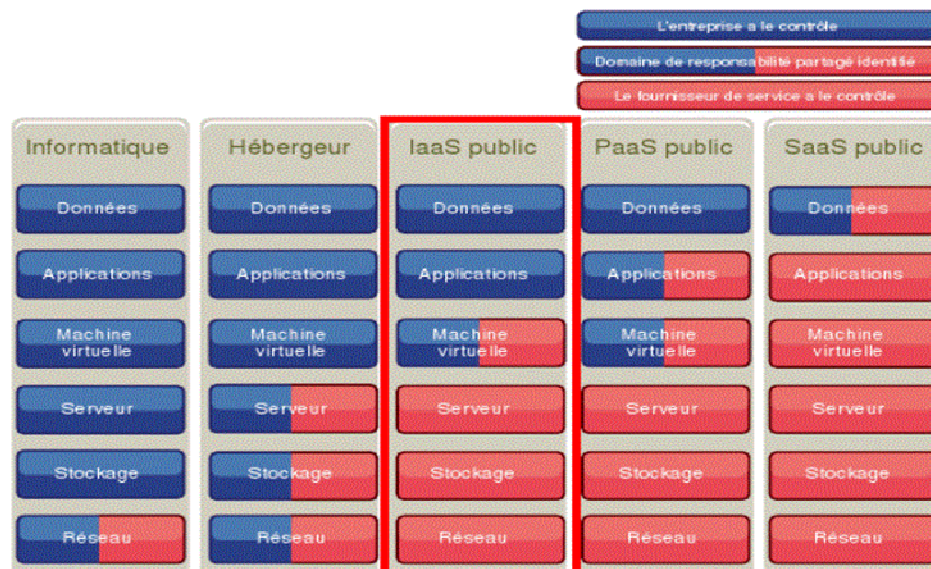
- ✓ Éviter d'achat et la gestion de matériel,
- ✓ Offre une grande flexibilité,
- ✓ La possibilité d'installer tout type de logiciel,



- ✓ Contrôle total des systèmes (administration à distance par SSH ou RDP "Remote Desktop"). [3]

Parmi les prestataires d'IaaS, on peut citer : Amazon avec EC2 qui est le roi de ce genre d'infrastructures [3], Gogrid et Terremark.

La figure (1.2) présente qui est responsable de contrôle des services, à l'égard d'IaaS.



**Figure 1.2:** Le contrôle des services dans IaaS [12]

### 1.5.2 PaaS : Platform as a service

PaaS (Platform as a service) en Français "plate-forme en tant que service", est un modèle composé de tous les éléments et les services nécessaires pour faciliter les développements des applications où PaaS prépare d'environnements spécialisés pour aider les utilisateurs dans la construction, la livraison, l'extension de leurs projets. L'utilisateur embauche une plateforme sur laquelle il peut développer, tester et exécuter ses applications, parce que PaaS évite d'acheter et d'installer des logiciels, aussi il ne gère ni ne contrôle l'infrastructure sous-jacente, mais contrôle les applications déployées.

Aussi est une plateforme d'exécution hébergée par un opérateur relié au réseau internet. [6] parmi ses avantages :

- ✓ Facilité à gérer des développements des applications,
- ✓ Le déploiement est automatisé, [3]
- ✓ Ne nécessite pas une infrastructure.

Il y a plusieurs fournisseurs des services PaaS, les principaux sont : Microsoft avec AZURE, Google avec Google App Engine et Orange Business Services.

La figure (1.3) présente qui est responsable de contrôle des services, à l'égard de PaaS.

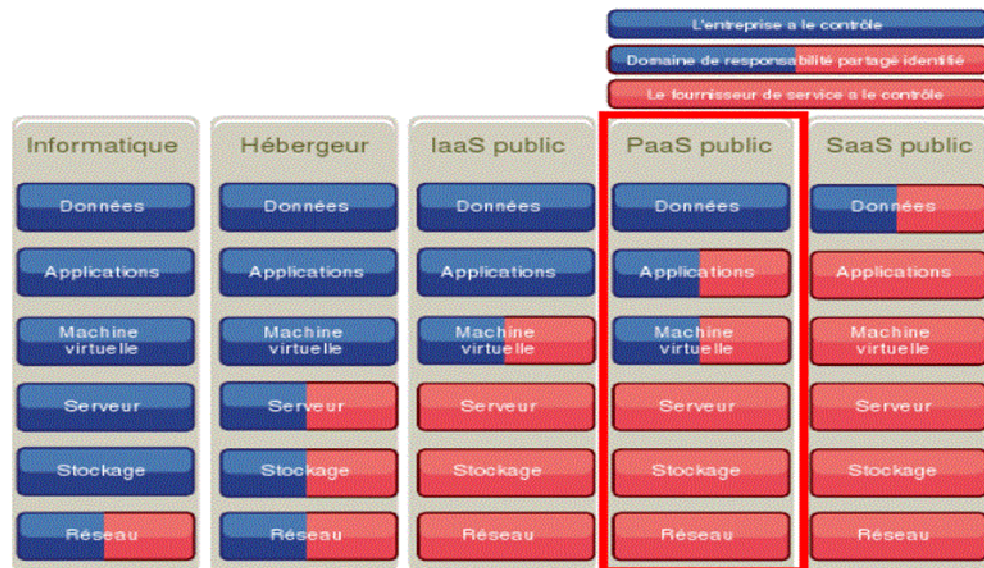


Figure 1.3: Le contrôle des services dans PaaS[31]

### 1.5.3 SaaS : Software as a Service

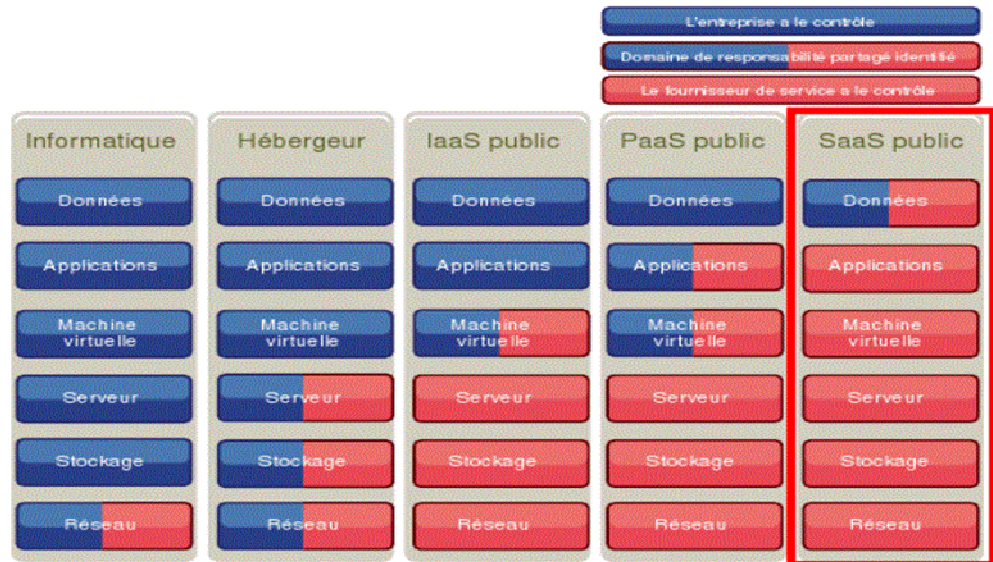
SaaS (Software as a Service) en Français " L'application en tant que service" est le modèle le plus utilisé dans le monde après le service d'email, est un modèle de distribution des logiciels et les applications qu'ils sont hébergées dans des centres de données, qu'il donne la possibilité des clients pour consommation ces application à la demande via l'internet avec une facturation à l'usage réel.

Ces applications, prêts à l'emploi, et ne nécessitent pas de maintenance, d'installation de logiciel, le maintenir et le mettre à jour, toutes ces opérations sont effectuées par le fournisseur d'application, dans SaaS l'utilisation d'application reste transparente pour les utilisateurs, qui ne se soucient ni de la plateforme, ni du matériel.

Le SaaS remplace l'ASP (Application Service Provider) en Français FAH (Fournisseur d'Applications Hébergées), qui est une entreprise qui fournit des logiciels ou des services informatiques à ses clients au travers d'un réseau, parmi ses avantages:

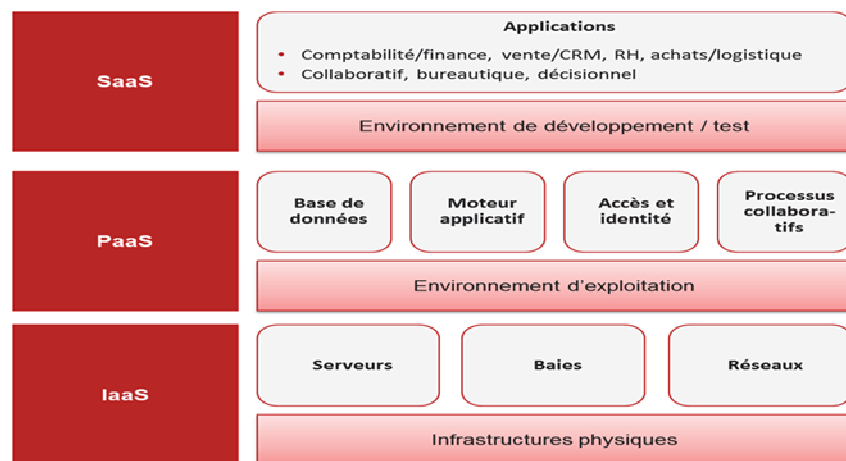
- ✓ Pas d'installation,
- ✓ Pas de mise à jour,
- ✓ Plus de License,
- ✓ Paiement à l'usage,
- ✓ Facilité de faire le test de nouveaux logiciels.

Les prestataires de solutions SaaS les plus connus sont : Google avec Google Drive et Microsoft avec Office 365. La figure (1.4) présente qui est responsable de contrôle des services, à l'égard de SaaS.



**Figure 1.4:** Le contrôle des services dans SaaS [36]

La figure (1.5) résume les trois services principaux de Cloud Computing :



**Figure 1.5:**Service du Cloud Computing [44]

Aussi il existe d'autre formes de service de Cloud Computing comme :

- ✓ **DaaS (Database as a Service)** en Français (Base de données en tant que service), offre des fonctions traditionnelles de bases de données, typiquement définition des données, le stockage et la récupération, sur une base d'abonnement sur le Web. [37]

- **CaaS (Communication as a Service)** en Français (Communication en tant que service), est un service de communication audio et vidéo, services collaboratifs, communications unifiées, messagerie électronique, messagerie instantanée partage de donnée (web conférence). [2]
- **NaaS (Network as a Service)** en Français (Réseau comme un service), fournir à l'utilisateur la possibilité d'utiliser le réseau, les services de connectivité de transport et inter-Cloud services de connectivité réseau. Parmi des modèles de NaaS, le modèle VPN "Réseau privé virtuel", Il permet à un ordinateur hôte pour envoyer et recevoir des données à travers des réseaux partagés ou publics, comme s'il s'agissait d'un réseau privé avec toutes les fonctionnalités, les politiques du ce réseau. [37]

## 1.6 Les Modèles de déploiement de Cloud Computing

On peut distinguer quatre types principaux dans le Cloud sont : le Cloud privé, le Cloud public, le Cloud hybride (Mixte) et le Cloud communautaire, nous avons expliqué chaque type plus détaillé.

### 1.6.1 Cloud privé:

En anglais "Private Cloud" est un ensemble des services et des ressources disponible à un seul client par exemple une entreprise, le Cloud privé peut être géré par l'entreprise elle-même, ou bien avec ses branches, dans ce cas il s'appelle "Le Cloud privé Interne", en d'autres façons il peut être géré par un prestataire externe qu'il est loué par l'entreprise, dans ce cas s'appelle "Le Cloud privé Externe", qu'il est accessible via des réseaux sécurisés de type VPN (Virtual Private Network), par exemple Amazon Virtual Private Cloud (Amazon VPC).

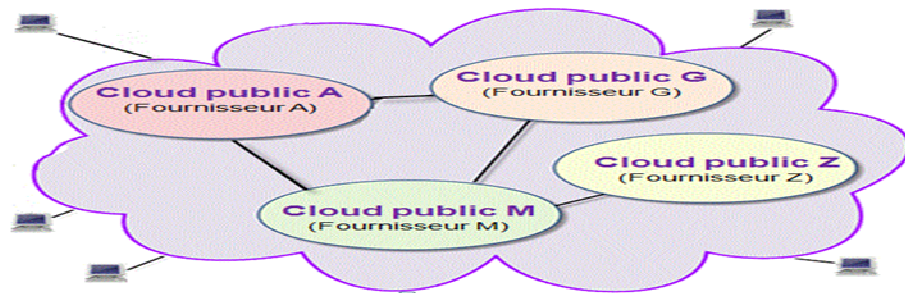


Figure 1.6: Cloud Privé [44]

### 1.6.2 Cloud public:

En anglais "Public Cloud" est un ensemble des services et des ressources accessibles par Internet et géré par un prestataire externe, ces ressources et services sont partagés entre plusieurs clients, qu'ils les utilisent à la demande et à tout moment sans savoir où elles existent, aussi ces services peuvent être gratuits ou payants.

On cas les services sont payants, il existe des contrats SLA (Service Level Agreement) entre les clients et les fournisseurs, SLA est un document qui définit la qualité de service requise entre les deux. Quelques exemples de Cloud public: Amazon Elastic Compute Cloud (EC2) et Amazon S3, Sun Cloud, IBM's Blue Cloud, Google AppEngine et Windows Azure Services Platform. [3]



**Figure 1.7:** Cloud public [44]

### 1.6.3 Cloud communautaire :

En anglais " Cloud Community" est un ensemble des infrastructures, qu'ils permettent à plusieurs clients ou organisations de partager les différentes ressources, ces ressources sont généralement spéciale à des organisations, ce type de Cloud peut être géré par ces organisation elles-mêmes ou par des fournisseurs externes.

Aussi il permet à un groupe des utilisateurs à créer leur propre Cloud avec des caractéristiques de Cloud privé tel que la sécurité, ressources dédiées et un coût réduit.



**Figure 1.8:**Cloud Community [44]

### 1.6.4 Cloud hybride :

En anglais " Hybrid Cloud" est une composition de deux ou trois types de Cloud (Privé, communautaire et public).

Par exemple l'utilisation des applications dans un Cloud public mais ces applications nécessitent des données stockées sur un Cloud privé.

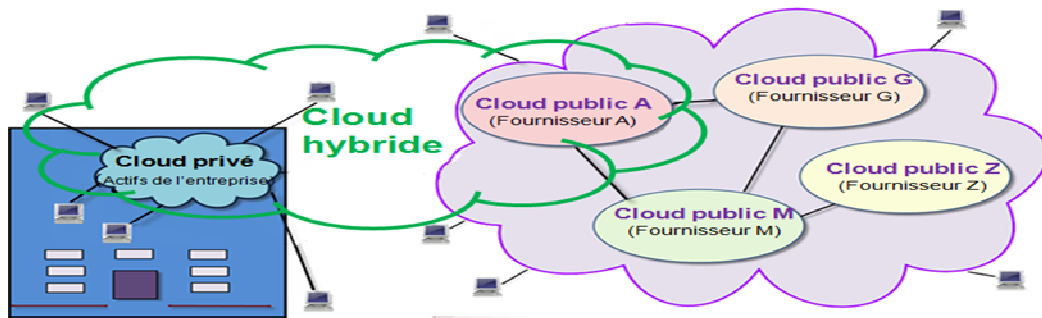


Figure 1.9: Cloud Hybride [44]

## 1.7 Les éléments constitutifs du Cloud Computing

### 1.7.1 Infrastructure

L'infrastructure informatique du Cloud est un assemblage de serveurs, d'espaces de stockage et de composants réseau organisés de manière à permettre une croissance incrémentale supérieure à celle que l'on obtient avec les infrastructures classiques. Ces composants doivent être sélectionnés pour leur capacité à répondre aux exigences d'extensibilité, d'efficacité, de robustesse et de sécurité. Les serveurs d'entreprise classiques ne disposent pas des capacités réseau, de la fiabilité ni des autres qualités nécessaires pour satisfaire efficacement et de manière sécurisée les accords de niveau de service SLA (service level agreement).

Par ailleurs, les serveurs d'un Cloud affichent des coûts de fonctionnement moins élevés et ils peuvent être plus fiables s'ils ne sont pas tous équipés de disques internes. [7]

### 1.7.2 Virtualisation

La virtualisation est la principale technologie dans Cloud, est une manière pour partitionner une ressource physique en plusieurs ressources virtuelles, par exemple: un serveur, un espace de stockage ou un réseau lors de la création des machines virtuelles.

Elle permet intégrer les différents serveurs en façons plus flexibles pour faciliter l'utilisation.

Le but de la virtualisation est de faire la transparence d'utilisation et d'efficacité l'exploitation des ressources, d'assurer le fonctionnement des différents services et la séparation entre de multiples locataires (utilisateurs) impliqués dans d'un matériel physique.

### 1.7.3 Interfaces de service

L'interface de service placée entre le fournisseur et le client est un élément de différenciation du Cloud. Elle représente un contrat qui fait respecter la proposition de valeur décrite par des SLA et des conditions tarifaires. Si le Cloud semble nouveau, c'est

principalement en raison de cette interface. Elle représente la valeur d'un fournisseur et sert de base à la concurrence. Par l'ajout d'interfaces de libre service, nous obtenons d'autres optimisations. Les clients du Cloud sont en mesure d'engager des ressources de manière automatisée sans que le service informatique soit un obstacle. L'espace de stockage et les ressources sont présentés au travers d'une interface graphique que l'utilisateur peut manipuler de manière à obtenir et à instancier une infrastructure informatique virtuelle. Un navigateur web ou une carte bancaire à titre d'exemple. [7]

#### 1.7.4 Centre de données (Datacenter)

Un centre de traitement de données en anglais « data centre » est un site physique sur lequel se regroupés des équipements constituant du système d'information de l'entreprise (mainframes, serveurs, baies de stockage, équipements réseaux et de télécommunications, etc.). Il peut être interne ou externe à l'entreprise, exploité ou non avec le soutien de prestataires. Il comprend en général un contrôle sur l'environnement (climatisation, système de prévention contre l'incendie, etc.), une alimentation d'urgence et redondante, ainsi qu'une sécurité physique élevée.

Cette infrastructure peut être propre à une entreprise et utilisée par elle seule ou à des fins commerciales. Ainsi, des particuliers ou des entreprises peuvent venir y stocker leurs données suivant des modalités bien définies. [30]



**Figure 1.10:** Exemple d'un Centre des données [43]

### 1.8 Quelle est la différence entre la virtualisation et le Cloud Computing ?

La virtualisation permet aux entreprises d'utiliser une seule pièce de matériel physique, pour effectuer le travail d'un grand nombre de machines. Plusieurs instances d'un système d'exploitation s'exécutant sur un seul dispositif matériel, sont beaucoup plus économiques qu'une pièce de matériel pour chaque tâche de serveur, par contre que le Cloud Computing est

l'accès par Internet aux applications d'entreprise fonctionnant dans un environnement non-locale. Le Cloud Computing peut certainement tirer des avantages de la virtualisation. [15]

### **1.9 Quelle est la différence entre le Grid et le Cloud Computing ?**

Le Grid Computing ressemble à le Cloud Computing dans : grand nombre de serveurs, communication réseau très importante, stockage généralement distribué, etc. Ces deux techniques sont assez différentes.

- Dans le cas du Grid Computing il s'agit surtout de mettre à disposition de quelques équipes des ressources informatiques très puissantes pour des périodes de temps donnés. Par exemple le projet Grid 5000 en France, qu'il est utilisé pour faire des calculs parallèles sur de très gros volumes de données.
- Dans le cas du Cloud Computing on note souvent une notion d'élasticité, de disponibilité et de virtualisation de la ressource.
- Avec le Cloud Computing on ne sait pas trop sur quel(s) serveur(s) fonctionne son application.
- Dans le Cloud on paye en fonction de l'utilisation. [35]

### **1.10 Les avantages du Cloud Computing**

Le Cloud présente de nombreux avantages :

- ✓ Possibilité de partager ses contenus avec son réseau personnel,
- ✓ Possibilité de partager les ressources, [31]
- ✓ Redondance de stockage de données en cas un disque dur tombe en panne, les données restent en sécurité et disponibles automatiquement sur un autre disque,
- ✓ Possibilité de spécifier des ressources,
- ✓ Aucune défaillance : est signifie qu'il n'ya ni de problèmes de matériel, ni de logiciel de la technologie,
- ✓ Sécurité dans Cloud où chaque utilisateur peut être isolé dans son propre environnement,
- ✓ Plus flexible,
- ✓ Augmente la collaboration,
- ✓ Mobilité : l'utilisateur peut à tout moment et à partir de n'importe quel appareil se connecter à ses applications (Any where - any time),
- ✓ Contrôle des coûts.



## 1.11 Les inconvénients de Cloud Computing

- ✓ L'accès au Cloud passe nécessairement par une connexion internet : par conséquent, si votre connexion est coupée, momentanément ou durablement, vous n'avez plus accès à vos documents ou service,
- ✓ Sécurité : la plateforme Cloud, si elle est externe (non installée sur le réseau interne ou avec une ouverture extérieure) doit être suffisamment sécurisée pour éviter le risque d'intrusion, de vol des données par piratage.

## 1.12 Les principaux fournisseurs de Cloud Computing

### 1.12.1 Amazon Web Services (IaaS): [9]

1. **EC2 : Elastic Compute Cloud**, ou comment démarrer/arrêter un nombre quelconque de machines virtuelles Linux ou Windows, en quelques minutes, par Web Services, à Dublin, en Virginie, en Californie, à Singapour et à Tokyo, pour consommer de la ressource Informatique à la demande, en payant à l'heure consommée, sans investissement minimal
2. **EBS : Elastic Block Storage**, ou comment créer et attacher à ces VMs des volumes de taille quelconque (quelques GB à plusieurs TB).
3. **CloudWatch** : monitoring des instances EC2 (% CPU, % RAM utilisés etc.)
4. **SimpleDB** : Base de données dans le Cloud, scalable et sans schéma.
5. **CloudFront** : CDN (Content Delivery Network), réplique géographique de données à proximité des clients Web pour une expérience utilisateur plus réactive.
6. **SQS : Simple Queue Service**, messagerie applicative par queue, pour une communication fiable entre applications, avec récupération d'erreur.
7. **RDS : Relational Database Service**, base de données MySQL ou Oracle scalable dans le Cloud (avec schéma).

### 1.12.2 Amazon Web Services (PaaS):

1. **S3 : Simple Storage Service**, stockage scalable et accessible directement par REST (pas besoin d'EC2).

### 1.12.3 Microsoft Azure (PaaS):

1. **Azure Compute** : hébergement scalable d'applications ASP.NET ou batches .NET / natifs.
2. **Azure Storage** : stockage scalable de blobs dans le Cloud, accessible directement par REST.
3. **SQL Azure** : base de données relationnelle (SQL Server) scalable dans le Cloud.

4. **Service Bus** : ESB dans le Cloud.

5. **Access Control** : fédération d'identité pour SSO dans le Cloud.

#### 1.12.4 Google AppEngine (PaaS):

1. **Cloud Hosting** : hébergement scalable d'applications Python ou Java, avec stockage et base de données plate BigTable.

#### 1.12.5 force.com (PaaS):

1. Hébergement scalable d'applications orientées CRM (SalesForce).

#### 1.12.6 salesforce.com (SaaS):

1. Consommation à la demande de logiciels Web CRM.

#### 1.12.7 Google Apps (SaaS):

1. Version professionnelle de Gmail et Google Apps : suite bureautique en ligne, consommable à la demande.

### 1.13 Sécurité dans le Cloud

La sécurité et la conformité sont souvent citées comme le rétenteur principal à l'utilisation des services Cloud, qu'elles sont les principales préoccupations des informaticiens pour développer cette technologie.

La sécurité permet de garantir la confidentialité, l'intégrité, l'authenticité et la disponibilité des informations. **La confidentialité**

La confidentialité assure que les données d'un client ne soient accessibles que par les entités autorisées. Les différentes solutions de Cloud Computing comportent des mécanismes de confidentialité comme la gestion des identités et des accès, l'isolation ou le cryptage.

La majorité des échanges internes ou externes au Cloud sont encapsulés en SSL (Secure Sockets Layer) et authentifiés avec un certificat [11] rédigé et signé par le client.

Tant que ce certificat assure le contrôle de clé privée de sa client, ce mécanisme permet d'augmenter le degré d'assurance et la confiance par qu'elle est spéciale à les clients autorisés.

#### 1.13.2 L'Intégrité

Les utilisateurs de Cloud peuvent stocker plusieurs données, et généralement doivent être protégés contre le vol et les modifications non autorisées.

Dans le nuage, il existe plusieurs systèmes qui fournissent des différents mécanismes pour protéger l'intégrité des informations.

Par exemple pour le service de stockage de Windows Azure, l'intégrité est définie par les applications utilisant le modèle de contrôle d'accès. Chaque compte de stockage a deux clés qui sont utilisées pour contrôler l'accès à toutes les données dans ce compte de stockage. [9]

### 1.13.3 La Disponibilité

L'un des principaux avantages fournis par des plates-formes de Cloud Computing est la disponibilité robuste basée sur la redondance réalisée avec des technologies de virtualisation. Par exemple Windows Azure offre de nombreux niveaux de redondance fournissant une disponibilité maximale des données et des applications [11], où les données sont répliquées sur trois nœuds distincts pour minimiser les problèmes des pannes matérielles.

## 1.14 La technologie de stockage en réseau

### 1.14.1 Stockage de SAN

#### 1.14.1.1 Définition

Le SAN (Storage Area Network) est une technologie de stockage en réseau. [10] C'est un réseau physique en fibre optique, il connecte l'ensemble des unités de stockages et des serveurs. Dans ce réseau les données stockées sont routées et structurées via des commutateurs.

Cette technologie est basée sur Fibre Channel, qui autorise le transfert de données entre périphériques sans surcharger les serveurs.

Cette matière de stockage permet :

- ✓ Des partages des ressources de stockage entre plusieurs des utilisateurs,
- ✓ L'accès à des informations très rapidement,
- ✓ La résolution de problèmes de connectivité entre plusieurs serveurs et unités de stockage. [10]

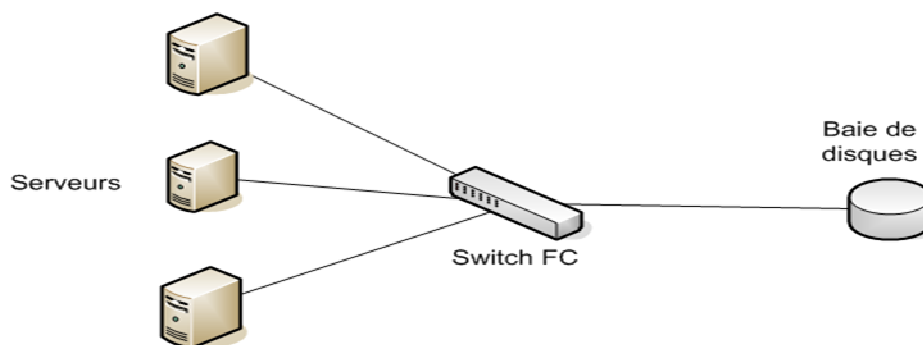
Le SAN est caractérisé par deux propriétés principales suivantes :

- ✓ Utilise un réseau Fibre Channel, où le Fibre Channel est un protocole utilisé par des mainframes,
- ✓ Permet l'installation distante de sous-systèmes de disques durs et de bibliothèques de bandes. [32]

Parmi les topologies de SAN :

1. **Point à Point** : est la plus simple, est un lien croisé entre les deux ports.
2. **Boucle Arbitrée (FC-AL)**: est constitué au maximum de 126 ports reliés entre eux, tous ces ports se partagent la largeur de bande et utilisent un concentrateur FC. [11]

#### 1.14.1.2 L'architecture générale se SAN



**Figure 1.11:** Architecture générale de SAN [46]

Cette architecture se compose par des équipements principaux:

- ✓ **Baie de disques:** est un ensemble des disques pilotés par un ou des contrôleurs qui seront regroupés en VG (Volum Group) grâce à un raid.
- ✓ **Raid:** est signifie (Raid Array of Independant Disks), est une technologie dans le SAN, permettant de constituer une unité de stockage à partir de plusieurs disques regroupés de différents niveaux de raid.
- ✓ **Switch** ou bien commutateur.

#### 1.14.1.3 Les avantages du SAN

- ✓ La consolidation des informations au sein d'un réseau de stockage centralisé,
- ✓ La simplification des sauvegardes et de la restauration. [10]
- ✓ Relie de l'ensemble des ressources de stockage,
- ✓ Performances,
- ✓ Fourni grand volume de capacités de stockage,

#### 1.14.1.4 Les inconvénients du SAN

Malgré tous les avantages de cette technologie, mais il y a quelques inconvénients:

- ✓ Coûteuse: par exemple le prix d'entrée environ de 283 000 € pour un réseau SAN de taille moyenne,
- ✓ Manque de standardisation génère certains problèmes d'interopérabilité, [10]
- ✓ La gestion des composants est plus difficile en fonction de l'augmentation en taille et en complexité du réseau SAN.
- ✓ La complexité du réseau SAN et son expansion augmentent la difficulté de gestion des composants de ce réseau.

## 1.14.2 Stockage de NAS

### 1.14.2.1 Définition

Le NAS (Network Attached Storage) : est un serveur fournit une interface au niveau du fichier pour un stockage pouvant être partagé sur plusieurs systèmes, ce serveur est attachée directement au réseau IP, donc il accessible pour toutes les équipements de ce réseau, généralement le serveur NAS est plus lent que le réseau SAN ou le stockage DAS.

Il utilise un protocole de transport de fichier comme NFS, CIFS et FTP.

Il peut accepter de consolider des données malgré les systèmes d'exploitation sont différents.

Le NAS se distingue par les caractéristiques suivantes :

- ✓ La connexion directe au réseau et permet l'accès des clients aux données sans arriver le serveur d'applications,
- ✓ Il est basé sur le partage de fichiers dans des réseaux hétérogènes par exemple WindowsNT, Unix, Windows 2000,
- ✓ Facile à déployer et à gérer.

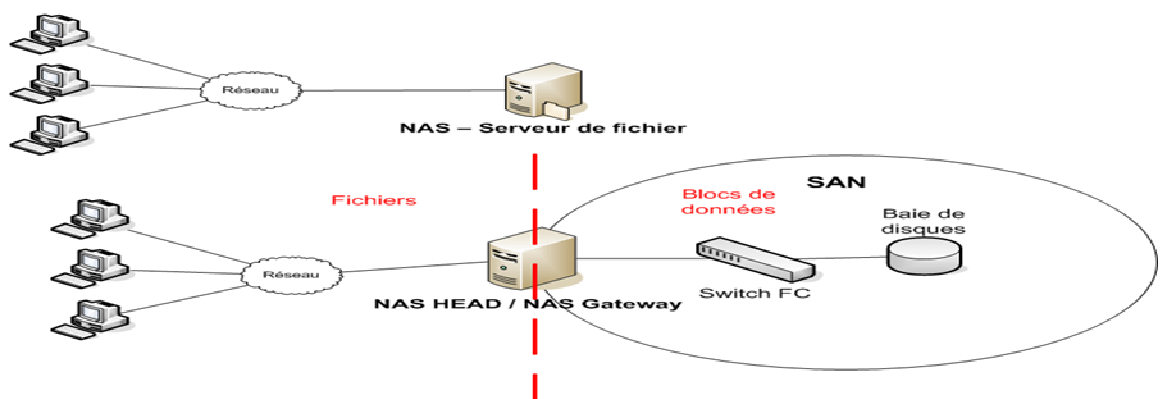


Figure 1.12: Architecture générale de NAS [46]

### 1.14.2.2 Les avantages du NAS

- ✓ Facile à mis en œuvre,
- ✓ Partage les fichiers facilement,
- ✓ Partage multi-environnement lié aux différentes implémentations du protocole que l'on utilise.

### 1.14.2.3 Les inconvénients du NAS

- ✓ Demande des ressources comme CPU,

- ✓ N'est pas valide, en cas des applications qu'ils nécessitent des grosses performances des disques durs.

### 1.14.3 Stockage de DAS

- ✓ Le DAS (**D**irect **A**tached **S**torage) : où chaque disque ou ensemble de disques est physiquement lié à une seule unité de commande, serveur de fichiers ou mainframe.
- ✓ Ces méthodes consistent à connecter, directement ou via un bus d'E/S, une ressource de stockage unique à un hôte dédié, tel qu'un serveur ou une station de travail. [10]

Le tableau suivant résume une comparaison entre les deux types de stockages SAN et NAS:

SAN	NAS
Orienté paquets SCSI	Orienté fichier
Basé sur le protocole Fibre Channel	Basé sur le protocole Ethernet
Le stockage est isolé et protégé de l'accès client général	Conçu spécifiquement pour un accès client général
Support des applications serveur avec haut niveau de performances SCSI	Support des applications client dans un environnement NFS/CIFS hétérogène
Le déploiement est souvent complexe	Peut être installé rapidement et facilement

**Tableau 1.1:** Le stockage SAN et NAS

### 1.15 Conclusion

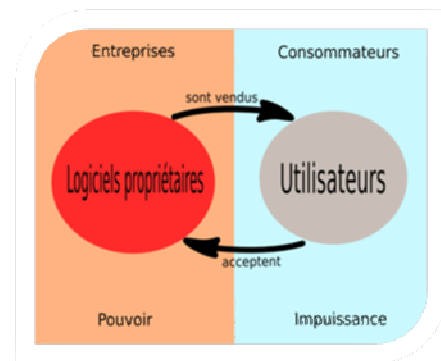
Le Cloud Computing est certainement contribué à réduire les problèmes, qu'ils sont tourmentés les entreprise comme : espace de stockage, réalisation les différents applications et la disponibilité de données.

Donc le Cloud est un modèle de distribution et de partage de ressources à grand échelle, qu'il offre à les utilisateurs un environnement plus simple et plus efficace pour travaille, parmi d'intérêt de Cloud : facilité de travaille, augmente la collaboration, réduire le coût de matériel et évité le déplacement de client pour obtenir un donnée.

On peut exploite les bénéfices de Cloud a partir des différent solutions (plate-formes), qui nous presenterons dans le chapitre suivant.

# *Chapitre 2 : Les plateformes de Cloud Computing*

## *Open Source et Commercial*



## 2.1 Introduction

Un intérêt de l'infrastructure est croissant parce que ce dernier fourni à les entreprises les différents outils de gestion et développement leur applications à grâce des plateformes de Cloud Computing qui gèrent cet infrastructure.

Ces plateformes permettent d'accéder facilement à différents services fournis par le Cloud, sans se soucier comment faire le stockage des données et la façon de sécurité.

Dans le monde de Cloud Computing, il y a deux type : les plate-formes open source, comme Eucalyptus et OpenStack et les plate-formes propriétaire et commercial comme Windows Azure et vCloud de VMware.

Dans ce chapitre nous présentons quatre plateformes de Cloud Computing. Où nous commenceront par définition de chaque plate-forme, avec leurs caractéristiques et composants.

## 2.2 Les Open Source

Beaucoup des solutions sont avant tout des solutions d'infrastructures permettant une gestion simplifiée d'architectures matérielles, les plus complexes. Dans ce domaine l'open source joue un rôle important dans les solutions de Cloud Computing. Voici quelques plateformes open source :

### 2.2.1 Eucalyptus

#### 2.2.1.1 Définition

Eucalyptus est signifie «**Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems**» en Français «Utilitaire d'Architecture informatique élastique pour relier vos programmes à des systèmes fonctionnels ».

Eucalyptus est une plate-forme open source de Cloud Computing, et a été développé par une équipe de chercheurs de l'université de Californie à partir de 2007, cette solution est la plus connue, car elle est intégrée dans les distributions Ubuntu Server.

Elle est compatible à Amazon Web Services, qui permet créer des Clouds IaaS de type privé ou hybride, elle supporte l'hyperviseur ESX de VMware , ainsi que les hyperviseurs Xen et KVM.

#### 2.2.1.2 Architecture d'Eucalyptus

L'architecture d'Eucalyptus est composée de cinq éléments de haut-niveau :

##### 2.2.1.2.1 Contrôleur de nœud (NC)

Le rôle du nœud est d'héberger le KVM, il sert ainsi d'hyperviseur pour les machines virtuelles qui sont déployées. Les machines virtuelles fonctionnant sur l'hyperviseur sont

---



appelées des « instances ». Eucalyptus permet aussi d'utiliser d'autres types d'hyperviseurs comme XEN, mais Canonical a fait le choix de privilégier KVM.

Le contrôleur de nœud fonctionne sur chaque nœud et est chargé de vérifier le cycle de vie des instances en cours d'exécution sur le nœud. Il interagit avec le système d'exploitation et l'hyperviseur en cours d'exécution sur le nœud d'un côté et le contrôleur de cluster (CC) de l'autre côté. Le contrôleur interroge le système d'exploitation s'exécutant sur le nœud afin de découvrir les ressources physiques du nœud (le nombre de cœurs, la taille de la mémoire, l'espace disque disponible et aussi de s'informer sur l'état des instances VM en cours d'exécution sur le nœud et propage ces données au contrôleur de cluster).

#### 2.2.1.2.2 Contrôleur de cluster (CC)

Ce contrôleur sert à déployer et gérer les différents contrôleurs de nœuds. Il sert également à gérer la mise en place du réseau entre les instances des différents nœuds. C'est lui qui communique l'ensemble des informations au contrôleur du Cloud.

Il a 4 fonctions principales :

- ✓ Il reçoit les requêtes de déploiement des instances du contrôleur de Cloud,
- ✓ Il décide sur quel contrôleur de nœuds les instances seront déployées,
- ✓ Il contrôle le réseau virtuel entre les instances,
- ✓ Il collecte les informations des contrôleurs de nœuds enregistrés.

#### 2.2.1.2.3 Walrus

C'est le composant qui gère l'accès aux services de stockage. En fait, Walrus peut être considéré comme un simple système de stockage de fichiers.

#### 2.2.1.2.4 Contrôleur de stockage (SC)

Ce composant fonctionne avec le composant Walrus et permet de stocker les images des machines virtuelles et les données des utilisateurs. Il est similaire au service « Elastic Block Storage (EBS) » d'Amazon.

Il a donc 3 fonctions essentielles :

- ✓ La création de dispositifs EBS persistants,
- ✓ Fournir le système de stockage de blocs aux instances,
- ✓ Autoriser la création d'images des volumes pour permettre leurs sauvegardes. [16]

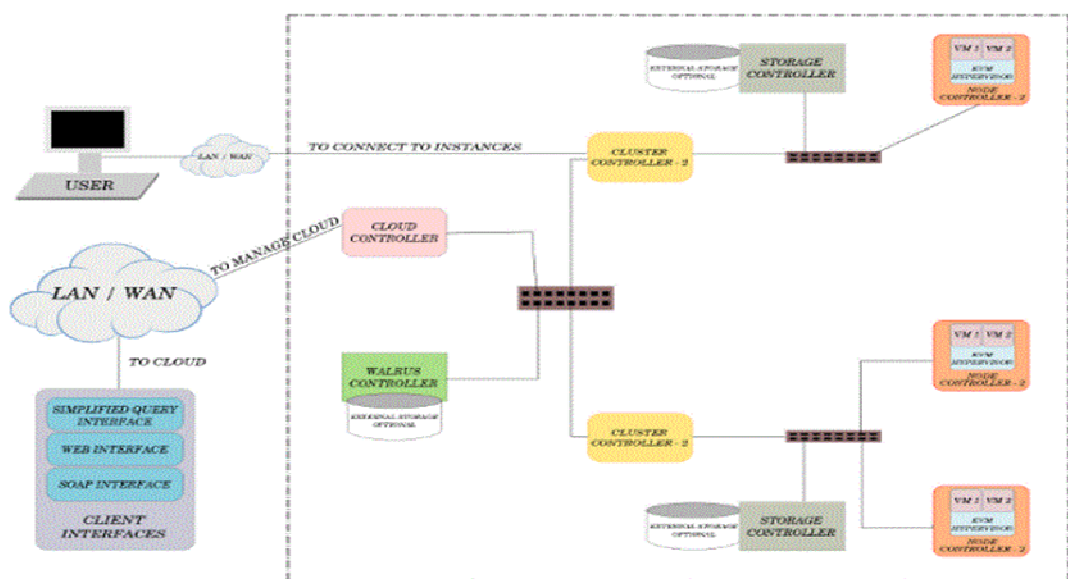
#### 2.2.1.2.5 Contrôleur du Cloud (CLC)

C'est l'unique point d'entrée "Front end" pour tous les utilisateurs et les administrateurs d'Eucalyptus. [17] Il est responsable de la gestion de tout le système, permet d'interagir avec les

autres composants de l'infrastructure, et donnée la possibilité à les utilisateurs de gérer certains aspects de l'infrastructure via une interface Web.

Il a 3 rôles principaux :

- ✓ Surveiller la disponibilité des ressources sur les différentes composantes de l'infrastructure du Cloud,
- ✓ L'arbitrage des ressources : il est décidé quel cluster sera utilisé pour la virtualisation des instances,
- ✓ Monitorer les instances en cours d'exécution. [16]



**Figure 2.1:** Architecture d'Eucalyptus [16]

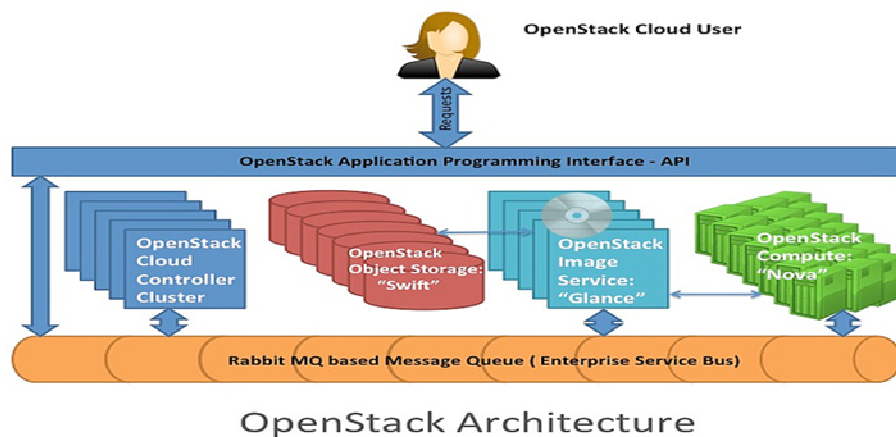
### 2.2.1.3 Fonctionnalités d'Eucalyptus

- ✓ Compatible avec Amazon Web Services API offrant par exemple la fonctionnalité d'auto-scaling.
- ✓ Installation et déploiement avec l'outil de gestion de grappe de serveurs Rocks Linux
- ✓ Communication sécurisée entre les processus interne via SOAP et WS-Security
- ✓ Outils d'administration basique
- ✓ Capacité à configurer de multiples grappes de serveurs comme un seul «Cloud»

## 2.2.2 Plateforme OpenStack

### 2.2.2.1 Définition

OpenStack est un projet open Source de Cloud Computing privé et public. Il est développé par la NASA "National Aeronautics and Space Administration", qui est l'agence gouvernementale qui a en charge la majeure partie du programme spatial civil des États-Unis, et Rackspace Cloud, un fournisseur de plate-forme de Cloud Computing depuis juillet 2010. Rackspace fournit les outils de gestion des fichiers dans le cloud, tandis que la NASA apporte les outils de gestion de l'IaaS. Ces deux sociétés ont ensuite été rejointes par Cloud.com, Citrix Systems, Dell, Cloudkick, Intel, et Cisco pour le développement d'OpenStack. Cette plate-forme est intégrée dans les distributions Ubuntu Server. L'objectif d'OpenStack est de rendre le Cloud simple à mettre en œuvre et très extensible.



**Figure 2.2:**Architecteur d'OpenStack [42]

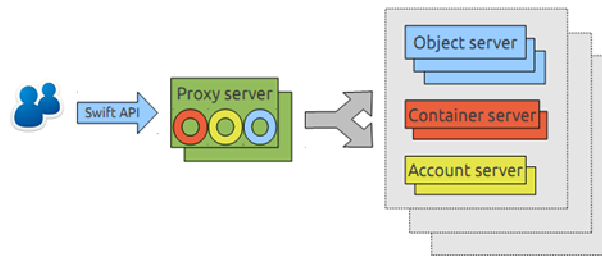
### 2.2.2.2 Composants d'OpenStack

L'architecture d'OpenStack est composée de trois éléments sont :

#### 2.2.2.2.1 Swift

Il permet de gère et de stocker une large capacité de données dans le Cloud avec une redondance pour assurer de la tolérance aux pannes, ainsi que le basculement entre les différents objets de stockage.

L'Architecture de Swift est compose par ces éléments suivant: [18]



**Figure 2.3:**Architecture de Swift [18]

### 1. Proxy Server

- ✓ Partie visible (public) de Swift,
- ✓ Détermine le nœud de stockage approprié,
- ✓ Coordonne les réponses.

### 2. L'anneau (Ring)

- ✓ Lie les requêtes au nœud de stockage,
- ✓ Gère les zones de disponibilité,
- ✓ Extensible sans affecter les autres entités.

### 3. Serveurs de stockage : Comptes et Containers

- ✓ Base de données SQLite,
- ✓ Les groupes de containers et les objets sont contenus dans les comptes,
- ✓ Schéma simple : table pour les listes et table pour les métadatas.

### 4. Serveurs de stockage : Objet

- ✓ Utilise la file system pour stocker les fichiers,
- ✓ Fichiers nommés avec un marqueur de temps.

### 5. Serveurs de consistance

- ✓ Réplicats,
- ✓ Gestion des mises à jour,
- ✓ Gestion des audits.

Les principales fonctionnalités sont

- ✓ Stockage d'objets,
- ✓ Sauvegarde et Archivage,
- ✓ Pas de limite de taille d'objet,
- ✓ Données répliquées 3x (« zones »),
- ✓ Stable et déployé en production.

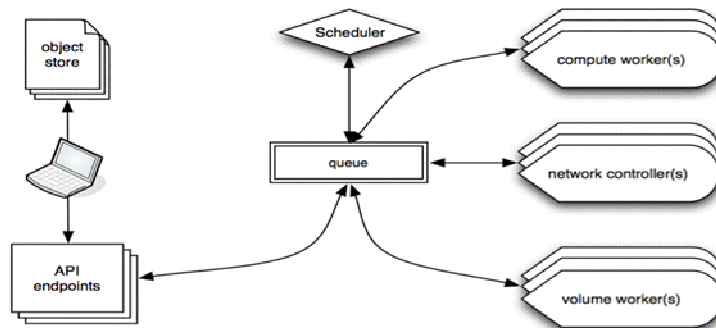
Et ses cas d'utilisation

- ✓ Sauvegarde,
- ✓ Contenus web (petits fichiers, beaucoup d'écritures, énormément de lecture),
- ✓ Régies publicitaires,
- ✓ Gestion de documents,
- ✓ Imagerie médicale.

#### 2.2.2.2.2 Nova Compute

Nova est le contrôleur de toutes les activités liées aux machines virtuelles, il permet de gérer et configurer le réseau de ces dernières.

L'Architecture de Nova est composée par ces éléments suivants :



**Figure 2.4:** Architecture Nova [23]

1. **Nova-API** (Serveur API): Il fournit une interface externe pour gérer et interagir dans l'infrastructure de Cloud. Ce serveur communique avec les composants à l'aide du Message Queue.
2. **RabbitMQ-Server** (serveur Message Queue) : est la manière de communication entre les composants d'OpenStack, via un protocole AMQP (Advanced Message Queue Protocol), le but de ce protocole est de standardiser les échanges des messages entre les serveurs d'OpenStack.
3. **Nova-Compute** (Compute Workers) : est responsable de la création et manipulation des machines virtuelles (Serveur Virtuelles), il reçoit les demandes par le Message Queue.
4. **Nova-Network** (Network Services) : il est appelé aussi le contrôleur de réseau, est en charge de la configuration réseau des machines hôtes. Il alloue des adresses IP, la configuration des VLAN pour les projets.

**5. Nova-Scheduler** (Scheduler) : est un planning, sa principe est le choix d'un serveur pour réalise certaine demande, qu'elles sont envoyées à Scheduler par le Message Queue. Scheduler fonctionne par des algorithmes pour assurer un fonctionnement optima, aussi il prend des décisions en fonction de plusieurs paramètres comme la charge, la mémoire, ou la disponibilité de zone, etc.

Il est basé sur trois choix d'ordonnancement :

- Simple : choisir le serveur le moins chargé pour exécuter l'instance,
- chance: choisit un serveur disponible au hasard,
- Zone : choisit un serveur au hasard depuis une zone disponible.

**6. Nova-Volume** (Volume Worker): est utilisé pour le management des volumes LVM (Logical Volume Manager) des machines virtuelles. Il effectue les opérations relatives aux volumes: création, suppression, attachement et détachement d'un volume à une machine virtuelle. Les volumes permettant de rendre les données des VMs persistantes.

Les principales fonctionnalités sont

- ✓ Management du cycle de vie d'une instance,
- ✓ Management des ressources de calcul,
- ✓ Management du réseau et des autorisations,
- ✓ Communication Asynchrone.

### 2.2.2.2.3 Glance

Est un système fournit les services de stockages, de découvertes, d'enregistrements, et de distributions pour les images disques des machines virtuelles.

Les principales fonctionnalités sont :

- ✓ Multiples formats de disque: raw, VHD, vmdk, vdi, qcow2, ISO, aki, ari, ami
- ✓ Multiples formats de conteneurs: ovf, bare. [16]

L'Architecture de Glance est compose par ces éléments suivant:

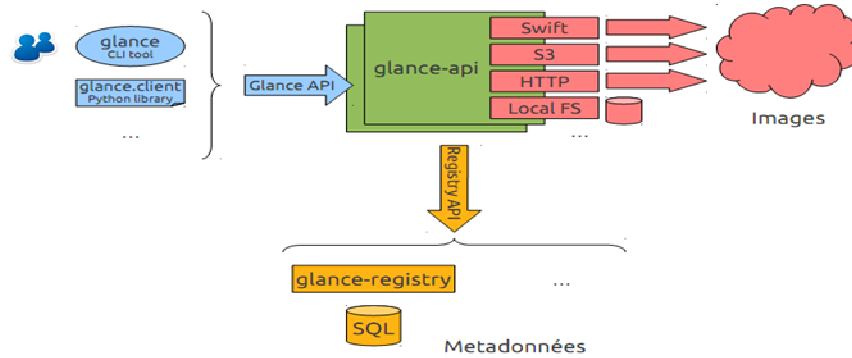


Figure 2.5 : Architecture de Glance [23]

#### 2.2.2.2.4 OpenStack Horizon (OpenStack Dashboard)

Est fournit aux administrateurs et aux utilisateurs une interface graphique pour accès aux différents service, il facilite l'intégration avec les ressources openstack.

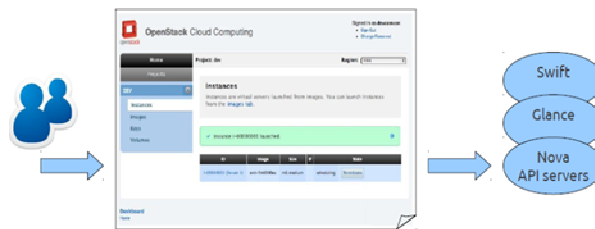
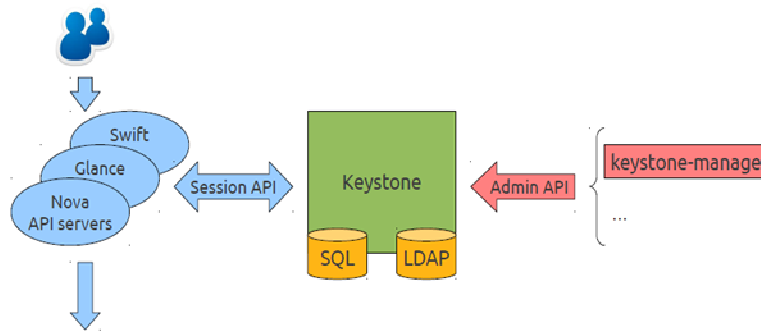


Figure 2.6: Architecture OpenStack Horizon [19]

#### 2.2.2.2.5 OpenStack Keystone (OpenStack Identity)

Est un projet OpenStack, qui fournit le service d'identité pour l'authentification, l'autorisation, Token aussi les services catalogue et politique pour une utilisation spécifique par les autres projets d'openStack.

Authentification dans Keystone, est la confirmation de l'identité d'un utilisateur. keystone faire la authentification à traverse un ensemble des affirmations spéciale telle que nom d'utilisateur et mot de passe. Après la fin de la authentification, keystone donnée à l'utilisateur un jeton pour démontrer que leur identité a été authentifiée.



**Figure 2.7:** Architecture OpenStack Keystone [19]

#### 2.2.2.2.6 OpenStack Quantum (virtualisation de réseau)

- ✓ Gestion d'un réseau virtuel au sein d'OpenStack,
- ✓ Fournit un bloc complet pour la gestion de réseaux complexes dans les infrastructures Cloud,
- ✓ Fourni une connectivité réseau entre une sélection d'interfaces réseaux (interface virtuelle d'un service Compute, interface sur un service de loadbalancing...),
- ✓ Expose des API décrivant la connectivité réseau et la configuration entre les interfaces.

#### 2.2.2.2.7 OpenStack Storage (cinder)

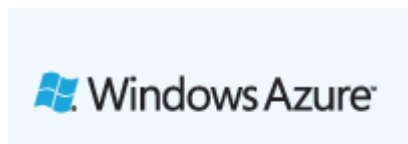
Est fournit des services de gère et accéder aux volumes de stockage de blocs pour une utilisation par les instances de machines virtuelles.

### 2.3 Plateforme propriétaires de Cloud Computing

Dans cette section, nous citons les plate-formes commerciaux les plus utilisés dans le monde du Cloud Computing.

#### 2.3.1 Plateforme Microsoft Windows Azure

##### 2.3.1.1 Définition



Microsoft Windows Azure est le système d'exploitation qui sert d'environnement de développement, d'exécution et de contrôle pour la plateforme Windows Azure.

Plateforme Microsoft Azure est une infrastructure commerciale et flexible de Cloud Computing, il est crée par le group de Microsoft, qui permet aux utilisateurs créer et



développer des applications directement dans le Cloud Computing, héberger, administrer des services Web à travers un réseau global, il permet aussi la simplicité de la maintenance et l'exploitation des applications, et de stockage à la demande en proposant des outils communs.

### 2.3.1.2 Caractéristiques

- ✓ L'utilisateur se concentre sur son application et sa valeur fonctionnelle, [32]
- ✓ Microsoft s'occupe de la gestion de l'infrastructure, qui est disponible à la demande pour les applications,
- ✓ La facturation est réalisée à l'usage,
- ✓ De nombreuses fonctions sont disponibles, immédiatement activables : VPN, réplication des données, Sécurité, Authentification multiple,
- ✓ Les applications et données sont localisées, répliquées, hautement disponibles, scalables,
- ✓ Une place de marché permet la vente et l'achat de données et de composants logiciels.

### 2.3.1.3 Les composants de plate-forme de Microsoft Windows Azure

#### 2.3.1.3.1 AppFabric :

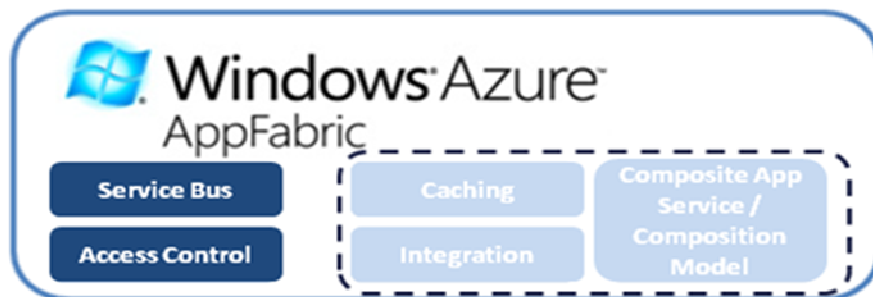
Il réalise le premier rôle de la plate-forme. C'est la plate-forme de développement des applications entreprises qui seront externalisées vers le cloud où l'intégration entre les serveurs déployés en entreprise et le Cloud.[21]

Les applications issues de cette plate-forme seront facilement administrées dans l'IaaS. C'est grâce à l'AppFabric que la plate-forme Azure est considérée dans la littérature comme un PaaS.

Il se compose par les éléments suivants :

1. **Service Bus** : est un bus logiciel sur lequel des points de connexion (endpoints) permettent de connecter soit des services, soit des clients. Chaque point de connexion est caractérisé par une URI qui l'identifie de manière unique dans Azure, Il supporte les connexions en full duplex. Ce bus logiciel joue aussi le rôle de relais de messages. Cette fonctionnalité permet à un service applicatif, hébergé dans votre système d'information, d'être exposé au monde extérieur par l'intermédiaire d'Azure.
2. **Access Control Services (ACS)** : Ce service permet d'authentifier des applications et d'accréditer un utilisateur humain ou logiciel. Cette accréditation est formalisée par l'obtention d'un jeton d'autorisation (token) associé à un ensemble d'informations ou de demandes (claims) permettant à une application de déterminer les autorisations effectives accordées à ce jeton. [22]

- 3. Caching:** cache distribué en mémoire vive, utilisable pour stocker des objets sérialisables. Cela sert par exemple pour la session ASP.NET dans le cadre d'une ferme Web (un Web Rôle).Caching présente plusieurs avantages pour les développeurs d'application. Par exemple il augmente les performances en stockant temporairement les informations depuis d'autres sources principales, aussi il peut réduire les coûts et accroître l'extensibilité des autres services de stockage, tels qu'une base de données SQL ou le stockage Azure. [39]
- 4. Composite App Service et Composition Model :** Composite App Service et Composition Model fournissent un environnement de développement pour faciliter la création, la gestion et le déploiement d'applications composites.
- Le Composition Model permettra aux développeurs de composer et de préparer le déploiement d'une application composite directement depuis Visual Studio.
- Composite App Service permet de gérer les applications composites, d'automatiser le déploiement, et la gestion et le monitoring de ces applications de façon automatique. [38]



**Figure 2.8:** Architecture Windows Azure AppFabric [38]

### 2.3.1.3.2 Windows Azure :

Il réalise le 2eme rôle de la plate-forme. C'est lui qui déploie et exécute les VM dans l'IaaS, il se divise en trois composants :

#### 1. Fabric :

Ce composant est constitué d'une collection de serveurs, chaque serveur faisant tourner plusieurs VMs, qui sont utilisés selon les besoins, pour représenter une instance Compute ou Storage.

La Fabric contient un composant de contrôle (Fabric Controller) conçu pour le système de virtualisation Hyper-V, qui va se charger des opérations sur les VMs, que ce soit la disponibilité, les libérer, etc. [38]

## 2. services Compute :

Ce composant permet de délivrer des machines virtuelles spécialisées et permettant l'exécution d'applications avec une haute disponibilité.

Il y a trois rôles, qui vont, selon les besoins, comme Web Rôle pour exécuter des applications Web dans le serveur Internet Information Services (IIS), Worker Rôle pour faire tourner les services Windows et le VM Rôle qui permettra d'obtenir une VM, basée sur un modèle choisi à la création, pour rediriger le trafic en interne de façon transparente pour les utilisateurs finaux.

## 3. services Storage :

Permettent le stockage sécurisé des données non relationnelles. Il propose trois méthodes de stockage sont les suivantes:

- ✓ Les blobs qui permettent de stocker de grands objets binaires tels que des vidéos et des images. [23]
- ✓ Table : une table représente un ensemble d'entités, chaque entité ayant un ensemble de propriétés. Ces tables sont non relationnelles, et peuvent stocker des types hétérogènes.
- ✓ Queue : permettre aux développeurs de gérer des communications asynchrones entre les différents composants de l'application.

La fonctionnalité Drive permet de configurer l'équivalent d'un disque virtuel qui peut ensuite être utilisé comme un disque dur réseau par un rôle Azure. Cette fonctionnalité est surtout prévue pour faciliter le portage d'une application existante sur le Cloud. [34]



Figure 2.9 : Les composants au cœur de Windows Azure [20]

### 2.3.1.3.3 SQL Azure :

C'est le système de stockage et gestion de base de données d'Azure. Il est possible d'accéder à ces données de la même façon que l'on accéderait à une base de données SQL Server classique, il se divise en deux composants :

1. **SQL Azure Data Sync** : permet de synchroniser les données entre plusieurs bases de données SQL Azure, à l'intérieur d'un centre de données.
2. **SQL Azure Reporting** : est un service de création de rapports informatique qui repose sur les technologies Windows Azure et SQL Server Reporting Services. Vous pouvez publier, afficher et gérer des rapports opérationnels contenant des données de sources de données des Base de données SQL. [39]



Figure 2.10: Architecteur SQL Azure [38]

### 2.3.1.3.4 Windows azure Marketplace :

Est une place de marché en ligne pour les professionnels de l'informatique permettant de partager, acheter et vendre des composants et des données, pour construire les différents services et applications et profiter de la plate-forme Windows Azure.

## 2.3.2 vCloud de VMware



vCloud développé par la société VMware, ce logiciel permet de créer et gérer votre Cloud privé en manière simple, il regroupe les ressources informatiques avec davantage d'efficacité pour offrir des services plus flexibles et plus dynamiques que les utilisateurs peuvent déployer automatiquement et il donne la possibilité de créer des machines virtuelles selon des besoins des utilisateurs.

Il permet aussi:

- ✓ La possibilité de transformer simplement les serveurs physiques en serveurs virtuels.

- ✓ Augmenter les ressources de votre Cloud privé sans interruption de service.
- ✓ Réduire facilement vos budgets IT.

### 2.3.1.4 Les composants de plate-forme de Microsoft Windows Azure

#### 2.3.1.4.1 vSphere

Une plate-forme pour l'infrastructure de virtualisation et de Cloud, il gère de grandes collections d'infrastructure (par exemple des processeurs, le stockage et la gestion de réseau) sous la forme d'un environnement d'exploitation transparent et dynamique, ainsi que la complexité d'un centre de données. [24]

La VMware vSphere est constituée des couches de virtualisation, de gestion et d'interfaces:

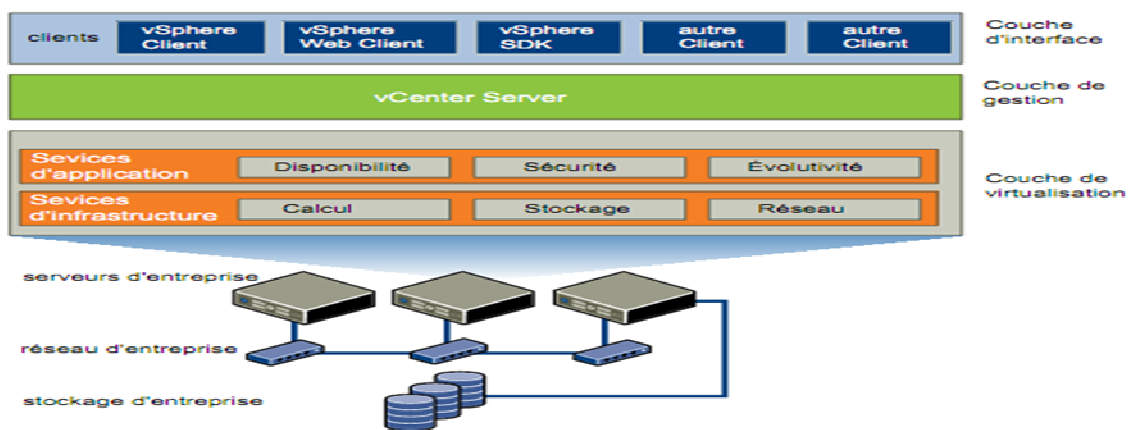


Figure 2.11: Les composants de VMware vSphere [24]

#### ➤ Couche de virtualisation

La couche de virtualisation de VMware vSphere inclut des services d'infrastructure et d'application.

Les services d'infrastructure comprennent les types suivants :

1. services de traitement (Calcul) Inclut les fonctions VMware qui permettent de ne pas tenir compte des ressources serveur hétérogènes sous-jacentes. Les services de traitement agrègent ces ressources sur un grand nombre de serveurs discrets les affectent aux applications.
2. Services de stockage Il s'agit de l'ensemble de technologies qui permettent d'utiliser et de gérer efficacement le stockage dans les environnements virtuels.
3. Services de réseau Il s'agit de l'ensemble de technologies qui simplifient et améliorent la gestion de réseau dans les environnements virtuels.

Les services d'application sont l'ensemble de services qui assurent la disponibilité, la sécurité et l'évolutivité des applications. Par exemple (HA) vSphere High Availability.

➤ **Couche de gestion**

VMware vCenter Server est le point central de la configuration, du provisionnement et de la gestion des environnements informatiques virtualisés.

➤ **Couche d'interfaces**

Les utilisateurs peuvent accéder au centre de données VMware vSphere via des clients à interface graphique, tels que vSphere Client ou vSphere Web Client. En outre, ils peuvent accéder au centre de données via des machines clientes qui utilisent des interfaces de ligne de commande et des kits SDK pour la gestion automatique. [24]

❖ **Composants et fonctions VMware vSphere**

VMware vSphere inclut les composants et fonctions suivants:

1. **VMware ESXi:** est une couche de virtualisation fonctionne sur des serveurs physiques qui analysent le processeur, la mémoire, le stockage et les ressources dans les machines virtuelles multiples.
2. **VMware vCenter Server:** est le point central pour configurer, approvisionner et gérer des environnements informatiques virtualités. Il fournit les services essentiels du centre de données : contrôle d'accès, surveillance des performances et gestion des alarmes.
3. **VMware vSphere Client:** est une interface permettant aux utilisateurs de se connecter à distance vCenter Server ou ESXi depuis n'importe quel PC Windows.
4. **VMware vSphere Web Client:** est une interface Web permet aux utilisateurs de se connecter à distance à vCenter Server depuis divers navigateurs Web et systèmes d'exploitation.
5. **VMware vSphere SDKs:** est une fonction qui fournit des interfaces standard pour les solutions VMware pour accéder à VMware vSphere.

❖ **Topologie physique de vSphere centre de données:**

Un centre de données VMware vSphere est constitué des composants suivants:

1. **Serveurs informatiques:** Serveurs standard x86 exécutant ESXi sans système d'exploitation. [24]
2. **réseaux IP:** Chaque serveur informatique peut avoir plusieurs cartes réseaux pour fournir une haute bande passante et une gestion de réseaux fiable à tout le centre de données VMware vSphere. [25]

3. **Clients de gestion:** VMware vSphere offre plusieurs interfaces pour la gestion du centre de données et l'accès à la machine virtuelle. Ces interfaces comportent VMware vSphere Client (vSphere Client), un accès web via le navigateur web, vSphere Command-Line Interface (vSphere CLI). [24]
4. **vCenter Server:** fournit un point de suivi unique du Datacenter. Il offre des services principaux du centre de données (la commande d'accès, la surveillance de performance et la configuration). Il unifie les ressources à partir des serveurs informatiques individuels à partager entre les machines virtuelles dans tout le centre de données. Pour réaliser cette unification vCenter gère l'attribution des machines virtuelles aux serveurs informatiques et l'attribution des ressources aux machines virtuelles dans un serveur informatique donné en fonction des stratégies que l'administrateur système définit. Les serveurs informatiques continuent de fonctionner même dans l'éventualité improbable où vCenter Server deviendrait inaccessible (par exemple si le réseau est coupé). Les serveurs peuvent être gérés séparément et continuent d'exécuter les machines virtuelles qui leur sont affectées selon la dernière affectation de ressources réglée. Une fois la connexion au vCenter Server restaurée, il peut de nouveau gérer le centre de données.
5. **Réseaux et baies de stockage:** Les modules SAN Fibre Channel, SAN iSCSI et NAS sont des technologies de stockage communément utilisées par VMware vSphere pour répondre aux besoins en stockage du centre de données. Les baies de stockage sont connectées et partagées entre groupes de serveurs dans les réseaux de zone de stockage. Cette option permet de regrouper des ressources en stockage et offre plus de souplesse pour leur fourniture aux machines virtuelles.

#### 2.3.1.4.2 vFabric

VMware vFabric est une plate-forme économique, tout particulièrement adaptée aux besoins de création et d'exécution des applications modernes dans des environnements physiques, virtuels de Cloud Computing. [23]

Malgré elle est caractérisée par des qualités telles que coûteux, infrastructures complexes ....etc. Il permet de créer facilement et rapidement des applications qui évoluent tout en souplesse et livrent grand quantité de données de manière accélérée et fiable.

➤ **Avantages**

- ✓ Créez et exécutez des applications qui évoluent tout en souplesse.
- ✓ Obtenez une densité de machines virtuelles plus forte et des taux d'utilisation renforcés sur le matériel existant.
- ✓ Exécutez des applications dans votre propre Cloud privé ou sur des services de Cloud Computing public. [40]
- ✓ Déploiement et gestion en environnements de Cloud hybrides.
- ✓ Gérer les états critiques de trafic imprévisibles et éliminer de bases de données incapables de s'adapter rapidement à la demande.

➤ **Fonctionnalités clés**

- 1. Provisionnement automatisé:** vFabric automatise le déploiement et autorise le provisionnement en libre-service d'applications dans des environnements de Cloud.
- 2. Plate-forme d'exécution agile et légère.**
- 3. Gestion globale des données:** la plate-forme de gestion des données distribuées de VMware vFabric exploite toute la puissance du traitement en mémoire avancé pour offrir ce que les bases de données traditionnelles sont incapables de faire, à savoir la distribution évolutive et élastique des données en temps réel et la mise en cache pour les applications virtuelles et de Cloud.
- 4. Messagerie sur le Cloud:** vFabric intègre RabbitMQ, pour garantir la haute disponibilité, l'évolutivité et la portabilité des applications basées sur le Cloud.
- 5. Base de données sous forme de service (DBaaS):** vFabric assure une virtualisation orientée base de données sur vSphere, ainsi que la gestion en libre-service du cycle de vie de bases de données hétérogènes. [41]

## 2.4 Conclusion

Nous avons présenté dans ce chapitre une idée générale sur les plate-formes de Cloud Computing, ses architectures et ses fonctionnalités, y compris les plate-formes open sources et propriétaires, ainsi que les techniques de virtualisation utilisées. Afin d'aider un simple utilisateur non expert de choisir l'un des ces solutions nous essayons dans le chapitre suivant de faire une comparaison détaillée sur deux plate-formes libre (Open source) qui sont OpenStack et Eucalyptus.



## *Partie II. Nos contributions*

# *Chapitre 3 : Mise en place d'OpenStack et Eucalyptus*



### 3.1 Introduction

Dans les chapitres précédents, nous avons expliqué le principe et le fonctionnement des plateformes qui gèrent l'infrastructure virtuelle de cloud computing.

Pour la mise en place de cette solution, nous avons choisi en particulier les solutions open sources Eucalyptus et OpenStack pour l'installer et faire une comparaison entre eux.

Pour réaliser de cette étude on va installer l'hyperviseur VMware vSphere server, système d'exploitation Ubuntu server version 12.04 dans le serveur et VMware vSphere client dans notre machine physique.

### 3.2 Environnement d'installations

Nous sommes installés les deux plateformes sous une infrastructure de cloud computing déployés au sein d'un centre de données au niveau de l'université Kasdi Merbah d'Oran.

Alors, les différents nœuds sont installés sur des machines virtuelles à des caractéristiques différentes selon les pré-requis de chaque nœud, dans la suite nous présentons une description détaillée de cette infrastructure.

#### 3.2.1 Serveur

Si le serveur machine qui nous déployons les différents VMs est caractérisé par ;

- Fabricant : HP.
- Modèle : ProLiant ML 370 G6.
- Cœurs de CPU : 4 CPU \*2.266 GHz.
- Type processeur : Intel(R) Xeon ( R) CPU.
- Noyaux par socket : 4.
- Processeurs logiques : 8.
- Nombre cartes NIC : 4.
- Hôte disponibilité ; HA.
- Hôte configuré pour tolérance aux pannes : FT.
- Hyperviseur : ESXI-5.0.0-20120302001.
- Capacité de stockage total : 268,50 GO.

#### 3.2.2 VMware vSphere

- Hyperviseur : ESXI-5.0.0-20120302001.
- VMware vSphere Client version 5.1.

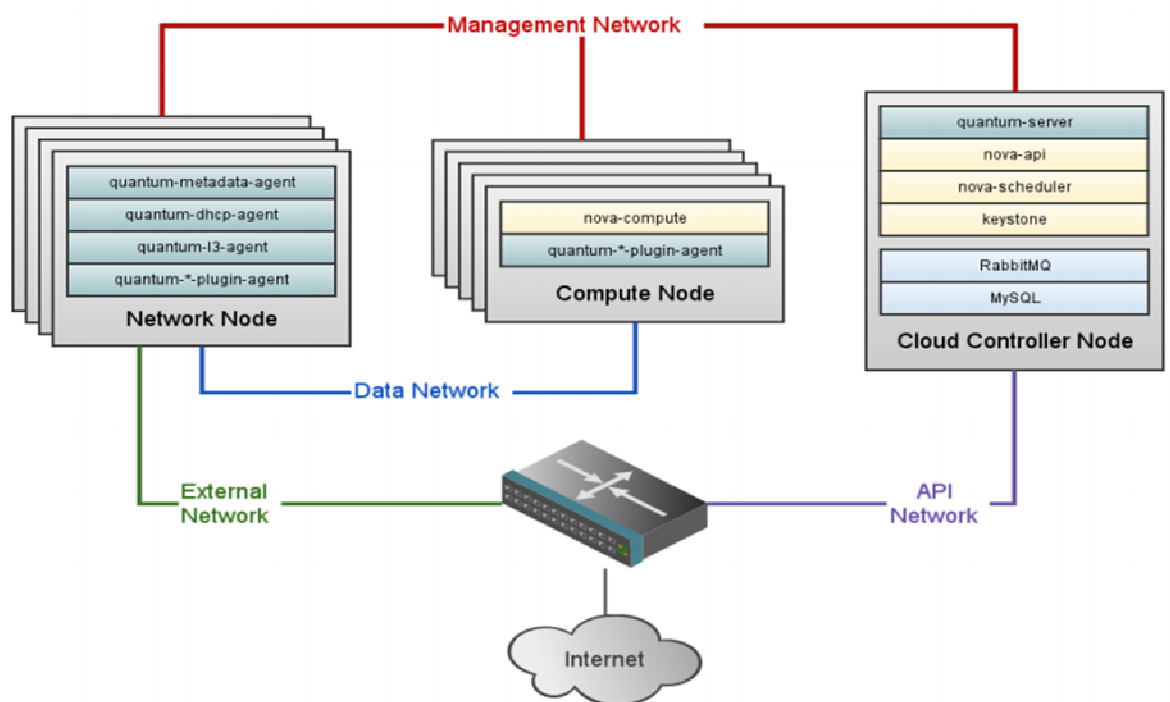
### 3.2.3 Les machine virtuelles

Nous créons quatre machines virtuelles, trois pour OpenStack et une pour Eucalyptus, chacune ayant les caractéristiques suivantes :

- IOS : Ubuntu Server 12.4 LTS.
- CPU ; 2 CPU \*2.266 GHz.
- RAM ; 2 Go.
- DD : 20 Go.
- Nombre cartes NIC : 2.

### 3.3 Installation d'OpenStack

Nous installons la version 7 d'OpenStack, qui appelé Grizzly. Il nécessite trois machines : Cloud Controller Node, Compute Node et Network Node.



**Figure 3.1:**L'architecture OpenStack pour l'installation [45]

Avant l'installation de plateforme OpenStack, nous expliquons les composants principaux suivants:

#### ➤ Cloud Controller

On française "le Contrôleur de Cloud" est le serveur principal dans Openstack, qui offre toutes les fonctionnalités de Cloud. Ce serveur regroupe les composants Glance, Keystone, Cinder, MySQL, Quantum, RabbitMQ, Nova pour gérer l'administration et la configuration d'infrastructure (instances virtuelles, réseau).

➤ **Network Controller**

On française "le Contrôleur de réseau", est fournit les services de réseau dans OpenStack tels que DHCP, est un protocole permet à un ordinateur qui se connecte sur un réseau d'obtenir dynamiquement sa configuration réseau son but est la simplification d'administration d'un réseau.

➤ **Compute Node**

On française "le Nœud de calcul", ce serveur gère les hyperviseurs compatibles avec l'openstack comme KVM ou Xen, il sera l'hôte des machines virtuelles réels.

➤ **Management network**

On française " Gestion de réseau", est utilisé pour la communication interne sur le réseau entre les composants OpenStack, via des adresses IP devrait être accessible dans le datacenter.

➤ **Data network**

On française " Réseau de données", est utilisé pour la communication de données des machines virtuelles dans le déploiement de Cloud (privé, public et hybrid).

Les besoins d'adressage IP de ce réseau, dépendent du Quantum Plugin.

Le rôle de ce dernier est de traduire les demandes de changement de réseau reçues du Service API.

➤ **External network**

On française " Réseau externe", est permet à les machines virtuelles l'accès à Internet, via des adresses IP, qui est accessible par tous les utilisateurs d'internet.

➤ **API network**

On française " Réseau API", expose tous les APIs d'Openstack, les adresses IP sur ce réseau doit être accessible par tous les utilisateurs l'internet.

	Nodes		
	Controller	Network	compute
Nom	Cloud	Network	c01
Services	Mysql, RabbitMQ, Nova, Cinder, Glance, Keystone, Quantum	Quantum-L3 agent, Quantum DHCP-agent, Quantum Agent with Open vSwith	Nova-Compute, KVM, Nova-api, Quantum Agent with Open vSwith
Nombre minimum de disque	2	1	1
Adresse externe	10.25.0.30	10.0.0.9	--
Adresse internet	10.10.10.10	10.10.10.9	10.10.10.11
Nombre total de NIC	2	2	1

Tableau 3.1 : Architecture et informations de nœud [45]

### 3.3.1 L'installation de Controller Node:

- Les caractéristiques générales de machine controller est :
  - Système d'exploitation : Ubuntu server V12.04 64 bits.
  - CPU : 4 Vcpu.
  - Mémoire : 4096 Mo.
  - Charge mémoire : 60.47 Mo.
- Les Ressource de machine controller est :
  - Consommation CPU hôte : 24 MHz.
  - Consommation mémoire hôte : 1011.00 Mo.
  - Mémoire invité active : 61.00 Mo.
  - Stockage provisionné : 44.04 Go.
  - Stockage non-partagé: 44.04 Go.
  - Stockage utilize: 44.04 Go.

## 1. Opération système

- Préparation le système d'exploitation (Ubuntu Server 12.04) qui est installée sur le machine virtuelle. Pendant l'installation nous choisissons « **UTC** » pour configuration l'horloge, puis le nom de serveur est « **cloud** » et installer deux packages « **OpenSSH-Server, wget** ». après la fin installation Ubuntu il doit redémarrer la machine.
- Pour accès la machine on va saisie le nom utilisateur « **cloud** » et mot de passe. Et lancer la table de bord « **terminal** » et enter à la mode « **root** » et saisie le commande : # **apt-get install ubuntu-cloud-keyring**. Pour installe « Ubuntu Cloud Archive for Grizzly ».

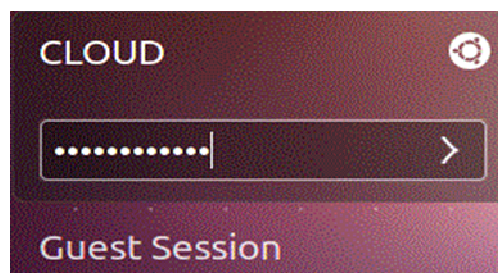


Figure 3.2: login de machine controller

- Après l'installation, nous téléchargeons les packages d'OpenStack Grizzly voir script **1.1 Page (70)**.
  - faire la mise à jour le système à laide : # **sudo apt-get update** et # **sudo apt-get upgrade**, redémarrer le système si nécessaire par #**sudo reboot**.
  - Après la démarrer le machine, faire la configuration d'internet voir script **1.2 Page (70)**.
  - Puis accès le fichier # **vi/etc/ sysctl.conf** et faire la vérification des normes voir script **1.3 Page (70)**.
  - Puis redémarrer le service de réseau par : # **service networking restart**.
  - Ajoute les différentes adresses des serveurs **Cloud, network et c01** voir script **1.4 Page (70)**.
  - Installation NTP par le commande # **apt-get install ntp** pour vérification l'horloge de serveur.
- ## 2. MySQL Database Service
- Installation de packages python-mysqldb et mysql-server par la commende suivent : #**apt-get install python-mysqldb mysql-server**

Cette commande demande le mot de passe « **root** » pour Mysql.

- Par défaut Mysql accepte la connexion de localhost.  
# **sed -i 's/127.0.0.1/0.0.0.0/g' /etc/mysql/my.cnf**, cette commande remplace l'adresse **127.0.0.1** par l'adresse **0.0.0.0** dans la file **etc/mysql/my.cnf**.
- Redémarrer le service de Mysql par la commande suivant : # **service mysql restart**.
- Puis accès à Mysql et faire la création des différentes bases de données des composent OponStack. **voir script 1.5 Page (70)**.

### 3. RabbitMQ Messaging Service

- Install the packages « **rabbitmq-server** » par la commande suivant:  
# **apt-get install rabbitmq-server**
- Changer le mot de passe par :  
# **rabbitmqctl change\_password guest password**

### 4. OpenStack Identity Service

- Install the packages **python-keystone python-keystoneclient** par la commande suivant:  
# **apt-get install keystone python-keystone python-keystoneclient**.
- Puis nous configurons keystone, après l'installation il. **voir script 1.6 Page (71)**.
- Puis redémarrer le service Keystone et lancement les tables de base de données. **voir script 1.7 Page (72)**.

```
mysql> show tables;
+-----+
| Tables_in_keystone |
+-----+
| credential          |
| domain              |
| ec2_credential       |
| endpoint             |
| group                |
| group_domain_metadata |
| group_project_metadata |
| migrate_version     |
| policy               |
| project              |
| role                  |
| service              |
| token                |
| trust                |
| trust_role           |
| user                  |
| user_domain_metadata |
| user_group_membership |
| user_project_metadata |
+-----+
```

**Figure 3.3:** Table de Keystone

- puis nous définissons les informations d'identification d'administrateur d'OpenStack qui utilisent pour l'interaction avec l'environnement OpenStack à partir le fichier **openrc**. **voir script 1.8 Page (71)**.



- Puis faire la création des Tentants, Roles, Users et Services via le fichier `keystone.sh openrc`. voir script 1.9 Page (72).

```

+-----+-----+-----+
|          id          |      name      | enabled |
+-----+-----+-----+
| 5daaa70bf2b44a4fb201fb112948b265 | admin         | True    |
| 567091f037004fe1a12846e53d520a9c | admin1        | True    |
| 8e8d0a1fe1ba4a3ca959b1237aa61f2a | demo          | True    |
| e63e258f24234c878004917dbb909ca0 | service       | True    |
+-----+-----+-----+

```

Figure 3.4:List de Tentant

```

+-----+-----+
|          id          |      name      |
+-----+-----+
| 12d70ac871bb4b949a3faa37d4f1d045 | Member        |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_      |
| f9349a2e63c34bb6a1be97fa0a1e24f4 | admin         |
+-----+-----+

```

Figure 3.5:List de Rôle

```

+-----+-----+-----+-----+
|          id          |      name      | enabled |      email      |
+-----+-----+-----+-----+
| 7ead2c9798c44c71b31dfae6359c4e9f | admin         | True    | admin@domain.com |
| 1febe89abbc4022802ab3e1a2daef0b  | cinder        | True    | cinder@domain.com |
| 14b05aff3dc24ac7bd4c1d59d5a638c3  | demo          | True    | demo@domain.com  |
| 253e902ae3734481ab7d77ae6366621b  | glance        | True    | glance@domain.com |
| 897eb5abd1204658847b4bb53afb2ad0  | nova          | True    | nova@domain.com  |
| 8da0257709e84c89a5683a91f7a05e19  | quantum       | True    | quantum@domain.com |
+-----+-----+-----+-----+

```

Figure 3.6:List de User

## 5. OpenStack Image Service

- Installation de packages Glance :

```
# apt-get install glance glance-api glance-registry python-glanceclient glance-common
```

- Puis faire la configuration de Glance voir script 1.10 Page (75).
- Puis redémarrer les services Glance voir script 1.11 Page (75).
- Créer des tableaux Glance dans Mysql:

```
# glance-manage db_sync
```

```
mysql> show tables;
+-----+
| Tables_in_glance |
+-----+
| image_locations  |
| image_members    |
| image_properties  |
| image_tags        |
| images            |
| migrate_version   |
+-----+
6 rows in set (0.00 sec)
```

Figure 3.7: Table de Glance

- Puis télécharger et importer Ubuntu 12.04 LTS image UEC et Cirros QCOW2 Image. voir script 1.12 Page (75).
- Vérifiez si les images Ubuntu 12.04 LTS image UEC et Cirros QCOW2 Image ont été téléchargé par la commandes **glance image-list**.

```
root@ubuntu:~# glance image-list
+-----+-----+-----+-----+-----+-----+
| ID | Name | Disk Format | Container Format | Size | Status |
+-----+-----+-----+-----+-----+-----+
| 353eba1c-2baa-4f3d-ac10-bc86257761df | Cirros | qcow2 | bare | 13147648 | active |
| a72a1a77-0a57-4116-be53-e0edd89d8824 | Ubuntu | qcow2 | bare | 251920384 | active |
+-----+-----+-----+-----+-----+-----+
```

Figure 3.8: Vérification d'image Glance

## 6. OpenStack Compute (Cloud Controller services)

- Installation de package Nova : **# apt-get install novnc**

Dans ce cas il existe un avertissement (warning) de fichier `/var/cache/apt/archives/nova-novncproxy_1%3a2013.1-0ubuntu2~cloud1_all.deb` et la solution avec la commande suivant :

```
#dpkg -i -- force-overwrite /var/cache/apt/archives/nova-novncproxy_1%3a2013.1-0ubuntu2~cloud1_all.deb
```

- Puis la configuration de Nova :
  - la configuration de service api-nova. voir script 1.13 Page (76).
  - Puis la configuration principale de Nova via le fichier `nova.conf` .voir script 1.14 Page (76).

- Puis charge des tableaux Nova dans Mysql par **# nova-manage db sync**

```
mysql> show tables;
+-----+
| Tables_in_nova |
+-----+
| agent_builds   |
| aggregate_hosts |
| aggregate_metadata |
| aggregates     |
| block_device_mapping |
| bw_usage_cache |
| cells         |
| certificates   |
| compute_node_stats |
| compute_nodes |
| console_pools |
| consoles      |
| dns_domains   |
| fixed_ips     |
| floating_ips  |
| instance_actions |
| instance_actions_events |
| instance_faults |
| instance_id_mappings |
+-----+
```

**Figure 3.9:**Table de Nova

- Puis redémarrer les services Nova. **voir script 1.15 Page (77).**

## 7. OpenStack Block Storage

Cinder est un service contient des différents utiles des stockages, nous avons utilisons le utile la plus communs et fondamentaux : LVM et iSCSI.

Ainsi utilisons un disque (/ dev / sdb) dans un groupe de volumes LVM appelé cinder-volumes, où volume logique est créé à partir de groupe de volumes.

- Installation de package Cinder :
 

```
# apt-get install cinder-api cinder-scheduler cinder-volume iscsitarget \ open-iscsi iscsitarget-dkms python-cinderclient linux-headers-`uname -r`
```
- Après l'installation, faire la configuration et démarrer les services iSCSI. **voir script 1.16 Page (77).**
- Puis la configuration de Cinder . **voir script 1.17 Page (77).**
- Puis la création du volume physique LVM et volume logique. **voir script 1.18 Page (77).**
- Puis charge des tableaux Cinder dans Mysql par **# cinder-manage db sync.**

```
mysql> show tables;
+-----+
| Tables_in_cinder |
+-----+
| backups           |
| iscsi_targetsets |
| migrate_version  |
| migrations        |
| quota_classes     |
| quota_usages      |
| quotas            |
| reservations      |
| services          |
| sm_backend_config |
| sm_flavors        |
| sm_volume         |
| snapshot_metadata |
| snapshots         |
| volume_glance_metadata |
| volume_metadata  |
| volume_type_extra_specs |
| volume_types     |
| volumes           |
+-----+
```

**Figure 3.10:** Table de Cinder

- Puis redémarrer les services Cinder. **1.19 Page (77)**.

## 8. OpenStack Network Service

- Installer le serveur Quantum par **# apt-get install quantum-server**.
- puis faire la configurer de service Quantum. **1.20 Page (78)**.
- Puis Activer le plugin OVS par:

```
# ln -s/etc/quantum/plugins/openvswitch/ovs_quantum_plugin.ini /etc/quantum/plugin.ini.
```

- Puis démarrer le service quantum par: **# service quantum-server restart**.

## 9. OpenStack Dashboard

Le service de Dashboard OpenStack fournit aux utilisateurs de l'environnement de Cloud Computing avec une interface graphique.

- Installation de package Horizon :

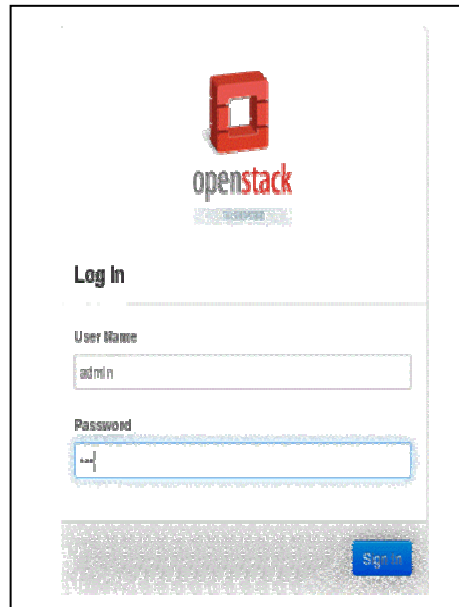
```
# apt-get install -y openstack-dashboard memcached python-memcache.
```

OpenStack Dashboard est maintenant disponible à <http://cloud/horizon>. Nous pouvons ouvrir une session avec **admin** pour User Name et Password est **key**.

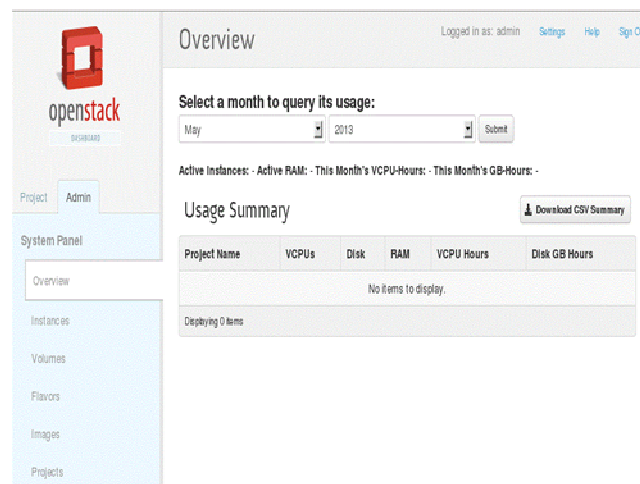
- puis nous définissons les informations d'identification d'administrateur d'OpenStack

qui utilisent pour l'interaction avec l'environnement OpenStack à partir le fichier **openrc**. voir script 1.8 Page (71).

- Puis faire la création des Projects, Roles, Users et Services via le fichier **keystone.sh** openrc. voir script 1.9 Page (72).



**Figure 3.11:** Login plate-forme OpenStack Horizon



**Figure 3.12:** Interface plate-forme OpenStack Horizon

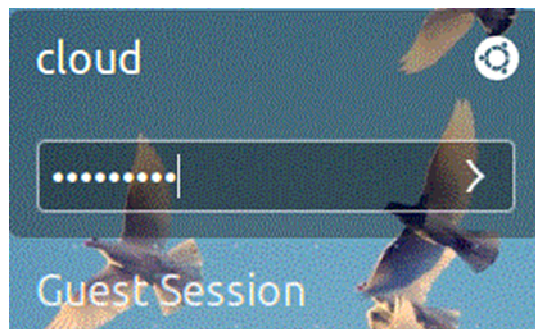
### I.1.1. L'installation de Network Node:

- Les caractéristiques générales de machine Network Node est :
  - Système d'exploitation : Ubuntu server V12.04 64 bits.

- CPU : 4 vCPU.
  - Mémoire : 4096 Mo.
  - Charge mémoire : 59.92 Mo.
- Les Ressource de machine controller est :
- Consommation CPU hôte : 23 MHz.
  - Consommation mémoire hôte : 705.00 Mo.
  - Mémoire invité active : 1556.00 Mo.
  - Stockage provisionné : 44.05 Go.
  - Stockage non-partagé: 26.04 Go.
  - Stockage utilise: 26.04 Go.

### 1. Opération système

- Préparation le système d'exploitation (Ubuntu Server 12.04) qui est installée sur le machine virtuelle. Pendant l'installation nous choisissons « **UTC** » pour configuration l'horloge, puis le nom de serveur est « **network** » et installer un packages « **OpenSSH-Server** ». après la fin installation Ubuntu il doit redémarrer la machine.
- Pour accès la machine on va saisie le nom utilisateur « **network** » et mot de passe. Et lancer la table de bord « **terminal** » et enter à la mode « **root** » et saisie le commande : # **apt-get install ubuntu-cloud-keyring**. Pour installe « Ubuntu Cloud Archive for Grizzly ».



**Figure 3.13:**login de machine network node

- Après l'installation, nous téléchargeons les packages d'OpenStack Grizzly voir script **1.1 Page (70)**.
- faire la mise à jour le système à laide : # **sudo apt-get update** et # **sudo apt-get upgrade**, redémarrer le système si nécessaire par #**sudo reboot**.

- Après la démarrage de la machine, faire la configuration d'internet **voir script 1.21 Page (78)**.
- Puis accéder au fichier **# vi/etc/ sysctl.conf** et faire la vérification des normes **voir script 1.3 Page (70)**.
- Puis redémarrer le service de réseau par : **# service networking restart**.
- Ajouter les différentes adresses des serveurs **Cloud, network et c01 voir script 1.4 Page (70)**.
- Installation NTP par la commande **# apt-get install ntp** pour vérifier l'horloge de serveur.

## 2. OpenStack Networking (Network Controller)

### A. Open vSwitch

- Installation de package:  
**# apt-get install quantum-plugin-openvswitch-agent\quantum-dhcp-agent quantum-l3-agent**
- Démarrer le service Open vSwitch par : **# service openvswitch-switch start**.
- Puis créer réseau bridge interne et externe, **voir script 1.22 Page (78)**.
- Puis faire la configuration de bridges **voir script 1.23 Page (79)**.
- Supprimer l'adresse IP de eth0 pour ajouter à br-ex . **voir script 1.26 Page (80)**.
- Redémarrer réseau par : **# service networking restart**.

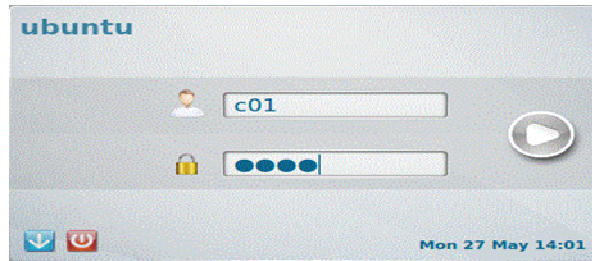
### B. Quantum

- Configurer de services Quantum **voir script 1.24 Page (79)**.
- Démarrer les services quantum **voir script 1.25 Page (80)**.

## I.1.2. Compute Node

### 1. Opération système

- Préparation le système d'exploitation (Ubuntu Server 12.04) qui est installée sur la machine virtuelle. Pendant l'installation nous choisissons « UTC » pour configuration l'horloge, puis le nom de serveur est « c01 » et installer deux packages « **OpenSSH-Server** ». après la fin installation Ubuntu il doit redémarrer la machine.
- Pour accéder à la machine on va saisir le nom utilisateur « c01 » et mot de passe. Et lancer la table de bord « **terminal** » et entrer à la mode « **root** » et saisir la commande : **# apt-get install ubuntu-cloud-keyring**. Pour installer « Ubuntu Cloud Archive for Grizzly ».



**Figure 3.14:** login de machine compute node

- Après l'installation, nous téléchargeons les packages d'OpenStack Grizzly voir **script 1.1 Page (70)**.
- faire la mise à jour le système à laide : **# sudo apt-get update** et **# sudo apt-get upgrade**, redémarrer le système si nécessaire par **#sudo reboot**.
- Après la démarrer le machine, faire la configuration d'internet voir **script 1.27 Page (80)**.
- Puis accès le fichier **# vi/etc/ sysctl.conf** et faire la vérification des normes voir **script 1.3 Page (70)**
- Puis redémarrer le service de réseau par : **# service networking restart**.
- Ajoute les différentes adresses des serveurs **Cloud, network et c01** voir **script 1.4 Page (70)**.
- Puis l'installation NTP par le commande **# apt-get install ntp** pour vérification l'horloge de serveur.

## 2. OpenStack Compute (Compute Node services)

Dans ce nœud nous installons nova compute service, qu'elle fournit à ce nœud la capacité d'héberge les machines virtuelles.

- Installation de package Nova Compute :

**# apt-get install nova-compute-kvm**

Nova-compute-kvm est un paquet fournit la virtualisation de logicielle, il nécessite un processeur support la virtualisation tel que Intel<sup>®</sup> Xeon(R).

- Configurer de service Nova. voir **script 1.28 Page (81)**.
- Puis redémarrer les services Nova par **# service nova-compute restart**.

## 3. OpenStack Networking (Compute Node)

### A. Open vSwitch

- Installation de package OVS par **# apt-get install openvswitch-switch**
- Démarrer de service Open vSwitch par **# service openvswitch-switch start**.



- Créer un bridge interne par `# ovs-vsctl add-br br-int`

## B. Quantum

- Installation des packages par `# apt-get install -y quantum-plugin-openvswitch-agent`
- Puis faire la configuration de quantum. **voir script 1.29 Page (82).**
- Puis démarrer d'Agent par `# service quantum-plugin-openvswitch-agent restart`.
- Fin l'installation.

### 3.4 Installation plateforme Eucalyptus

- Nous installons Eucalyptus sur une seule machine, où nous installons CLC, CC, SC, Walrus et NC sur une machine sa caractéristique sont: CPU Intel (R) 64bit, disque dure 40 GB et deux cartes réseaux.
- Nous préparons une machine avec le système d'exploitation Ubuntu Server 12.04 64 bit.
- Puis faire l'installation et la configuration de bridge. **voir script 2.1 Page (82).**
- puis faire l'installation et démarrer de server NTP. **voir script 2.2 Page (83).**
- la première étape est le téléchargement des packages par une inscription dans la plateforme Eucalyptus, via le site : <https://communitycloud.eucalyptus.com>, puis l'installation ces packages, après leur transformation de forme **rpm** à **deb**.
- la commande `sudo alien "nom de paquet" .rpm` pour le changement de format à deb.
- la commande `dpkg -i "nom de paquet" .deb` pour installer les paquets.
- puis faire la mise à jour de système par `apt-get update`.
- Puis installer les packages CLC, CC, SC, NC et Walrus. **voir script 2.3 Page (83).**
- Puis redémarrer les services d'eucalyptus. **voir script 2.4 Page (83).**
- Le problème de la limite de ressources gêné la suite de l'installation pour obtenir l'interface finale.

### 3.5 Conclusion

Dans ce chapitre nous installons deux plate-formes de cloud computing, et pendant l'installation, nous avons rencontré quelques difficultés, pour cela nous allons faire une comparaison générale entre eux dans le chapitre prochaine, à but, choisir le plate-forme la meilleure et plus facile dans l'installation.

# *Chapitre 4 : Un Framework Comparatif entre OpenStack et Eucalyptus*

## 4.1 Introduction

Dans le chapitre précédant nous avons expliqué les différentes étapes d'installation pour les deux plateformes OpenStack et Eucalyptus.

Dans ce chapitre nous essayerons de construire un framework comparatif sur deux volets , un volet basé sur des caractéristiques et standards générales inspirer du littérature et un deuxième volet technique basé sur des caractéristiques techniques inspirer du l'installation et la mise en place de deux plates-formes OpenStack et Eucalyptus..

## 4.2 Étude comparative entre deux plates-formes

Nous avons réalisé une étude comparative entre les deux plate-formes en deux façons générale et les ressources et le coût pour mise en place.

### 4.2.1 Comparaison générale

Nous comparons entre les deux, à partir les différents critères de classification :

- **Produit par** : la société qui a produit cette plate-forme.
- **But** : l'objectif attendu de sa création.
- **Le domaine d'utilisation** : le Cloud Computing est présent dans plusieurs domaines d'utilisation. Donc pour chaque plate-forme il y a un ou plusieurs domaines d'utilisation, qui spécifient les tendances de ce dernier.
- **Langage de programmation** : il représente le langage de programmation, que ces plate-formes sont créés par elle.
- **Tolérance aux pannes** : c'est la capacité d'un système à fonctionner malgré une défaillance d'un de ses composants.
- **Migration des machines virtuelles** : est le déplacement des machines virtuelles d'un serveur physique à un autre. On distingue deux types de migration: la migration à chaud qui se fait lors de l'exécution des machines virtuelles et la migration à froid ne peut être réalisée sauf si la machine virtuelle est arrêtée.
- **Compatibilité avec Amazon EC2** : Amazon EC2 est la solution d'infrastructure du Cloud Computing la plus connue et la plus utilisée.

Le tableau suivant résume cette comparaison :

	<b>Eucalyptus</b>	<b>OpenStack</b>
<b>Produit par</b>	l'université Santa Barbara de l'université de Californie	Rackspace, NASA, Dell, Citrix, Cisco, Canonical et plus que 50 autres organisations
<b>But</b>	Mettre en œuvre un Cloud Computing. Aspect Commercial.	Mettre en œuvre un Cloud Computing. Gérer l'infrastructure de Cloud en manière plus efficace.
<b>Domaine d'utilisation</b>	les entreprises qui souhaitent construire leur propre Cloud.	Les sociétés, les fournisseurs de services, les chercheurs et les centres de données mondiaux.
<b>Langage de programmation</b>	Java, C, et Python	Python
<b>Tolérance aux pannes</b>	Séparation des clusters Controllers	Réplication de données
<b>Migration à froid des VMs :</b>	Oui	Oui
<b>Migration à chaud des VMs</b>	No	No
<b>Compatibilité avec Amazon Web Service</b>	Oui	Oui

**Tableau 4.1:**Framwork01 : caractéristiques générales

Nous expliquons les critères **Tolérance aux pannes** de Tableau 2 :

**La tolérance aux pannes** : plusieurs méthodes pour éviter les risques de pertes des données

- **La séparation des clusters Controller** pour Eucalyptus réduit la chance de défaillance. parce qu'il ce composant gère plusieurs Nodes Controllers et il est responsable de la gestion du cycle de vie des instances virtuelles.
- **le principe de réplication** dans OpenStack est lorsqu'une machine tombe en panne, une des copies de cette machine prend sa place dans le système. Cependant, cette approche nécessite la disponibilité d'un nombre important de ressources.

#### 4.2.2 Comparaison pendant l'installation

- **Les besoins Matériels en réalité:** pour l'installation d'une plate-forme, il doit fournir l'espace sur la machine et des matériels spéciaux
- **Systèmes d'exploitation supportés** : pour construire un Cloud Computing via ces plate-formes, on doit disposer d'un système d'exploitation qui permet leur installation et la gérance des ressources physiques.
- **L'hyperviseur:** est un logiciel de virtualisation.
- **L'architecture** : c'est la structure du système qui comprend les ressources matérielles et logicielles et l'interaction entre eux.
- **Les besoins des serveurs:** pendant l'installation les premières préparations.
- **Les serveurs:** les serveurs principaux dans l'architecture d'une plate-forme.
- **Téléchargement des packages sources** : la manière de téléchargement des packages de la plate-forme. le paquet est une archive comprenant les fichiers informatiques, les informations et procédures nécessaires à l'installation d'un logiciel sur un système d'exploitation, il y a plusieurs formes des paquets comme **deb, rpm et tgz**.
- **L'installation des packages:** pour chaque plate-forme il y a des étapes très importantes pour faire la mise en place sur la machine. Ici, nous décrivons la manière d'installation de plate-forme.
- **Temps d'installation de paquet spéciaux** : le temps consacré dans l'installation de paquet.
- **le composant responsable:** est le chef de travail en sien de plate-forme.
- **Service Stockage** : gérer des grands volumes de données.

- **Service Réseau** : la responsable de réseau dans le Cloud Computing et réseau virtuelle.
- **Service Hébergement de VMs**: qui gérée et contrôler le cycle de vie des instances.
- **Services de Stock de Mode Bloc**: permettant de stockage aux machines virtuelles.
- **Base de données** : pour gère l'authentification et sauvegarde les données.
- **Droit d'accès** : qui responsable de vérification d'accès à la plate-forme.
- **Configuration de Réseau** : la connexion dans plate-forme.
- **Server NTP**: Network Time Protocol est un protocole permettant de distribuer l'heure sur un réseau informatique.
- **L'interface utilisateur** : il représente l'interface de communication entre l'utilisateur et le Cloud.
- **Sécurité** : le Cloud Computing offre plusieurs des services et des ressources, il est donc essentiel de contrôler l'accès des utilisateurs et des administrateurs aux ces services.
- **Start de plate-forme**: le démarrage les composants de plate-forme.
- **Orientation**: le type de Cloud, qu'elle est crée.
- **Les risque d'installation**: les problèmes; les étapes qui' ils n'installent pas.
- **L'existant de plate-forme**: autre façon d'installation de plate-forme.

	<b>Eucalyptus</b>	<b>OpenStack</b>
<b>Les besoins Matériels en réalité</b>	<p>Processeur: Intel® 64bit.</p> <p>RAM pour chaque serveur : maximum 10 GB.</p> <p>Disque dure pour deux serveurs : 40 GB.</p> <p>Espace de stockage: 250 GB.</p> <p>Carate réseau pour chaque serveur : 2 carates réseau 1 GB.</p> <p>Deux Machines physiques.</p>	<p>Processeur: Intel® 64bit.</p> <p>RAM pour chaque serveur : maximum 35GB.</p> <p>Disque dure pour deux serveurs : minimum 40 GB.</p> <p>Deux Disque dure pour le serveur principal: 2 Tira B.</p> <p>Espace de stockage: 250 GB.</p> <p>Carate réseau pour chaque serveur : 2 carates réseau 1 GB.</p> <p>Trois Machines physiques.</p>
<b>Systèmes d'exploitation supportés</b>	Linux (Ubuntu, Red Hat Enterprise Linux, CentOS, OpenSUSE et Debian).	Linux (Ubuntu et Debian). Windows Server 2008.
<b>L'architecture</b>	<p>Hiérarchique, Cinq composants (CLC;CC;SC;NC et Walurs)</p> <p>Supporte multiple cluster.</p> <p>Minimum deux serveurs.</p>	<p>Hiérarchique, sept composants (Nova;Swift;Galance;Quantum; Kystone, Rabbit MQet Cinder).</p> <p>Minimum Trois serveurs chaque serveur contient des composants spéciaux.</p>
<b>L'hyperviseur</b>	KVM et Xen.	KVM et Xen.
<b>Les besoins des serveurs</b>	Server SSH et Time Zone UTC pour deux serveurs.	<p>Serveur principal (serveur 1) nommé Cloud, nécessite Server SSH, Time Zone UTC et le package wget.</p> <p>Serveur 2 nommé Network, nécessite Server SSH et Time Zone UTC.</p>

		Serveur 3 nommé C01, nécessite Server SSH et Time Zone UTC.
<b>Les serveurs</b>	Serveur 1 contient (CLC;CC;SC et Walrus). Serveur 2 contient NC.	Cloud contient MySQL; RabbitMQ; Keystone, Glance, Nova, Cinder, Quantum et Horizon. Network contient Open-vSwitch et Quantum L3 Agent. C01 contient KVM, Nova-Compute et Open-vSwitch.
<b>Téléchargement des packages sources</b>	Par une inscription dans la plateforme eucalyptus, et faire un téléchargement simple.	Par une ligne de archive.
<b>L'installation des packages</b>	Très difficile parce que ces packages en forme rpm, ils nécessitent la convertir en forme deb, et après faire l'installation à partir des commandes	à partir des commandes <b>apt-get updates</b> et <b>apt-get upgrade</b>
<b>Temps d'installation des paquets spéciaux</b>	Est liée à rapidité de réseau. L'installation des paquets (CLC, CC....etc.) via des commandes très simple.	Est liée à rapidité de réseau. L'installation des paquets (keystone, nova ....etc.) via des commandes moins facile.
<b>le composant responsable</b>	Cloud Contrôle <b>CLC</b>	OpenStack Compute <b>NOVA</b>
<b>Service Stockage</b>	<b>Walrus</b> qui permet de stocker les données persistantes. Très bien, <b>Walrus</b>	<b>Swift</b> gérer une large capacité de stockage. Bien parce que l'espace de stockage est partagée.



	contrôle chaque nœud.	
<b>Service Réseau</b>	<b>Contrôleur de cluster</b> gérer la mise en place du réseau entre les instances des différents nœuds. N'est pas large.	<b>Rabbit MQ</b> gérer la communication dans OpanStack. <b>Nova-compute</b> gérer la mise en place du réseau entre les instances des différents.
<b>Service Hébergement de VMs</b>	<b>Contrôleur de nœud (NC)</b> est chargé de vérifier le cycle de vie des instances en cours d'exécution sur le nœud.	<b>Nova-compute</b> est la responsable de la création et manipulation des machines virtuelles.
<b>Services de Stock de Mode Bloc</b>	<b>Contrôleur de stockage (SC)</b> fournir le système de stockage de blocs aux instances.	<b>Cinder</b> gère l'accède aux volumes de stockage de blocs pour les instances.
<b>Base de données</b>	<b>N'exit pas.</b>	<b>Mysql, PostgreSQL,</b> pour sauvegarde les entrés.
<b>Droit d'accès</b>	<b>Bien</b> , chaque contrôleur est authentifié par <b>clé SSH</b> et fichiers de permission pour authentifier toutes les transactions. Les paquets est authentifié par <b>clé publique</b> , est téléchargée pas l'utilisateur de plate-forme.	<b>Très bien</b> , authentification d'utilisateur par Token de keystone. Utilisateur montre son l'accès via Token à tous le cycle de session.
<b>Configuration de Réseau</b>	<b>Très difficile.</b> Configure par server DHCP.	<b>Simple</b> , nécessite des adresses fixes.

<b>Serveur NTP</b>	<b>Existe.</b>	<b>Existe.</b>
<b>L'interface utilisateur</b>	Une interaction facile.	Une interaction facile, fourni par <b>Horizon.</b> L'accès à cette interface via l'adresse suivant <a href="http://10.0.0.10/horizon">http://10.0.0.10/horizon</a>
<b>Start de plateforme</b>	<b>Très facile</b> par des commandes service "nom package" start.	<b>Très facile</b> par des commandes service "nom package" restart.
<b>Orientation</b>	Cloud public et privé	Cloud public et privé
<b>Les risques d'installation</b>	Téléchargement et l'installation des packages. Configuration réseau, l'ajoute bridge et la limite les ressources, parce qu'il nécessite des machines physiques. Généralement <b>moyen.</b>	L'installation de keystone. Le bridge ne connecte pas, et la limite les ressources. Généralement <b>simple.</b>
<b>L'existant de plateforme</b>	Ubuntu Enterprise Cloud UEC.	<b>Sans opinion.</b>

Tableau 3.2 : Framwork02 : caractéristiques techniques

### 4.3 Conclusion

Dans ce chapitre nous comparons entre les deux solutions de Cloud et Nous avons vu les aspects similaires et différences entre Eucalyptus et OpenStack en façon générale et technique, et à partir de cette comparaison nous concluons que :

Eucalyptus est une solution mature et qui permet l'installation d'une infrastructure de type cloud assez facilement tant que l'on reste dans les clous de son fonctionnement. Cependant, le but de cloud computing de faciliter l'évolution ce qu'Eucalyptus ne permet pas. C'est donc pour cela qu'il est actuellement délaissé pour d'autres solutions.

OpenStack est une solution encore jeune, mais qui a un potentiel très important par rapport à son architecture et sa communauté ainsi que le support de ses partenaires. Il s'agit donc d'une solution à surveiller car nous pensons qu'elle peut devenir la référence des solutions de cloud computing libre.

## Conclusion générale

Ce mémoire aborde le Cloud Computing comme un modèle d'informatique dématérialisée permettant l'accès via un réseau et sur demande à un ensemble partagé de ressources configurables : réseaux, serveurs, stockage, applications, etc., réparties "en nuages" à divers endroits.

Un des avantages clés de Cloud Computing vient du fait qu'ils peuvent être utilisés par des entreprises et des individus qui désirent externaliser certains de leurs besoins informatiques vers un fournisseur de nuage. Ce fournisseur peut ensuite fournir des services payés à l'utilisation à ses clients. D'un point de vue financier, cette relation est intéressante parce qu'elle permet à toutes les parties de réaliser des économies d'échelle. En effet, un fournisseur peut acquérir des ressources physiques en grande quantité pour servir ses nombreux clients.

L'objectif de notre travail de fin d'étude, effectuée dans le cadre de ce master aborde principalement le monitoring des applications de la couche IaaS du Cloud Computing.

Nous avons donné une idée générale sur le Cloud Computing, son architecture, ses avantages, ses caractéristiques principales et ses différents services. Nous avons fait par la suite un état de l'art pour les différentes solutions propriétaires et open source et du Cloud particulièrement : Microsoft Windows Azure, vCloud de VMware, Eucalyptus et OpenStack. Après l'installation et la mise en place de deux plateformes Eucalyptus et OpenStack nous récapitulons une comparaison intéressante entre les deux on le peut résumer comme suit :

Eucalyptus est une solution mature et qui permet l'installation d'une infrastructure de type cloud assez facilement tant que l'on reste dans les limites de son fonctionnement. Cependant, le but de cloud computing est de faciliter l'évolution ce qu'Eucalyptus ne permet pas. C'est donc pour cela qu'il est actuellement délaissé pour d'autres solutions.

OpenStack est une solution encore jeune, mais qui a un potentiel très important par rapport à son architecture et sa communauté ainsi que le support de ses partenaires. Il s'agit donc d'une solution à surveiller car nous pensons qu'elle peut devenir la référence des solutions de cloud computing libre.

Notre perspectives est de etendu cette etude analytique et comparatif à des autre solutions de cloud computing que se soit libre ou commercial on prenant en considération d'autre critères d'evaluation et particulièrement les solutions qui déjà déployer sur le marché algerienne.

## Bibliographie

- [1]. **Marc Jalabert** .*LE CLOUD COMPUTING Du concept à la réalité* .ÉDITO.
- [2]. *Cloud Computing en Afrique Situation et perspectives*.UIT. 2012.
- [3]. **M. Lucas Nussbaum** .*Cloud Computing* .IUT Nancy Charlemagne .2009/2010.
- [4]. *CLUSIF-Cloud-2010-Securite* .CLUSIF .14/04/2010.
- [5]. *Cloud Computing* .ADIRA .2012.
- [6]. **Jules-Henri Gavetti** .*Le Cloud en toute confiance Ikoula l'architecte de vos projets Cloud* .iKoula.
- [7]. **Vic (J.R.) Winkler** .*La sécurité dans le Cloud* .Pearson Education France. 2011.
- [8]. **Maurice Audin** .*Etat de l'art du Cloud Computing et adaptation au Logiciel Libre* .bearstech .2009.
- [9]. *Le Cloud Computing : Réelle révolution ou simple évolution ?*. Bureau d'expertise technologique **Wygwam**™ .
- [10]. **Abdelwahed DIDI, Gilles SCALA, Michael DIOT**. *Les Solutions de Stockage en Réseau Le SAN Contre Le NAS*. 10.02.2003.
- [11]. **Michel Ricard** .*Stockage sur Disque DAS, NAS, SAN Les Concepts de Base*. 12 avril 2006.
- [12]. **Darryl Humphrey, Harley Young**. *Élaborez le nuage de votre entreprise*.
- [13]. **Frédéric Desprez**.*les clouds, la vraie science*. inria. 2010.
- [14]. **Nicolas HENNION**. *Introduction aux technologies cloud aux technologies cloud*. Janvier 2011.
- [15]. **Hmida DJEDIA**. *Cloud Computing: est-il réelle révolution ou simple évolution*.
- [16]. **Yohan PARENT, Maxime LEMAUX, Cyprien FORTINA, Hyacinthe CARTIAUX**. *Cloud Computing*. Université Nancy 2 IUT Nancy-Charlemagne. Année universitaire 2010-2011.
- [17]. **Meriam MAHJOUB**. *Mémoire étude et expérimentations du Cloud Computing pour le monitoring des applications orientées services* .L'École Nationale d'Ingénieurs de Sfax.
- [18]. **Raphaël Ferreira**. *OpenStack Presentation*.
- [19]. **Thierry Carrez**. *OpenStack Le cloud libre*.
- [20]. **Julie Desoeuvres**. *LA PLATEFORME WINDOWS AZURE*.

- [21]. *Présentation de Windows Azure L'OFFRE PLATEFORM-AS-A-SERVICE DE MICROSOFT*. Microsoft Corporation 2010.
- [22]. **Nicolas CLERC**. *Windows Azure – Présentation technologique*.
- [23]. *Plate-forme applicative de Cloud VMware vFabric*. Copyright © 2011 VMware, Inc.
- [24]. **Quartier Boieldieu, Hillview Ave**. *Composants de base VMware vSphere*. Copyright © 2009–2011 VMware, Inc.
- [25]. **Quartier Boieldieu, Hillview Ave**. *Présentation de VMware vSphere*. Copyright © 2009, 2010 VMware, Inc.
- [26]. **Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, et Matei Zaharia**. *A view of cloud computing*. Séminaire ACM, New York, Avril 2010.
- [27]. **A.Verma, G.Kumar et R.Koller**. *The cost of reconfiguration in a cloud*. proceedings de la 11ème International Middleware Conférence voie industrielle. Bangalore. Novembre 2010.
- [28]. **P. Sempolinski et D. Thain**. *A Comparison and Critique of Eucalyptus, OpenNebula and Nimbus*. la deuxième Conférence internationale IEEE sur le Cloud Computing Technologie et Science (CloudCom). Indianapolis, novembre 2010.
- [29] **T. Cordeiro, D. Damalio, N. Pereira, P. Endo, A. Palhares, G. Gonçalves, D. Sadok, J. Kelner, B. Melander, V. Souza et J. Mångs**. *Open Source Cloud Computing Platforms*. la 9ème Conférence internationale sur la grille et coopérative informatique (GCC).Nanjing, novembre 2010

- **Web graphiques**

- [30]. URL : [http://www.memoireonline.com/01/13/6713/m\\_Etude-et-mise-en-place-dune-solution-cloud-computing--privee-dans-une-entreprise-moderne-cas7.html](http://www.memoireonline.com/01/13/6713/m_Etude-et-mise-en-place-dune-solution-cloud-computing--privee-dans-une-entreprise-moderne-cas7.html), Consulté de 28.02.2013.
- [31]. URL : <http://www.cloudconsulting.fr/index.php/fr/le-cloud-computing/fournisseurs-cloud> , Consulté de 06.03.2013.
- [32]. URL : <http://www.dataligence.com/site/dossier-special-stockagedonnees/stockage-et-disques-durs/les-nas-et-les-san.html>, Consulté de 20.03.2013.

- [33]. URL : <http://communication.sysdis.fr/2012/05/09/cloud-computing-dfinitions-et-notions-de-base/>, Consulté le 01.04.2013.
- [34]. URL : [http://fr.wikipedia.org/wiki/Cloud\\_computing](http://fr.wikipedia.org/wiki/Cloud_computing), Consulté de 01.02.2013.
- [35]. URL : <http://www.haute-disponibilite.net/2008/07/15/difference-entre-grid-et-cloud-computing/> , Consulté de 04.04.2013.
- [36]. URL : [http://en.wikipedia.org/wiki/Network\\_as\\_a\\_service](http://en.wikipedia.org/wiki/Network_as_a_service), Consulté de 06.04.2013.
- [37]. URL : <http://cloudcomputing.sys-con.com/node/1203562>, Consulté de 06.04.2013.
- [38]. URL : <http://philippe.developpez.com/articles/azure-introduction/>, Consulté de 2013.04.13.
- [39]. URL : <http://msdn.microsoft.com/fr-fr/library/windowsazure/gg278356.aspx>, Consulté de 2013.04.14.
- [40]. URL : <http://www.vmware.com/fr/next-decade/cloud-application-platform.html>, Consulté de 2013.04.16.
- [41]. URL : [http://info.vmware.com/content/EMEA\\_17873\\_Transform\\_vfabric\\_FR](http://info.vmware.com/content/EMEA_17873_Transform_vfabric_FR)], Consulté de 2013.04.18.
- [42]. URL : <http://www.google.dz/imgreshttp://cloud-oss-fr.blogspot.com> ,de 2013.04.24.
- [43]. <http://cdn-static.zdnet.com/i/r/story/70/00/016753/fb-datacenter-620x353.jpg>, Consulté de 2013.04.10.
- [44].URL: <http://fr.slideshare.net/search/slideshow?searchfrom=header&q=cloud+computing>, Consulté de 2013.04.10.
- [45]. URL: <http://docs.openstack.org>, Consulté de 2013.05.02.
- [46]. URL: <http://www.lemondeinformatique.fr>, Consulté de 2013.05.20.
- [47].URL: <http://gigaom.com/2009/11/06/10-top-open-source-resources-for-cloud-computing/> , Consulter le 19/06/2013.
- [48]. URL:<http://socializedsoftware.com/2010/01/20/eleven-open-source-cloud-computing-projects-to-watch/>, Consulter le 19/06/2013.



## Glossaire

Nous avons définie dans le glossaire les termes qui est utilise en mémoire plus rapide de ceux-ci :

- **ADSL (Asymmetric Digital Subscriber Line):** Service d'accès à l'Internet utilisant les lignes téléphoniques classiques, en utilisant une bande de fréquence plus élevé que celles utilisée pour la téléphonie.
- **TIC (Information and Communication Technologies):** regroupent les techniques utilisées dans le traitement et la transmission des informations, principalement de l'informatique, de l'Internet et des télécommunications.
- **MIT (Institut de Technologie du Massachusetts):** est une institution de recherche et une université américaine, spécialisée dans les domaines de la science et de la technologie. Située à Cambridge, dans le Massachusetts, à proximité immédiate de Boston, au nord-est des États-Unis.
- **AFS (Andrew File System) :** est un système d'archivage distribué.
- **VPN (Réseaux Privés Virtuels):** est un réseau de données privé exploitant un réseau public tel qu'Internet, mais invisible des autres utilisateurs, qui chiffre les données au niveau d'un nœud et utilise des procédures de sécurité pour créer un « tunnel » à travers lequel les données peuvent être transmises.
- **REST (Representational State Transfer) :** est une manière de construire une application pour les systèmes distribués comme le World Wide Web. Le terme a été inventé par Roy Fielding en 2000. REST n'est pas un protocole ou un format, c'est un style d'architecture, c'est le style architectural original du Web.
- **SOAP (Simple Object Access Protocol) :** est un protocole de RPC orienté objet bâti sur XML. Il permet la transmission de messages entre objets distants, ce qui veut dire qu'il autorise un objet à invoquer des méthodes d'objets physiquement situés sur un autre serveur. Le transfert se fait le plus souvent à l'aide du protocole HTTP, mais peut également se faire par un autre protocole, comme SMTP.
- **Simple Storage Service :** est un service de stockage via service web d'Amazon. Amazon S3 est un service de stockage pour Internet. Il est pensé pour simplifier l'accès à l'informatique à l'échelle du web, pour les développeurs. Amazon S3 offre une simple interface de services web à utiliser pour stocker et récupérer n'importe

---

quelle quantité de données, à tout moment, de n'importe où sur le web. Il permet aux développeurs d'accéder à la même infrastructure hautement évolutive, fiable, sûre, rapide, et peu coûteuse qu'Amazon utilise pour faire fonctionner son propre réseau mondial de sites. Le service vise à maximiser les avantages d'échelle et les transmettre aux développeurs.

- **Le stockage extensible (Amazon EBS) :** fournit des volumes de stockage en mode bloc à utiliser avec les instances AmazonEC2. Les volumes EBS Amazon sont des stockages hors instances qui persistent indépendamment de la vie d'une instance. Amazon Elastic Block Store fournit des volumes de stockage hautement disponible et fiable qui peuvent être annexés à une instance Amazon EC2 en cours d'exécution et exposés comme un périphérique au sein de l'instance. EBS Amazon est spécialement adapté aux applications qui nécessitent une base de données, un système de fichier ou un accès à un stockage brut en mode bloc.
- **La licence BSD :** est une licence libre utilisée pour la distribution de logiciels [.http://www.freebsd.org/copyright/license.html/](http://www.freebsd.org/copyright/license.html)
- **KVM (Kernel-based Virtual Machine) :** est une solution de virtualisation libre pour Linux. Elle fonctionne sur les architectures x86 disposant des technologies Intel VT ou AMD SVM (AMD-V).
- **Xen :** est un logiciel libre de virtualisation, plus précisément un hyperviseur de machine virtuelle. Son développement a débuté sous la forme d'un projet de recherche de l'université de Cambridge au Royaume-Uni. La société XenSource a par la suite été créée et en a poursuivi le développement. Xen permet de faire fonctionner plusieurs systèmes d'exploitation virtuels (invités) sur une seule machine hôte.
- **Grid 5000 :** est une infrastructure distribuée sur 9 sites en France, pour la recherche dans les systèmes parallélisés à grande échelle et les systèmes distribués.
- **NASA :** est l'agence gouvernementale qui a en charge la majeure partie du programme spatial civil des États-Unis.
- **Rackspace Cloud :** est un fournisseur de plate-forme de Cloud Computing.
- **ASP.NET :** est un ensemble de technologies de programmation web créé par Microsoft. il permet de créer des sites webs dynamiques, des applications web ou des web services XML.

- **Internet Information Services**, communément appelé **IIS** : est le logiciel de serveur services Web (ou FTP, SMTP, HTTP etc.) de la plateforme Windows NT.
- **Microsoft Visual Studio** : est une suite de logiciels de développement pour Windows conçue par Microsoft. La dernière version s'appelle Visual Studio 2012.

Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications Web ASP.NET, des Services WebXML, des applications bureautiques et des applications mobiles.

- **X86**: est un nom d'une famille de processeurs de chez Intel. Il y eut les 8086, 186, 286, 386, 486, le Pentium.
- **AMQP** : est un protocole ouvert pour les systèmes de messagerie orientés intergiciel développé par la banque JPMorgan Chase.  
L'objectif d'AMQP basée sur les principes suivants : orienté message, utilisation de files d'attente, routage (point a point et par diffusion), fiabilité et sécurité.
- **Le Temps Universel Coordonné (UTC)** : est la référence de temps actuelle. Contrairement au Temps Universel (TU), il n'est pas lié à la rotation de la Terre. Il est donc stable (car la vitesse de rotation de la Terre varie).
- **Apt-get** : est le programme en ligne de commande pour la gestion des paquets. Il peut être considéré comme l'outil de base pour les autres programmes de la bibliothèque APT.
- **Update** : est un commande permet de resynchroniser un fichier d'index répertoriant les paquets disponibles et sa source. Ces fichiers sont récupérés aux endroits spécifiés dans /etc/apt/sources.list. Ainsi, lorsqu'on utilise une archive Debian, cette commande récupère les fichiers Packages.gz et les analyse de manière à rendre disponibles les informations concernant les nouveaux paquets et les paquets mis à jour.
- **Upgrade** : est un commande permet d'installer les versions les plus récentes de tous les paquets présents sur le système en utilisant les sources énumérées dans /etc/apt/sources.list. Les paquets installés dont il existe de nouvelles versions sont récupérés et mis à niveau. En aucun cas des paquets déjà installés ne sont supprimés ; de même, des paquets qui ne sont pas déjà installés ne sont ni récupérés ni installés. Les paquets dont de nouvelles versions ne peuvent pas être installées sans changer le statut d'installation d'un autre paquet sont laissés dans leur version courante.

- **Install** : est une commande suivie par un ou plusieurs paquets à installer. Chaque paquet est un nom de paquet ; ce n'est pas un nom complet de fichier (par exemple, dans un système Debian GNU/Linux, on indiquera libc6 et non pas libc6\_1.9.6-2.deb). Tous les paquets requis par le(s) paquet(s) que l'on veut installer sont aussi récupérés et installés. Le fichier /etc/apt/sources.list est utilisé pour retrouver les paquets désirés.
- **Open vSwitch** : est une implémentation logicielle d'un switch ethernet. Concrètement il est constitué d'un service (ovs-vswitchd) et d'un module kernel (openvswitch\_mod). Le service permet de commuter effectivement les paquets vers les bons ports virtuels, alors que le module kernel permet de capturer le trafic provenant des interfaces réseau, et d'y ré-injecter le trafic légitime.
- **Deb** : est le format de fichier des paquetages logiciels de la distribution GNU/Linux Debian. Presque toutes les distributions basées sur Debian utilisent aussi ce format.
- **RPM Package Manager** : est un système de gestion de paquets de logiciels utilisé sur certaines distributions GNU/Linux. Le système est composé d'un format ouvert et d'un logiciel libre de manipulation des fichiers de ce format. C'est le format utilisé par Linux Standard Base (LSB).
- **Tgz** : est un fichier archive créé avec **tar** et ensuite compressé généralement avec **gzip**.
- Le programme **tar** (de l'anglais **tape archiver**, littéralement « archiveur pour bande ») : est un logiciel d'archivage de fichiers standard des systèmes de type.
- **gzip** (acronyme de **GNU zip**) : est un logiciel libre de compression.
- **Wget** : est un programme en ligne permet le simple téléchargement d'un fichier depuis le Web. Il supporte les protocoles HTTP, HTTPS et FTP ainsi que le téléchargement au travers des proxies HTTP.

---

## Abréviations

### A

**ADSL:** Asymmetric Digital Subscriber Line.

**API:** Application Programming Interfaces.

**AMQP:** Advanced Message Queue Protocol.

**ASP.NET:** Active Server Page.

**AFS:** Andrew File System.

**ASP:** Application Service Provider.

### C

**CLI:** Command Line Interface.

**CaaS:** Communication as a Service.

**CPU :** Central Processing Unit.

**CIFS :** Common Internet File System.

**CRM:** Customer Relationship Management.

### D

**DHCP:** Dynamic Host Configuration Protocol

**DAS:** Direct Attached System.

### E

**Eucalyptus:** Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems.

**ESXi:**

**EBS:** Elastic Block Storage.

**EC2:** Elastic Compute Cloud.

### F

**FTP:** File Transfer Protocol.

**FC-AL:** Fibre Channel.

**FAH :** Fournisseur Applications Hébergées.

### H

**HTTPS:** Hypertext Transfer Protocol Secure.

**HTML:** Hyper Text.

**HA:** High Availability.

### I

**IIS:** Internet Information Services

**IT:** Information Technique.

**IAAS:** Infrastructure as a Service.

**IP:** Internet Protocol.

**IBM:** International Business Machines.

### K

**KVM:** Kernel-based Virtual Machine.

### L

**LVM:** Logical Volume Manager.

**LDAP:** Lightweight Directory Access Protocol

### N

**NASA:** National Aeronautics and Space Administration.

**NAS:** Network Attached Storage.

**NFS:** Network File System.

**NaaS:** Network as a Service.

## **U**

**URI:** Uniform Resource Locator.

**UTC :** Temps Universel Coordonné

## **P**

**PAAS:** Platform as a Service.

**PC:** Personal Computer.

## **S**

**SCSI:** Small Computer System Interface

**SDK:** Software Development Kits.SAN

Iscsi: Internet Small Compute System Interface.

**SQL:** Structured Query Language.

**SAAS:** Software as a Service.

**SOAP:** Simple Object Access Protocol.

**SAN:** Storage Area Network.

**SSL:** Secure Sockets Layer.

**SSO:** Single Sign On.

**S3:** Simple Storage Service.

## **V**

**VLAN:** Virtual Local Area Network.

**VM:** Virtual Machine.

**VPN:** Virtual Private Network.

**VG :** Volum Group

## Annexes

### 1 .Etapes d'installation d'openStack :

#### 1.1 Télécharge OpenStack Grizzly:

Entre à fichier `#vi /etc/apt/sources.list.d/ cloud-archive.list` puis saisie:

**deb http://ubuntu -cloud.archive.canonical.com/ubuntu precise- updates/grizzly main,**  
puis sauvegarde et quitter par commande « `:wq` ».

Et Entre à fichier `#vi cd /etc/apt/sources.list.d/ grizzly.list` puis saisie les lignes suivants :

**deb http://archive.gplhost.com/debian grizzly main**

**deb http://archive.gplhost.com/debian grizzly-backports main**

1.2 Configuration d'internet par fichier `#vi /etc/network/interfaces` saisie les commandes suivent puis sauvegarde et quitter par commande « `:wq` » :

```
# Internal Network
auto eth0
iface eth0 inet static
address 10.10.10.10
netmask 255.255.255.0

# External Network
auto eth1
iface eth1 inet dhcp
```

1.3 Vérification les normes de fichier `# vi/etc/ sysctl.conf`:

**net.ipv4.conf.all.rp\_filter = 0** et **net.ipv4.conf.default.rp\_filter = 0.**

1.4 Ajouter les adresse des serveurs, ils ont saisi dans fichie `# vi /etc/ hosts`:

```
127.0.0.1 localhost
10.10.10.10 cloud
10.10.10.9 network
10.10.10.11 c01
```

1.5 MySQL Database Service:

Pour accès le root Mysql nous utilisons le commande `:# mysql -u root -p <<EOF`, puis la création les différentes bases de données des composent OponStack :

**create database nova;** commande création le basse de donnée

**grant all privileges on nova.\* TO 'nova'@'localhost'identified by 'passnova';** commande défini les droits d'accès de utilisateur.

**create database cinder;**

```
grant all privileges on cinder.* TO 'cinder'@'localhost'identified by 'passcind';
create database glance;
grant all privileges on glance.* to 'glance'@'localhost'identified by 'passglan';
create database keystone;
grant all privileges on keystone.* to 'keystone'@'localhost'identified by 'passkeys';
create database quantum;
grant all privileges on quantum.* to 'quantum'@'localhost'identified by 'passquan';
grant all privileges on quantum.* to 'quantum'@'10.10.10.9'identified by 'passquan';
grant all privileges on quantum.* to 'quantum'@'10.10.10.11'identified by 'passqua';
flush privileges; EOF
```

Les bases de données ont été créées automatiquement.

### 1.6 OpenStack Identity Service

Accéder au fichier `#vi/etc/keystone/keystone.conf` et saisir ce qui suit :

```
[DEFAULT]
```

```
admin_token =key, le mot de passe de l'administrateur.
```

```
debug = True
```

```
verbose = True
```

```
[sql]
```

```
connection = mysql://keystone:passkeys@localhost/keystone, pour la connexion à
base de données
```

### 1.7 redémarrer et lancement des bases de données

```
# service keystone restart puis # keystone-manage db_sync.
```

1.8 Créer un fichier `openrc` # `vi openrc` puis saisir les informations suivantes :

```
export OS_TENANT_NAME=admin
```

```
export OS_USERNAME=admin
```

```
export OS_PASSWORD=key
```

```
export OS_AUTH_URL="http://localhost:5000/v2.0/"
```

```
export SERVICE_ENDPOINT="http://localhost:35357/v2.0"
```

```
export SERVICE_TOKEN=key
```

Puis lancer la commande `source ~/openrc`, fait le fichier précédent une source dans l'environnement.

Puis la commande `echo "source ~/openrc" >> ~/.bashrc` charger le fichier précédent dans le fichier `bashrc`, est le fichier lancé automatiquement au démarrage.



**1.9** Nous créons un fichier **keystone .sh** contient les informations suivants :

Projects: admin et services

Roles: admin, Member

Users: admin, demo, nova, glance, quantum, and cinder

Services: compute, volume, image, identity, ec2, and network

### Modèle de données dans keystone

Utilisateur: Nom pour les identifiants du compte, est associé à un ou plusieurs tenants (groupes ou projets).

Tenant : unité de propriété dans OpenStack, contient un ou plusieurs utilisateurs.

Rôle: qui définit le rôle de l'utilisateur sur un *Tenant*, aussi un utilisateur peut avoir un ou plusieurs rôles sur différents Tenants.

Token: certificat d'identification associé à un utilisateur ou à l'utilisateur et le tenant.

endpoint : Une adresse réseau accessible, généralement décrite par URL, où un service peut être accessible.

Function `get_field ()` pour récupérer les IP.

```
#!/bin/bash
```

```
# Modify these variables as needed
```

```
ADMIN_PASSWORD=${ADMIN_PASSWORD:-key}
```

```
SERVICE_PASSWORD=${SERVICE_PASSWORD:-$ADMIN_PASSWORD}
```

```
DEMO_PASSWORD=${DEMO_PASSWORD:-$ADMIN_PASSWORD}
```

```
export OS_SERVICE_TOKEN="key"
```

```
export OS_SERVICE_ENDPOINT="http://localhost:35357/v2.0"
```

```
SERVICE_TENANT_NAME=${SERVICE_TENANT_NAME:-service}
```

```
#
```

```
MYSQL_USER=keystone
```

```
MYSQL_DATABASE=keystone
```

```
MYSQL_HOST=localhost
```

```
MYSQL_PASSWORD=passkeys
```

```
#
```

```
KEYSTONE_REGION=RegionOne
```

```
KEYSTONE_HOST=10.10.10.10
```

```
# Shortcut function to get a newly generated ID
```

```
function get_field() {
```

```
while read data; do
if [ "$1" -lt 0 ]; then
field="(\$(NF$1))" else field="\${$((1 + 1))"
fi echo "$data" | awk -F'[ \t]*\\| [ \t]*' '{print $field}'
done}
# Tenants
ADMIN_TENANT=$(keystone tenant-create --name=admin | grep " id " | get_field 2)
DEMO_TENANT=$(keystone tenant-create --name=demo | grep " id " | get_field 2)
SERVICE_TENANT=$(keystone tenant-create --name=$SERVICE_TENANT_NAME |
grep " id " | get_field 2)
# Users
ADMIN_USER=$(keystone user-create --name=admin --
pass="$ADMIN_PASSWORD" --
email=admin@domain.com | grep " id " | get_field 2)
DEMO_USER=$(keystone user-create --name=demo --pass="$DEMO_PASSWORD" --
email=demo@domain.com --tenant-id=$DEMO_TENANT | grep " id " | get_field 2)
NOVA_USER=$(keystone user-create --name=nova --pass="$SERVICE_PASSWORD"
--tenant-id $SERVICE_TENANT --email=nova@domain.com | grep " id " | get_field 2)
GLANCE_USER=$(keystone user-create --name=glance --
pass="$SERVICE_PASSWORD"--tenant-id $SERVICE_TENANT --
email=glance@domain.com | grep " id " |get_field 2)
QUANTUM_USER=$(keystone user-create --name=quantum --pass=
"$SERVICE_PASSWORD" --tenant-id $SERVICE_TENANT --
email=quantum@domain.com | grep " id " | get_field 2)
CINDER_USER=$(keystone user-create --name=cinder --
pass="$SERVICE_PASSWORD"--tenant-id $SERVICE_TENANT --
email=cinder@domain.com | grep " id " |get_field 2)
# Roles
ADMIN_ROLE=$(keystone role-create --name=admin | grep " id " | get_field 2)
MEMBER_ROLE=$(keystone role-create --name=Member | grep " id " | get_field 2)
# Add Roles to Users in Tenants
keystone user-role-add --user-id $ADMIN_USER --role-id $ADMIN_ROLE --tenantid
$ADMIN_TENANT
```

---

```

keystone user-role-add --tenant-id $SERVICE_TENANT --user-id $NOVA_USER --
role-id $ADMIN_ROLE
keystone user-role-add --tenant-id $SERVICE_TENANT --user-id $GLANCE_USER --
role-id $ADMIN_ROLE
keystone user-role-add --tenant-id $SERVICE_TENANT --user-id $QUANTUM_USER
--role-id $ADMIN_ROLE
keystone user-role-add --tenant-id $SERVICE_TENANT --user-id $CINDER_USER --
role-id $ADMIN_ROLE
keystone user-role-add --tenant-id $DEMO_TENANT --user-id $DEMO_USER --roleid
$MEMBER_ROLE
# Create services
COMPUTE_SERVICE=$(keystone service-create --name nova --type compute --
description 'OpenStack Compute Service' | grep " id " | get_field 2)
VOLUME_SERVICE=$(keystone service-create --name cinder --type volume --
description 'OpenStack Volume Service' | grep " id " | get_field 2)
IMAGE_SERVICE=$(keystone service-create --name glance --type image --
description 'OpenStack Image Service' | grep " id " | get_field 2)
IDENTITY_SERVICE=$(keystone service-create --name keystone --type identity
--description 'OpenStack Identity' | grep " id " | get_field 2)
EC2_SERVICE=$(keystone service-create --name ec2 --type ec2 --description
'OpenStack EC2 service' | grep " id " | get_field 2)
NETWORK_SERVICE=$(keystone service-create --name quantum --type network --
description 'OpenStack Networking service' | grep " id " | get_field 2)
# Create endpoints
keystone endpoint-create --region $KEYSTONE_REGION --service-id
$COMPUTE_SERVICE --publicurl 'http://"$KEYSTONE_HOST":8774/v2/
$(tenant_id)s' --adminurl 'http://"$KEYSTONE_HOST":8774/v2/$(tenant_id)s'
--internalurl 'http://"$KEYSTONE_HOST":8774/v2/$(tenant_id)s'
keystone endpoint-create --region $KEYSTONE_REGION --service-id
$VOLUME_SERVICE --publicurl 'http://"$KEYSTONE_HOST":8776/v1/
$(tenant_id)s' --adminurl 'http://"$KEYSTONE_HOST":8776/v1/$(tenant_id)s'
--internalurl 'http://"$KEYSTONE_HOST":8776/v1/$(tenant_id)s'
keystone endpoint-create --region $KEYSTONE_REGION --service-id

```

---

```

$IMAGE_SERVICE --publicurl 'http://"$KEYSTONE_HOST":9292/v2' --
adminurl 'http://"$KEYSTONE_HOST":9292/v2' --internalurl 'http://
"$KEYSTONE_HOST":9292/v2'
keystone endpoint-create --region $KEYSTONE_REGION --service-id
$IDENTITY_SERVICE --publicurl 'http://"$KEYSTONE_HOST":5000/v2.0' --
adminurl 'http://"$KEYSTONE_HOST":35357/v2.0' --internalurl 'http://
"$KEYSTONE_HOST":5000/v2.0'
keystone endpoint-create --region $KEYSTONE_REGION --service-id
$EC2_SERVICE
--publicurl 'http://"$KEYSTONE_HOST":8773/services/Cloud' --adminurl
'http://"$KEYSTONE_HOST":8773/services/Admin' --internalurl 'http://
"$KEYSTONE_HOST":8773/services/Cloud'
keystone endpoint-create --region $KEYSTONE_REGION --service-id
$NETWORK_SERVICE --publicurl 'http://"$KEYSTONE_HOST":9696/' --
adminurl 'http://"$KEYSTONE_HOST":9696/' --internalurl 'http://
"$KEYSTONE_HOST":9696/'.

```

#### 1.10 Configuration Glance:

Crée dans les fichiers `#vi /etc/glance/glance-api.conf` et `#vi /etc/glance/glance-registry.conf`, les données suivants

```

[DEFAULT]
sql_connection = mysql://glance:passglan@localhost/glance
[keystone_authtoken]
admin_tenant_name = service
admin_user = glance
admin_password = key

```

#### 1.11 Redémarrer les services glance:

```

# service glance-api restart
# service glance-registry restart

```

#### 1.12 Télécharger et importer Ubuntu 12.04 LTS image UEC et par:

```

# wget http://uec-images.ubuntu.com/releases/12.04/release/ubuntu-12.04-
server-cloudimg-amd64-disk1.img
# glance image-create --is-public true --disk-format qcow2 --containerformat
bare --name "Ubuntu" < ubuntu-12.04-server-cloudimg-amd64-disk1.img

```

Et Télécharger et importer Cirros QCOW2 Image par:

```
# wget http://download.cirros-cloud.net/0.3.1/cirros-0.3.1-x86_64-disk.img
# glance image-create --is-public true --disk-format qcow2 --containerformat
  bare --name "Cirros" < cirros-0.3.1-x86_64-disk.img
```

1.13 accès le fichier #vi cd /etc/nova/api-paste.ini, et saisie

```
admin_tenant_name = service
admin_user = nova
admin_password = key
```

1.14 la configuration principal de Nova, saisie les données suivant dans le fichier #vi / etc / nova / nova.conf:

```
[DEFAULT]
sql_connection=mysql://nova:passnova@localhost/nova
rabbit_password=password
auth_strategy=keystone
# Networking
network_api_class=nova.network.quantumv2.api.API
quantum_url=http://10.10.10.10:9696
quantum_auth_strategy=keystone
quantum_admin_tenant_name=service
quantum_admin_username=quantum
quantum_admin_password=key
quantum_admin_auth_url=http://10.10.10.10:35357/v2.0
libvirt_vif_driver=nova.virt.libvirt.vif.LibvirtHybridOVSBridgeDriver
linuxnet_interface_driver=nova.network.linux_net.LinuxOVSIInterfaceDriver
# Security Groups
firewall_driver=nova.virt.firewall.NoopFirewallDriver
security_group_api=quantum
# Metadata
quantum_metadata_proxy_shared_secret=password
service_quantum_metadata_proxy=true
metadata_listen = 10.10.10.10
metadata_listen_port = 8775
# Cinder
```

---

```

volume_api_class=nova.volume.cinder.API
# Glance
glance_api_servers=10.10.10.10:9292
image_service=nova.image.glance.GlanceImageService
# novnc
novnc_enable=true
novncproxy_port=6080
novncproxy_host=10.0.0.10
vncserver_listen=0.0.0.0

```

**1.15** redémarrer les services Nova par:

```

# service nova-api restart
# service nova-cert restart
# service nova-consoleauth restart
# service nova-scheduler restart
# service nova-novncproxy restart

```

**1.16** La configurer et démarrer les services iSCSI par:

```

# sed -i 's/false/true/g' /etc/default/iscsitarget
# service iscsitarget start
# service open-iscsi start

```

**1.17** La configuration de Cinder, saisie dans le fichier **# vi cd /etc/cinder/cinder.conf** les données suivants:

**[DEFAULT]**

```

sql_connection = mysql://cinder:passcind@localhost/cinder
rabbit_password = passrabb

```

Et dans fichier **# vi cd /etc/cinder/api-paste.ini** les données suivants:

```

admin_tenant_name = service
admin_user = cinder
admin_password = key

```

**1.18** la création du volume physique LVM et volume logique par:

```

# pvcreate /dev/sdb
# vgcreate cinder-volumes /dev/sdb

```

**1.19** redémarrer les services Cinder par:

```

# service cinder-api restart

```

---

```
# service cinder-scheduler restart
```

```
# service cinder-volume restart
```

**1.20** La configuration de quantum, saisie sur le fichier # vi /etc/quantum/quantum.conf les données suivants:

```
[DEFAULT]
```

```
verbose = True
```

```
rabbit_password = passrabb
```

```
  [keystone_authtoken]
```

```
  admin_tenant_name = service
```

```
  admin_user = quantum
```

```
  admin_password = key
```

Et dans le fichier #vi cd /etc/quantum/plugins/openvswitch/ovs\_quantum\_plugin.ini la suite:

```
[DATABASE]
```

```
sql_connection = mysql://quantum:passquan@localhost/quantum
```

```
[OVS]
```

```
tenant_network_type = gre
```

```
tunnel_id_ranges = 1:1000
```

```
enable_tunneling = True
```

```
local_ip = 10.10.10.10
```

**1.21** La configuration de réseau de Network Node, saisie dans le fichier

```
#vi/etc/network/interfaces:
```

```
#Internal Network
```

```
  auto eth0
```

```
  iface eth0 inet static
```

```
  address 10.10.10.9
```

```
  netmask 255.255.255.0
```

```
# External Network
```

```
  auto eth1
```

```
  iface eth1 inet dhcp
```

**1.22** création de réseau bridge interne et externe par:

```
# ovs-vsctl add-br br-ex
```

```
# ovs-vsctl add-port br-ex eth0
```

---

```
# ovs-vsctl add-br br-int
```

1.23 La configuration de bridges, saisie dans fichier # **vi cd /etc/network/interfaces :**

- Pour modifier d'eth0 :

```
Auto eth0
iface eth0 inet manual
up ip address add 0/0 dev $IFACE
up ip link set $IFACE up
down ip link set $IFACE down
```

- Ajouter br-ex

```
auto br-ex
iface br-ex inet static
address 10.0.0.9
netmask 255.255.255.0
gateway 10.0.0.1
```

1.24 La configuration quantum, saisie dans le fichier #**vi cd /etc/quantum/quantum.conf:**

```
[DEFAULT]
verbose = True
rabbit_password = passrabb
rabbit_host = 10.10.10.10
[keystone_authtoken]
auth_host = 10.10.10.10
admin_tenant_name = service
admin_user = quantum
admin_password = key
```

Et dans fichier #**vi cd /etc/quantum/plugins/openvswitch/ovs\_quantum\_plugin.ini:**

```
[DATABASE]
sql_connection = mysql://quantum:passquan@10.10.10.10/quantum
[OVS]
tenant_network_type = gre
tunnel_id_ranges = 1:1000
enable_tunneling = True
local_ip = 10.10.10.9
[SECURITYGROUP]
```



```
firewall_driver = quantum.agent.linux.iptables_firewall.
```

```
OVSHybridIptablesFirewallDriver
```

Et dans le fichier `#vi cd /etc/quantum/dhcp_agent.ini`:

```
[DEFAULT]
```

```
enable_isolated_metadata = True
```

```
enable_metadata_network = True
```

Et dans le fichier `#vi cd /etc/quantum/metadata_agent.ini`:

```
[DEFAULT]
```

```
auth_url = http://10.10.10.10:35357/v2.0
```

```
auth_region = RegionOne
```

```
admin_tenant_name = service
```

```
admin_user = quantum
```

```
admin_password = password
```

```
nova_metadata_ip = 10.10.10.10
```

```
metadata_proxy_shared_secret = password
```

1.25 démarrer les services quantum par.

```
# service quantum-plugin-openvswitch-agent start
```

```
# service quantum-dhcp-agent restart
```

```
# service quantum-metadata-agent restart
```

```
# service quantum-l3-agent restart
```

1.26 Supprimer l'adresse IP d'eth0 pour ajouter à br-ex.

```
# ip addr del 10.0.0.9/24 dev eth0
```

```
# ip addr add 10.0.0.9/24 dev br-ex
```

1.27 La configuration de réseau de compte Node, saise les données suivant dans le fichier

`#vi/etc/network/interfaces`:

```
# Internal Network
```

```
auto eth0
```

```
iface eth0 inet static
```

```
address 10.10.10.11
```

```
netmask 255.255.255.0
```

```
# External Network
```

```
auto eth1
```

```
iface eth1 inet dhcp
```

**1.28** La configuration de service Nova :

saisie dans le fichier **#vi cd /etc/nova/api-paste.ini:**

```
[filter:authtoken]
auth_host = 10.10.10.10
admin_tenant_name = service
admin_user = nova
admin_password = key
```

Et dans le fichier **#vi cd /etc/nova/nova.conf:**

```
[DEFAULT]
# General
verbose=True
rabbit_host=10.10.10.10
rabbit_password=passrabb
auth_strategy=keystone
ec2_host=10.10.10.10
ec2_url=http://10.10.10.10:8773/services/Cloud
# Networking
libvirt_use_virtio_for_bridges=True
network_api_class=nova.network.quantumv2.api.API
quantum_url=http://10.10.10.10:9696
quantum_auth_strategy=keystone
quantum_admin_tenant_name=service
quantum_admin_username=quantum
quantum_admin_password=key
quantum_admin_auth_url=http://10.10.10.10:35357/v2.0
# Security Groups
firewall_driver=nova.virt.firewall.NoopFirewallDriver
security_group_api=quantum
# Compute #
compute_driver=libvirt.LibvirtDriver
connection_type=libvirt
# Cinder
volume_api_class=nova.volume.cinder.API
```

```
# Glance
glance_api_servers =10.10.10.10:9292
image_service=nova.image.glance.GlanceImageService
# novnc
vnc_enabled=true
vncserver_proxycient_address=10.10.10.11
novncproxy_base_url=http://10.0.0.10:6080/vnc_auto.html
vncserver_listen=0.0.0.0
```

1.29 La configuration quantum :

Saisie dans fichier **#vi cd /etc/quantum/quantum.conf:**

```
[DEFAULT]
rabbit_host = 10.10.10.10
rabbit_password = passrabb
verbose = True
```

Et dans fichier **#vi cd/etc/quantum/plugins/openvswitch/ovs\_quantum\_plugin.ini:**

```
[DATABASE]
sql_connection = mysql://quantum:passquan@10.10.10.1/quantum
[OVS]
tenant_network_type = gre
tunnel_id_ranges = 1:1000
local_ip = 10.10.10.11
enable_tunneling = True
[SECURITYGROUP]
firewall_driver = quantum.agent.linux.iptables_firewall.
OVSHybridIptablesFirewallDriver
```

2. L'installation d'eucalyptus :

2.1 L'installation de bridge par: **apt-get install bridge-utils**

Change les données de fichier **#vi/etc/network/interfaces**

```
auto lo
iface lo inet loopback
auto br0
iface br0 inet dhcp
bridge_ports eth0
```

**2.2** L'installation et redémarrer le serveur ntp

**apt-get install ntp**

**service ntp start**

**2.3** L'installation des packages par

**apt-get install eucalyptus-cloud eucalyptus-cc eucalyptus-sc eucalyptus-walrus**

**eucalyptus-nc**

**2.4** Redémarrer les services CLC, SC et Walrus par: **#service eucalyptus-cloud start**

Redémarrer CC par **:#service eucalyptus-cc start**

Redémarrer Nc **#service eucalyptus-nc start**

## Résumé :

Cloud computing est la convergence et l'évolution de plusieurs concepts de virtualisation, des applications distribuées, grille computing et d'autres pour permettre une approche plus souple pour le déploiement et la mise à l'échelle des applications et des services délivrer via le nuage (Internet).

Le Cloud Computing, comme un service public, a le potentiel de transformer l'IT de l'industrie par la fournissant des softwares, des plates-formes et des infrastructures comme un service, et en révolutionnant les moyens et les manières traditionnel de gestion d'IT de l'entreprise.

Plusieurs plates-formes de Cloud Computing ont été lancées au cours des dernières années.

Avec l'avènement de plusieurs plates-formes Open Source et commercial garantissant une disponibilité et une performance, il n'est pas facile pour un non expert utilisateurs à choisir une parmi eux sans comprendre les caractéristiques et les avantages de chacun de ces plate-forme. Cette mémoire analyse les caractéristiques, l'architecture d'un certains des open sources solution plus populaire de l'informatique au nuage exhaustivement OpenStack et EUCALYPTUS pour aider l'utilisateur débutant de choisir parmi eux une plate-forme qui répond et satisfait ses besoins.

**Mots clés :** Cloud computing, SaaS, PaaS, IaaS, Open source, OpenStack, Eucalyptus, Microsoft Windows Azure, VMware vCloud et virtualisation.

## المخلص:

الحوسبة السحابية تتضمن العديد من المفاهيم الافتراضية، والتطبيقات الموزعة، وغيرها لنشر وتوزيع مختلف الخدمات و التطبيقات بشكل سهل ومرن، وذلك عن طريق السحابة (الإنترنت).

الحوسبة السحابية لديها القدرة على تغيير و تطوير تكنولوجيا المعلوماتية من خلال توفير مختلف البرمجيات و المنصات وخدمة البيانات التحتية. حيث أنها أحدثت ثورة على مستوى الوسائل والطرق المستعملة في ادارة. في السنوات الأخيرة تم اطلاق العديد من المنصات الحوسبة السحابية منها: منصات حرة و منصات تجارية هنا ليس من السهل على المستخدمين الغير خبراء الأختيار واحد منهما،دون معرفة مزايا وسلبيات كل منصة. هذه الأطرحة تتضمن تحليل للخصائص و الهندسة التكوينية لمختلف المنصات الحوسبة، بالأخص OpenStack و Eucalyptus ، وذلك لمساعدة المستخدمين الأختيار بينهما على حسب احتياجاتهم.

**الكلمات المفتاحية :** Cloud Computing, SaaS, PaaS, IaaS, Open Source, OpenStack, Eucalyptus,

Microsoft, Windows Azure, VMware vCloud et virtualisation.

## Abstract:

Cloud computing is the convergence and evolution of several concepts from virtualization, distributed application, grid computing, and others to enable a more flexible approach for deploying and scaling applications. Cloud computing, as a utility, has the potential to transform the IT industry by providing software; platform and infrastructure as a service.

Several Cloud computing platforms have been launched in the last few years. With the advent of several Open Source and commercial Cloud computing platforms, guaranteeing performance and uptime it is not easy for a non expert users to choose from the different platforms without comprehending the characteristics and advantages of each of these platforms. This thesis analyses the characteristics, architecture and applications of some of the most popular open source cloud computing platforms like OpenStack and EUCALYPTUS to help novice user to select one solution from the different cloud platforms.

**Key words:** Cloud computing, SaaS, PaaS, IaaS, Open source, OpenStack, Eucalyptus, Microsoft Windows Azure, VMware vCloud and virtualisation.