Chapitre1 Introduction à la virtualisation

1. Définition

La virtualisation est une technique informatique consistant à faire fonctionner plusieurs environnements logiques indépendants séparément sur une même machine. Il s'agit d'une extension du principe d'émulation. L'émulation consiste à substituer un ou plusieurs éléments informatiques par une application. Appliquée à la virtualisation, un système prétend être plusieurs systèmes différents.

La virtualisation fait appel au multiplexage des systèmes d'exploitation comme on le trouve au niveau des processus. Différents processus cohabitent indépendamment tout en se partageant les ressources physiques de la machine. Un processus ne peut pas monopoliser toutes les ressources de calcul puisque dans ce cas-là le système d'exploitation reprendra le contrôle pour allouer des ressources à d'autres processus. Au niveau des ressources de stockage, chaque processus a son espace virtuel d'adresse en mémoire lui donnant l'impression qu'il est seul à utiliser la mémoire vive. Historiquement cela fait appel à plusieurs techniques déjà connues comme : mémoire virtuelle, machine java, réseau virtuel, circuit virtuel,…etc.

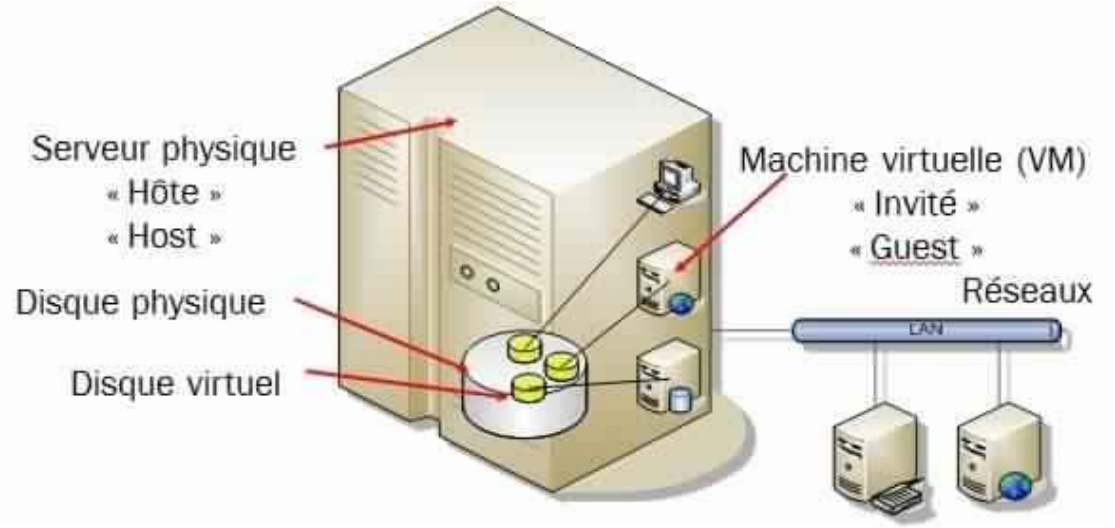

Figure.1 La virtualisation (des serveurs, disques, réseaux,…)

2. Historique

Historiquement une grande partie des travaux de recherche ont été fait par IBM dans les années1960 au centre de recherche de Grenoble aujourd'hui fermé. Ils développèrent un système expérimental faisant partie du projet System/360 (Main Frame) appelé VM/CMS (Virtual Machine / Conversation Monitor System). CMS est le système d'utilisation qui s'appuie sur VM. Une caractéristique de ce premier système de virtualisation était le fait que chaque CMS était attribué à un seul utilisateur, sachant que plusieurs CMS fonctionnaient sur la VM. Nous pouvons faire une analogie par rapport à la terminologie actuelle entre VM et hyperviseur ainsi qu'entre CMS et environnement logiciel invité.

Le System/360 était déjà capable de gérer de la virtualisation récursive. La finalité de ce produit d'IBM était de pouvoir consolider les postes de travail liés. La dernière implémentation par IBM de VM est z/VM qui fonctionnait sur les zSeries.

Entre la fin des années 80 et le milieu des années 90 Commodore International commercialise l'Amiga qui est ordinateur personnel très populaire à l'époque. Il était aussi bien capable de lancer des pc X386, des Macintoshs 6800 et des solutions X11 en multitâches.

Suite à l'Amiga on trouve des systèmes Unix basés sur l'architecture NUMA, qui est une architecture mémoire de systèmes multiprocesseurs. Cette architecture consiste à cloisonner et partitionner la mémoire, les accès se faisant via de multiples bus, un par processeur.

En 1999 VMware proposa un système propriétaire de virtualisation de systèmes x86 à base de systèmes hôtes x86. D'autres projets libres ont suivi VMware, tels que QEMU, Xen, Bochs, kvm, VirtualBox ainsi que des logiciels gratuits mais propriétaires tels que VirtualPC, VMware Server,Virtual Server.

Dans les années 2000 afin d'améliorer les capacités des solutions de virtualisation. Les fabricants de processeurs Intel et AMD ont implémenté des fonctions de virtualisation dans leurs processeurs permettant la prise en charge de systèmes d'exploitation non modifiés plus efficacement.(intel VT et AMD-V).

3. Intérêts de la virtualisation

A l'époque où les ordinateurs n'étaient capables de ne faire exécuter qu'un seul processus en même temps, l'intérêt de se diriger vers un système supportant la gestion multiprocessus était de pouvoir optimiser les ressources de calcul. La

virtualisation va plus loin encore, nous allons voir ici les différents intérêts qu'elle porte.

- **La sécurité**

La virtualisation va permettre une isolation des différents environnements logiciels au niveau des ressources physiques. La communication entre les différentes machines virtuelles sera uniquement possible via des connexions réseau de manière identique à la communication entre deux machines physiques. L'isolation est telle que la compromission d'un système invité par du code malicieux ne pourra pas se propager à d'autres systèmes invités. Il sera donc tout à fait possible d'isoler chaque service sans devoir acheter un nouveau serveur à chaque fois.Un problème de sécurité pourrait apparaitre si un système invité avait la possibilité de lire les données mémoire ou disque d'un autre système invité. De la même façon, si deux applications de deux systèmes invités tentent d'accéder à une même ressource physique, il se produira une interférence. Avec un système de virtualisation, il sera possible de gérer les accès aux ressources physique de manière à éviter les conflits. Ceci pourra se faire soit par l'allocation de temps d'accès soit par l'allocation exclusive.

Une autre application très intéressante pour les professionnels et les chercheurs en sécurité est l'observation de logiciels malveillants à travers de systèmes d'exploitation invités surveillés. Ce type d'utilisation permettra de suivre l'évolution de l'infection d'un système. En outre, il sera possible de manipuler ces systèmes d'exploitation pour revenir en arrière et expérimenter diverses techniques de désinfection et de détection.

- **Le coût**

Actuellement, nous réalisons le fait que les serveurs sont largement sous utilisés. En effet, une étude Gartner a montré qu'en moyenne les serveurs sont utilisés à 5% de leur capacité. La sous-utilisation de serveurs empire avec la présence de serveurs en *hot spare*. Un tel serveur est un serveur qui attend un dysfonctionnement du primaire pour prendre le relais. En absence de dysfonctionnement, ce serveur n'effectue aucun travail alors qu'il consomme de l'énergie et prend de la place en centre de données. Avec les coûts grandissants des matières premières énergétiques et donc *a fortiori* l'électricité, les gestionnaires de parcs serveurs se retrouvent contraints de se soucier des enjeux d'optimisation de la consommation électrique. De plus, il est nécessaire de gérer des contraintes liées au manque

d'espace dans les centres de données dont le prix augmente à cause de la demande croissante.

Un hébergeur de serveurs va pouvoir bénéficier de la virtualisation par le biais d'une gestion de ressources plus fine. Il va lui être possible de gérer finement la répartition des ressources physiques entre les différents serveurs. Il va également bénéficier de fonctionnalités de comptabilité de l'utilisation des ressources pour pouvoir proposer une facturation plus adaptée à l'utilisation de ses clients.

- **Criticité et performances**

L'accumulation de systèmes invités sur un même système physique augmente sa criticité. Une panne matérielle rendra indisponibles tous les systèmes invités. Pour pallier à la criticité accrue du système physique, des solutions ont été prévues. La solution la plus basique est de prévoir la possibilité de faire des images des systèmes invités. Des taches très longues à effectuer vont pouvoir être sauvegardés en cours d'exécution. Il sera ensuite possible de relancer ces tâches à partir de ce point en cas de panne. Un autre exemple de l'utilisation de ce procédé est le clonage de différentes instances d'un invité. Par exemple, si plusieurs tâches nécessitent un même travail préliminaire long, il sera possible de faire effectuer dans une première instance ce travail, ensuite de la cloner autant de fois qu'il y a d'autres tâches et de les lancer séparément. Une solution plus efficace pour palier à la criticité des serveurs de virtualisation est la migration dynamique de systèmes invités. Le prérequis est d'avoir un support de stockage accessible en réseau. Un système invité est exécuté sur un premier serveur.

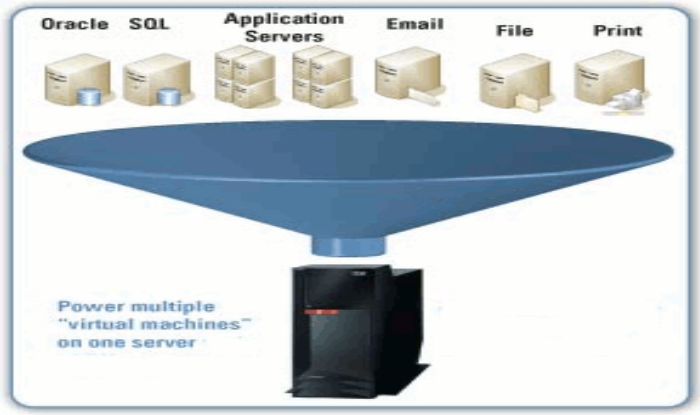

Figure.2 La consolidation des ressources

Dans le cas d'une panne matérielle ou d'une maintenance planifiée, il sera possible de migrer dans de très courts délais ce système vers une autre machine tout en sauvegardant son état avant l'interruption. Ce procédé permet à lui seul de réduire la criticité des systèmes individuels à un niveau largement acceptable. Il fait reposer une plus grande criticité sur les systèmes de stockage en réseau qui peuvent cependant être dédoublés.

Une autre utilité de ce procédé en termes de performances va être de pouvoir allouer des ressources à la volée aux systèmes invités. Dans le cas d'une montée en charge d'un système invité, il sera possible de l'isoler sur un serveur en déplaçant les autres systèmes présents à ces cotés sur d'autres serveurs moins chargés. Ce procédé peut également être utile dans un objectif de réduction de la consommation électrique d'un parc de serveur. Il est envisageable de n'utiliser que peu de machines lorsque le système est soumis à une faible charge et d'allumer progressivement les autres machines en fonction de la montée en charge. Ceci nécessite une gestion de l'allumage électrique du serveur par le réseau mais bon nombre de serveurs récents disposent de tels outils.

4. Conclusion

En fin, la virtualisation présente de nombreuses réponses à des problèmes qui se posent aujourd'hui que ce soit au niveau optimisation de l'utilisation des ressources informatiques matérielles ou logicielles avec une sécurité garantie, ou au niveau énergétique avec la forte augmentation du cout de l'énergie électrique (Green IT).

Chapitre 2 La virtualisation : modèles et techniques

1. Les types de virtualisation

Dans cette partie nous allons aborder les différents types de virtualisation. Nous pouvons voir à travers l'arbre ci-dessous (fig.3) les solutions que nous allons expliquer ci-après. Nous pouvons diviser la virtualisation en deux catégories, celle des systèmes et celle des processus. La virtualisation au niveau du système d'exploitation permet de faire fonctionner des environnements utilisateurs complètement cloisonnés sur un système d'exploitation unique. Du coté des systèmes on trouve deux types de virtualisation. Celle qui accepte des environnements invités non modifiés et inversement. La virtualisation totale permet de faire fonctionner différents types d'architecture et donc différents systèmes d'exploitation en même temps, ceci sur la base d'une machine avec un système d'exploitation complet. La virtualisation matérielle assistée quant à elle permet le fonctionnement de machines virtuelles sur différents OS en tirant partie des instructions processeurs dédiées à la virtualisation. La base de la machine qui accueille les machines virtuelles (MV) est une couche logicielle plus légère qu'un OS complet. L'émulation du matériel reste complète. Enfin la paravirtualisation demande de son côté une modification des MV afin de modifier la vue que celles-ci ont, du matériel sous-jacent. C'est le seul type de virtualisation qui n'émule pas de matériel pour les MV.

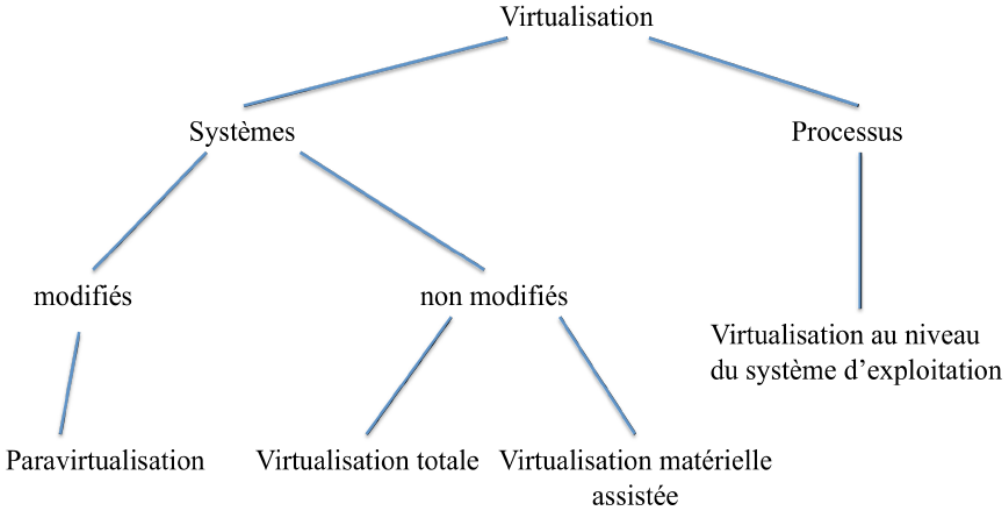

Figure.3 Les types de virtualisation

2. La virtualisation au niveau du système d'exploitation

Ce type de virtualisation consiste à séparer le système d'exploitation (OS) d'une machine en différents environnements utilisateurs distincts. Ainsi les utilisateurs de la machine ne se voient pas entre eux, et l'accès aux données des autres n'est pas possible. Les environnements utilisateurs sont entièrement cloisonnés. En effet ici le matériel et l'OS sont les même pour tout le monde. On ne peut pas parler de systèmes d'exploitations invités pour ces systèmes invités mais plutôt d'environnements utilisateur cloisonnés ou 'jails '.

L'isolation (aussi appelé cloisonnement) est une technique qui intervient au sein d'un même système d'exploitation. Elle permet de séparer un système en plusieurs contextes ou environnements. Chacun d'entre eux est régi par l'OS hôte, mais les programmes de chaque contexte ne sont capables de communiquer qu'avec les processus et les ressources associées à leur propre contexte. Il est ainsi possible de partitionner un serveur en plusieurs dizaines de contextes, presque sans ralentissement. L'isolation est utilisée sous Unix pour protéger les systèmes. Via des mécanismes comme chroot ou jail il est possible d'exécuter des applications dans un environnement qui n'est pas celui du système hôte, mais un mini système ne contenant que ce dont l'application a besoin, et n'ayant que des accès limités aux ressources. Il est possible également de lancer des programmes dans une autre distribution que celle du système principal. Avec l'isolation, l'espace noyau n'est pas différencié, il est unique, partagé entre les différents contextes. Mais on définit de multiples espaces utilisateurs cloisonnés. C'est ainsi que l'on peut faire cohabiter différentes distributions de système d'exploitation, à condition qu'elles partagent le même noyau. L'isolation des contextes est une solution légère, tout particulièrement dans les environnements Linux. La principale solution pour l'isolation est Linux-VServer, la plus mature et la plus avancée. OpenVZ est une alternative, qui se présente de la même façon et propose quasiment les mêmes fonctionnalités. Elle est à la base du produit commercial Virtuozzo. La figure 4 représente l'architecture d'une telle solution.

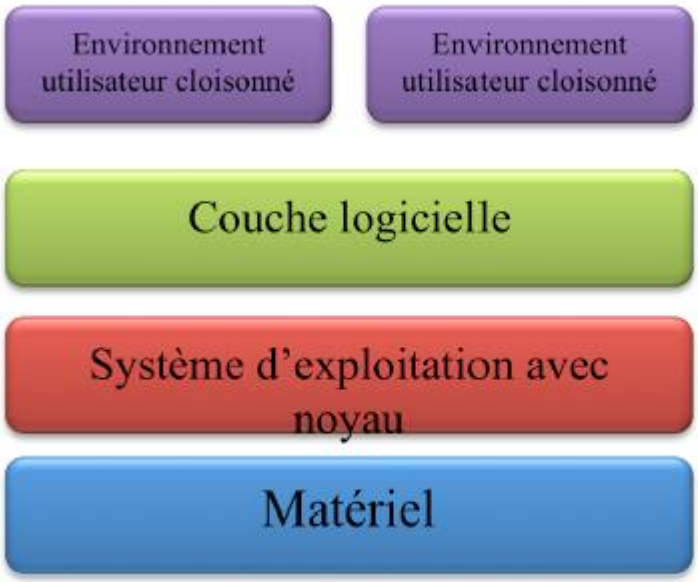

Figure.4 La virtualisation : isolation des environnements (Jail)

3. La virtualisation totale

La virtualisation totale consiste à émuler l'intégralité d'une machine physique pour le système invité. Le système invité « croit » s'exécuter sur une véritable machine physique. Le concept de virtualisation totale est déjà bien ancré dans la littérature, mais ce n'est pas toujours ce terme qui est employé. La virtualisation partielle peut être confondue avec. Nous allons voir dans les paragraphes qui suivent les définitions de ces deux termes. En virtualisation totale, la machine physique qui va émuler le matériel pour le système invité doit être doté d'un OS ainsi que d'une surcouche applicative. Un des gros intérêts de cette technique de virtualisation est de pouvoir émuler n'importe quelle architecture matérielle. On peut donc faire fonctionner les OS que l'on désire indépendamment de l'architecture du système hôte. On l'utilise essentiellement en milieu industriel pour faire fonctionner des applications sur des architectures matérielles non encore commercialisées. Il faut savoir que ce type de virtualisation n'est que logiciel, aucune fonctionnalité matérielle de virtualisation n'est utilisée. On peut voir sur la figure 5 que les 3 premières couches sont identiques à la virtualisation au niveau de l'OS. Ici sur une 4ème couche on trouve l'émulation du matériel désiré afin de pouvoir accueillir tout type d'OS en tant que machine virtuelle.

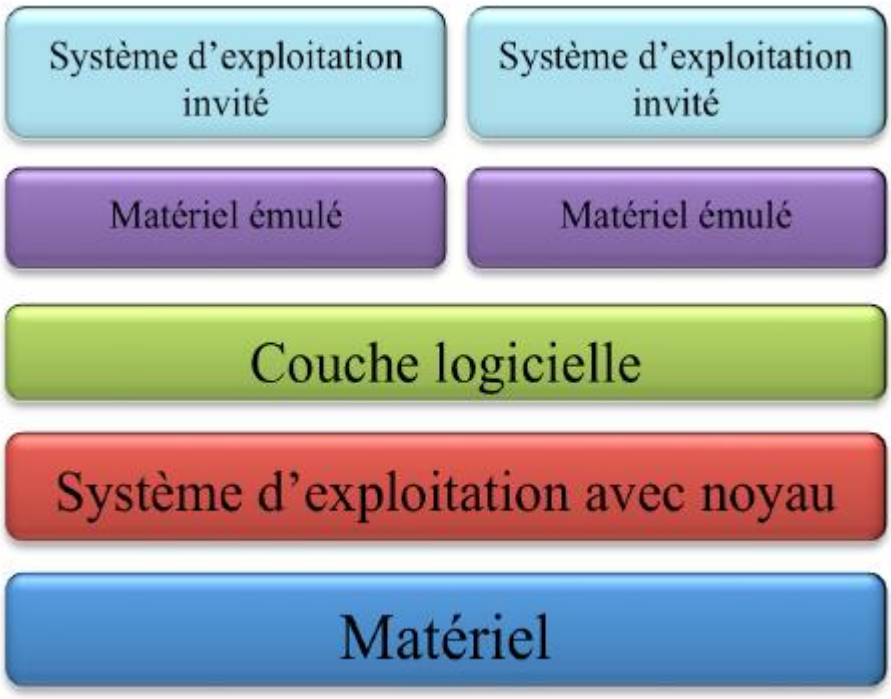

Figure.5 Virtualisation totale

La virtualisation partielle quant à elle n'a pas pour but de simuler de nouvelles architectures pour les systèmes invités, mais uniquement une partie. En effet on va émuler la partie matérielle qui nous intéresse pour faire fonctionner une application particulière. Dans ce cas l'émulation d'une seule ressource peut être suffisante. On parle donc de virtualisation partielle. L'intérêt est donc de ne pas ré-simuler tout le matériel de la plateforme émulée.

En virtualisation totale la machine accueillant les systèmes invités doit donc implémenter de façon logicielle une gestion complète du matériel des invités. La gestion de la mémoire est un des facteurs les plus critiques en terme de performances. Les performances de la machine virtuelle sont donc limitées par les performances de la couche d'abstraction du système hôte et par la qualité de l'émulation du matériel implémenté. La virtualisation totale est la technique qui s'éloigne le plus des performances que peut avoir un système classique. Exemples : VirtualBox, Workstation et Parallels.

4. La paravirtualisation

Cette technique présente un logiciel en tant qu'intermédiaire entre le matériel et les systèmes d'exploitation invités et non un système d'exploitation. La deuxième différence par rapport aux techniques vues précédemment est que les systèmes invités sont modifiés. Ils sont conscients du fait qu'ils sont virtualisés.

L'application située entre le matériel est les systèmes invités est appelée VMM (Virtual Machine Monitor) ou hyperviseur. C'est lui qui est chargé d'appliquer la politique d'accès aux ressources matérielles pour les systèmes invités. Etant donné des modifications apportées, ces systèmes ont été adaptés au niveau de leur noyau afin qu'ils puissent communiquer avec l'hyperviseur. Dans les systèmes natifs, il existe des instructions privilégiées. Ces instructions processeurs permettent d'accéder directement à la mémoire physique du processeur sans passer par la mémoire virtuelle. Une autre utilisation de ces instructions est la possibilité de changer l'état de la machine dans le sens ou cela influe sur des processus. En paravirtualisation cela pose un problème non négligeable puisque les machines virtuelles n'ont pas accès directement au matériel et ne peuvent ainsi pas faire exécuter ces instructions privilégiées. Ce problème ne s'était pas posé jusqu'alors puisque dans les techniques vues précédemment, le processeur est émulé donc le système invité exécute ces instructions privilégiées sur ce « faux » processeur. Mais un jeu d'instruction appelé « hypercalls » existe pour fournir des fonctionnalités similaires.

Les applications n'étant pas modifiées, celles-ci émettrons des appels systèmes privilégiés classiques. Ces appels seront captés par le noyau modifié du système invité et seront transformés en hypercalls. Ceux-ci seront alors envoyés à l'hyperviseur qui pourra donc les interpréter. Ce cheminement est expliqué par la figure ci-dessous (Fig.7). Exemples : Xen, KVM, ESX/ESXi et Hyper-V

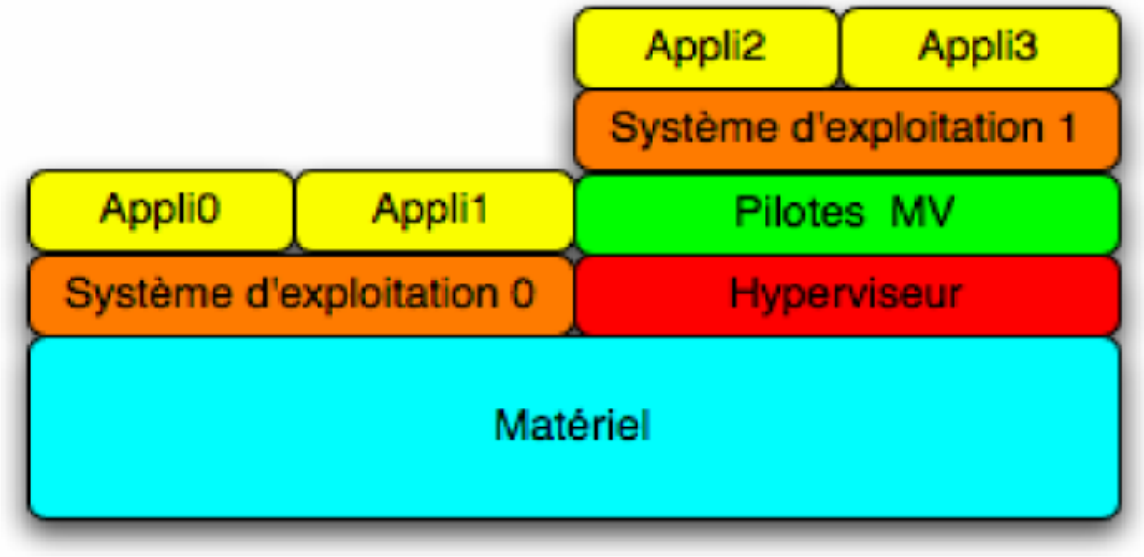

Figure.6 La paravirtualisation

5. La virtualisation matérielle assistée

Ce type de virtualisation a pour but de faire fonctionner des systèmes invités dont les OS peuvent être différents mais non modifiés. La différence avec la virtualisation totale est qu'ici on tire pleinement partie du matériel et de sa puissance. La perte de performances est minimum particulièrement au niveau du processeur.

Cette technique de virtualisation a été récemment implantée dans les processeurs à base d'architecture x86 (2003) sous les noms de : technologies Intel-VTx (32 bits) et Intel VT-i (64 bits) pour Intel, de AMD-V pour AMD, Advanced Power Virtualization pour IBM et de Ultra SPARC T1 Hypervisor pour SUN. Encore une fois on trouve plusieurs termes pour définir ce type de virtualisation, et ce sont les entreprises qui le dénomment différemment. Xen l'appelle HVM (Hardware Virtual Machine), Virtual Iron quand à lui la dénomme virtualisation native. Le but de ce type de virtualisation était de pouvoir faire fonctionner tout type d'OS non modifié de façon parallèle sur une même machine sans perte de performance. L'utilisation des fonctionnalités processeurs liées à la virtualisation a permis en partie cet exploit mais les différentes sources ne sont pas en accord vis-à-vis du facteur de perte de performance.

Des instructions sont ajoutées au processeur pour qu'il serve d'hyperviseur à l'aide du HAL (Hardware Abstraction Layer). Les systèmes invités sont au même niveau que ceux des hôtes (Fig. 7)

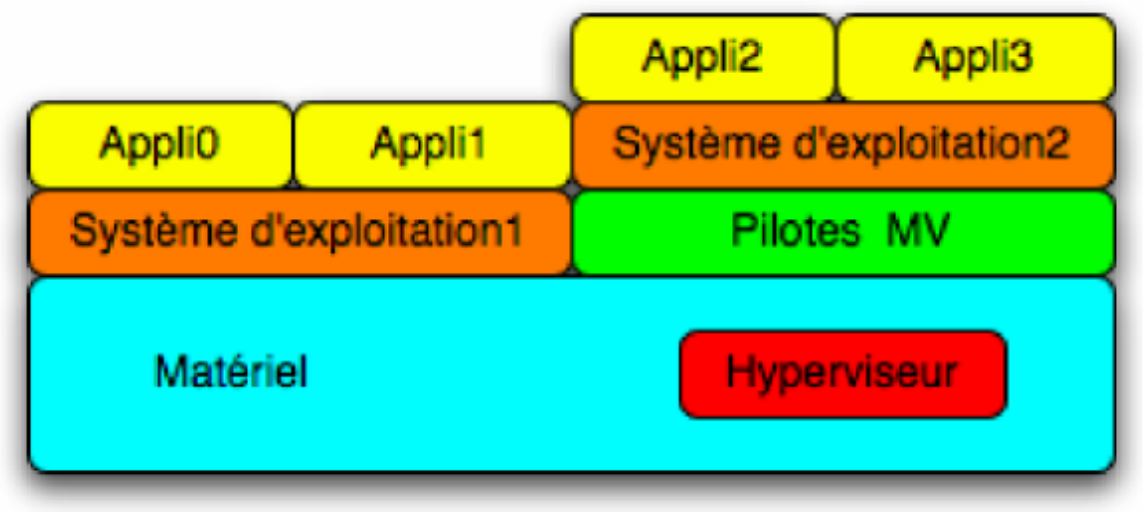

Figure.7 La virtualisation matérielle

6. Conclusion

La virtualisation permet d'ajouter une couche d'abstraction qui sépare le système d'exploitation du matériel afin de délivrer une meilleure utilisation et flexibilité des ressources de traitement.

La virtualisation repose sur trois éléments importants :

- L'abstraction des ressources informatiques ;
- La répartition des ressources par l'intermédiaire de différents outils, de manière à ce que celles-ci puissent être utilisées par plusieurs environnements virtuels ;
- La création d'environnements virtuels.

La virtualisation permet la consolidation des ressources, l'isolation des environnements, la sécurité des données, l'optimisation des ressources et la facilité de l'administration.

Chapitre 3 Le cloud computing

1. Introduction

Face à l'augmentation continuelle des coûts de mise en place et de maintenance des systèmes d'informations, les entreprises externalisent de plus en plus leurs services informatiques en les confiant à des entreprises spécialisées comme les fournisseurs de Cloud. L'intérêt principal de cette stratégie pour les entreprises réside dans le fait qu'elles ne paient que pour les services effectivement consommés.

Le Cloud Computing est aujourd'hui le sujet phare dans le domaine des systèmes d'information et de communication. Après la virtualisation, le Cloud parait être la révélation qui va permettre aux entreprises d'être plus performantes et de gérer le coût des systèmes d'information plus sereinement. En fait, le terme Cloud Computing, ou « informatique dans les nuages », est un nouveau modèle informatique qui consiste à proposer les services informatiques sous forme de services à la demande, accessibles de n'importe où, n'importe quand et par n'importe qui (autorisé). Cette nouvelle technologie permet à des entreprises d'externaliser le stockage de leurs données et de leur fournir une puissance de calcul supplémentaire pour le traitement de grosse quantité d'informations.

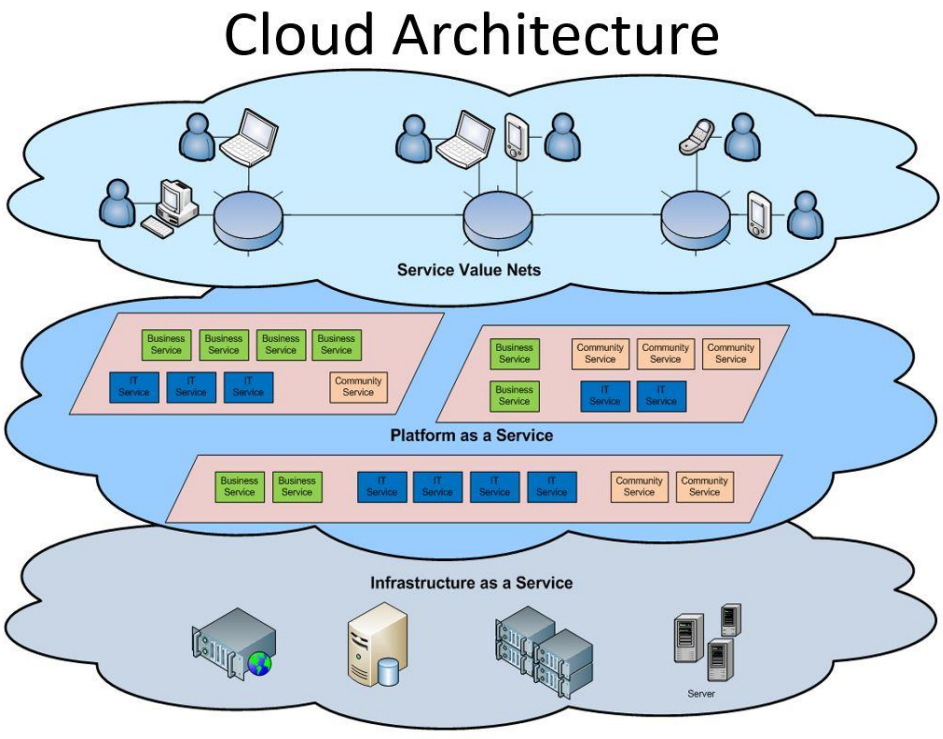

Figure.8 Architecture d'un cloud

2. En quête d'une définition

Techniquement, le concept de Cloud Computing est loin d'être nouveau, il est même présent depuis des décennies. On en trouve les premières traces dans les années 1960, quand John McCarty (MIT) affirmait que cette puissance de traitement informatique serait accessible au public dans le futur. Le terme en lui-même est apparu plus couramment aux alentours de la fin du XXe siècle et il semblerait que Amazon.com soit l'un des premiers à avoir assemblé des data-center et fournit des accès à des clients. Les entreprises comme IBM et Google ainsi que plusieurs universités ont seulement commencé à s'y intéresser sérieusement aux alentours de 2008, quand le Cloud Computing est devenu un concept à la mode.

Réalisant ce qu'ils pourraient faire de toute cette puissance, de nombreuses compagnies ont ensuite commencé à montrer un certain intérêt, puis à échanger leurs anciennes infrastructures et applications internes contre ce que l'on appelle les «pay per-use service».

Le cloud computing est fondé essentiellement sur trois concepts de base :

- Le grid computing qui consiste à mettre en commun des ressources logicielles et matérielles distribuées (ensemble que l'on appelle la « grille ») afin de fournir une puissance de calcul importante. La charge de travail est divisée en sous-tâches qui sont traitées en parallèle par les ressources de la grille, les résultats étant ensuite agrégés dans un résultat global rendu à l'utilisateur.
- La virtualisation qui implique la mutualisation, la consolidation et l'isolation
- L'utility computing ou l'informatique à la demande, qui consiste à offrir des services informatiques (serveurs, disques durs, réseaux, MV, logiciels …etc.) sans acquisition locale du matériel.

Le Cloud Computing, littéralement l'informatique dans les nuages est un concept qui consiste à déporter sur des serveurs distants des stockages et des traitements informatiques traditionnellement localisés sur des serveurs locaux ou sur le poste de l'utilisateur. Il consiste à proposer des services informatiques sous forme de service à la demande, accessible de n'importe où, n'importe quand et par n'importe qui, grâce à un système d'identification, via un PC et une connexion à Internet. Cette définition est loin d'être simple à comprendre ,toutefois l'idée

principale à retenir est que le Cloud n'est pas un ensemble de technologies, mais un modèle de fourniture, de gestion et de consommation de services et de ressources informatiques.

Pour Wikipédia, il s'agit: «d'un concept de déportation sur des serveurs distants des traitements informatiques traditionnellement localisés sur le poste client ».

Pour CISCO: «Le Cloud Computing est une plateforme de mutualisation informatique fournissant aux entreprises des services à la demande avec l'illusion d'une infinité de ressources» .

Pour le groupe de travail CIGREF le Cloud Computing est défini par les quatre points suivant :

- Un Cloud est toujours un espace virtuel.
- Les informations sont fragmentées.
- Les fragments sont toujours dupliqués et répartis dans cet espace virtuel, composé d'un ou plusieurs supports physiques.
- une console (programme) de restitution permet de reconstituer l'information.

Selon la définition du National Institute of Standards and Technology (NIST), le cloud computing est l'accès via un réseau de télécommunications, à la demande et en libre-service, à des ressources informatiques partagées configurables. Il s'agit donc d'une délocalisation de l'infrastructure informatique.

3. Les services du Cloud Computing

Le Cloud Computing propose essentiellement trois services dénommés généralement sous l'acronyme Xaas.

a. ***Iaas (Infrastructure as a Service)***

Il s'agit de la mise à disposition, à la demande, de ressources d'infrastructures dont la plus grande partie est localisée à distance dans des Data-centers. L'IaaS permet l'accès aux serveurs et à leurs configurations pour les administrateurs de l'entreprise. Le client a la possibilité de louer des clusters, de la mémoire ou du stockage de données. Le coût est directement lié au taux d'occupation. Une analogie peut être faîte avec le mode d'utilisation des industries des commodités (électricité, eau, gaz) ou des télécommunications. Eucalyptus est un exemple de logiciel qui

permet d'implémenter un service Iaas, AWS une solution offerte par Amazon , Azure est la solution de Microsoft et Compute Engine celle de Google.

Avantages

Grande flexibilité, contrôle total des systèmes, qui permet d'installer tout type de logiciel métier.

Inconvénients

Besoin d'administrateurs système comme pour les solutions de serveurs classiques sur site .

b. *Paas (Plateform as a Service)*

Platform as a Service, souvent appelé simplement PaaS, est une catégorie de services de Cloud computing qui fournit la plateforme et l'environnement informatique nécessaire aux développeurs pour mettre en place leurs différents services et applications sur Internet. Les services PaaS sont hébergés dans le Cloud et les utilisateurs y accèdent simplement, par leur navigateur web.

Les services Paas permettent aux utilisateurs de créer des applications logicielles en utilisant les outils fournis par le fournisseur. Ils peuvent prendre la forme de fonctionnalités pré-configurées auxquelles les clients peuvent souscrire, en ne choisissant que celles qui conviennent à leurs exigences. Cela signifie que les packs PaaS peuvent aller de la simple offre "point-and-click", où le client n'a pas besoin d'avoir des connaissances particulières en hébergement, à la fourniture d'options d'infrastructure pour développement avancé.

L'infrastructure et les applications sont gérées par le fournisseur et les clients peuvent accéder à des services de support si nécessaire. Les services sont constamment mis à jour, avec l'amélioration des fonctionnalités existantes et ajout de fonctionnalités additionnelles. Les fournisseurs de Paas sont en mesure d'assister les développeurs depuis la conception de leur idée originale jusqu'à la création de leurs applications, en passant par la phase de test et de déploiement, le tout grâce à un système de services managés.

On trouvera ci-dessous certaines des fonctionnalités pouvant être inclues dans une offre PaaS:

- Système d'exploitation
- Environnement de script serveur
- Système de gestion de bases de données
- Logiciel serveur
- Support
- Stockage
- Accès réseau
- Outils de design et de développement
- Hébergement

Avantages

Le déploiement est automatisé, pas de logiciel supplémentaire à acheter ou à installer.

Inconvénients

Limitation à une ou deux technologies (ex. : Python ou Java pour Google AppEngine, .NET pour Microsoft Azure, propriétaire pour force.com). Pas de contrôle des machines virtuelles sous-jacentes. Convient uniquement aux applications Web.

c. ***Saas (Software as a Service)***

Les Logiciels en tant que Service (traduction de SaaS, Software as a Service) désignent des logiciels qui sont hébergés sur le serveur d'un prestataire, accessibles à distance (par exemple au travers d'un navigateur web), et dont la facturation s'effectue sous forme d'abonnement, ou proportionnellement à l'utilisation de certaines ressources. En français, on parle couramment de solutions en "mode SaaS". Ce modèle s'oppose à la distribution de logiciels sous forme de produit, c'est à dire moyennant une licence qui donne droit à l'utilisateur d'installer le logiciel sur une ou plusieurs machines (qu'il s'agisse de serveurs ou d'ordinateurs personnels). Les logiciels peuvent concerner : le web, l'analyse de données, l'email, la gestion des clients (CRM), la bureautique, le paiement, l'ERP (Enterprise resource planning) ,…etc.

Comme toute solution, le mode SaaS offre des avantages et des inconvénients.

Avantages

- Le logiciel est maintenu par le prestataire : plus de mise à jour à effectuer ou d'équipe technique à maintenir.
- Le logiciel est évolutif : de nouvelles fonctions sont ajoutées régulièrement.
- La flexibilité est importante : on peut décider de changer d'outil avec des conséquences moindres que dans le cas de logiciels achetés sous licence et maintenus en interne.
- La sécurité est théoriquement plus forte que sur des systèmes internes.

Inconvénients

- Les services hébergés sont régulièrement la cible d'attaques massives de hackers, ces attaques étant particulièrement rentables si elles réussissent (un seul système contient les données de milliers d'utilisateurs).
- La localisation des données et l'absence de contrôle sur ces données, surtout lorsqu'elles sont critiques pour l'entreprise (par exemple les données client d'un CRM), créé une certaine réticence à les faire héberger chez un tiers.

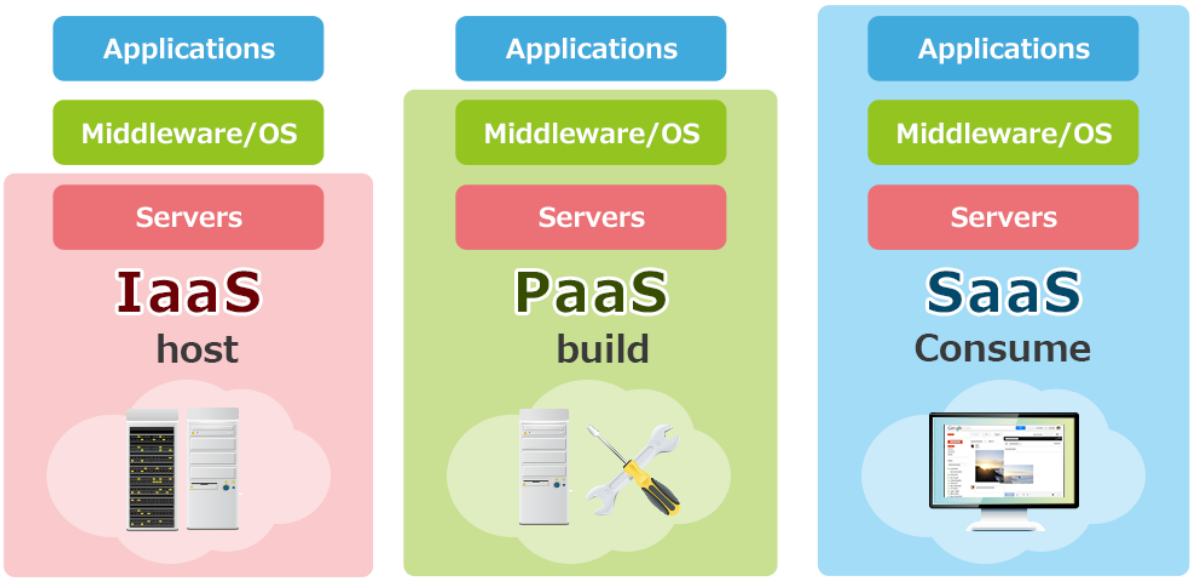

Figure.9 Les services du cloud computing

4. Les formes de déploiement du Cloud Computing

Nous distinguons trois formes de Cloud Computing : Le Cloud publique, historiquement apparu le premier (2000) , le Cloud privé et le Cloud hybride qui est en fait la combinaison des deux premiers.

Le Cloud publique

Le principe est d'héberger des applications, en général des applications Web, sur un environnement partagé avec un nombre illimité d'utilisateurs. La mise en place de ce type de Cloud est gérée par des entreprises tierces (exemple Amazon, Google, etc.) et il est accessible selon le modèle pay-as-you-go (payer selon la consommation) . Les fournisseurs du Cloud publique les plus connus sont Google et Amazon.

Ce modèle :

- Demande de lourds investissements pour le fournisseur de services
- Offre un maximum de flexibilité
- N'est pas sécurisé

Le Cloud privé

C'est un environnement déployé au sein d'une entreprise. Ainsi, elle doit gérer toute seule son infrastructure. Dans ce cas, implémenter un Cloud privé signifie transformer l'infrastructure interne en utilisant des technologies telles que la virtualisation pour enfin délivrer, plus simplement et plus rapidement, des services à la demande. L'avantage de ce type de Cloud par rapport au Cloud publique réside dans l'aspect de la sécurité et la protection des données. En effet, l'ensemble du matériel est conservé au sein de votre propre emplacement. De ce fait, les ressources sont détenues et contrôlées par votre propre département informatique. Eucalyptus, OpenNebula et OpenStack sont des exemples de solution pour la mise en place du Cloud privé.

Ce modèle est :

- Cher pour le client
- Dédié et sécurisé
- Moins flexible comparé au Cloud public.

Le Cloud hybride

En général, on entend par Cloud hybride la cohabitation et la communication entre un Cloud privé et un Cloud publique dans une organisation partageant des données et des applications (Par exemple, un Cloud dédié pour les données et un autre pour les applications). Ce modèle :

- Permet d'allier les avantages des deux modèles de déploiement
- Permet la gestion de deux Clouds qui peut s'avérer plus contraignant.

Le Cloud Communautaire

Le Cloud Communautaire, est un Cloud utilisé par plusieurs entités ou organisations, qui sont animées de besoins communs. Il peut être employé pour des applications génériques, avec des particularités ajustées aux contraintes du groupe.

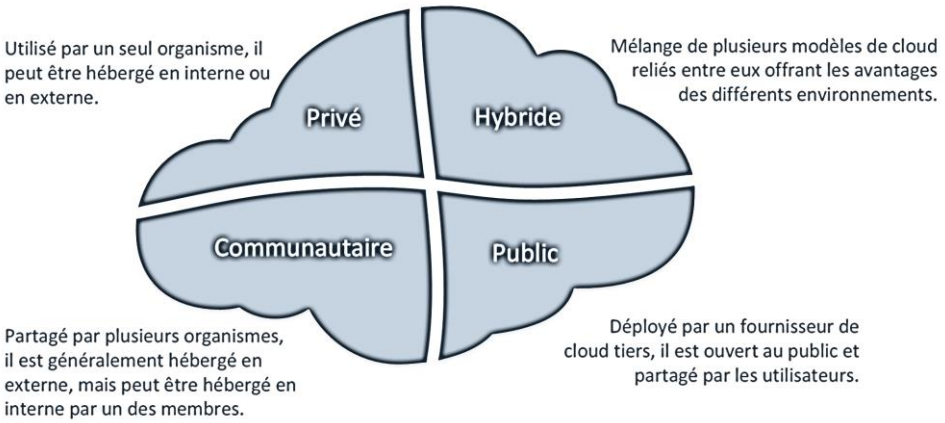

Figure.10 Les modèles de déploiement d'un cloud

5. Conclusion

Le Cloud Computing permet de tester le business plan rapidement, à coûts réduits et avec facilité. Il permet la résolution des problèmes de gestion informatique simplement sans avoir à s'engager à long terme. Il réduit le temps de recherche pour les développeurs sur le paramétrage des applications. Enfin, Il n'y a plus besoin des locaux pour installer et élargir les infrastructures informatiques. Néanmoins, plusieurs questions restent à poser : es ce que mes données sont-elles sures dans le Cloud?, où sont stockées mes données ?, Qui va avoir accès à mes données ?, aurais-je accès à mes données à n'importe quel moment ?,q ue deviendrons mes données s'il y a interruption du service ?.

Chapitre 4 Cloud computing : les solutions techniques

1. Introduction

Le Cloud Computing représente un nouveau défi dans le monde informatique. Plusieurs solutions sont proposées : des solutions propriétaires, des solutions open sources et des solutions offertes par des vendors (Google, Amazon, Microsoft,…etc.). Dans ce chapitre, nous allons présenter des exemples des solutions Cloud existantes ainsi que les services proposées aux sociétés.

2. Google Cloud Platform

L'offre de Google Cloud s'articule en 3 axes :

- Le 'compute' : Il regroupe App Engine et Compute Engine. Il permet de faire tourner votre code.
- Le 'storage' : Il regroupe des bases de données comme DataStore, Cloud SQL et Cloud Storage pour stocker les données et les fichiers.
- Les services. Développés par Google, ils apportent de nouvelles fonctionnalités sans ajouter de développement : ce sont des services "clefs en main". Cela va de l'exposition d'une API REST facilitée via Cloud Endpoint à la gestion de DNS avec Cloud DNS

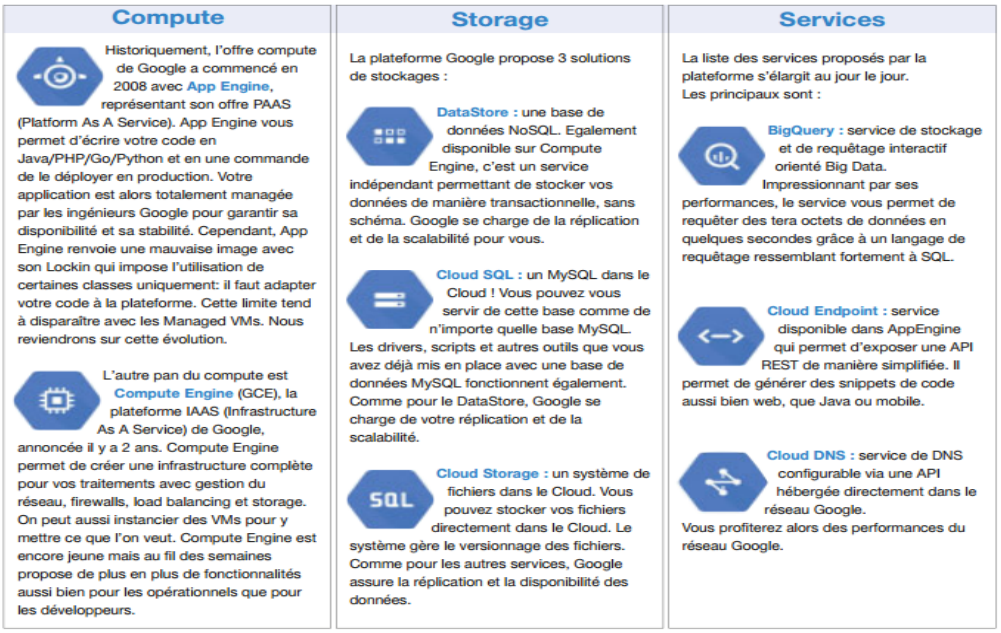

Figure.11 Google Cloud Platform

3. Amazon Web Services

La solution AWS de la société Amazon fournit une plate-forme de cloud computing facile à utiliser, évolutive, rentable et flexible qui convient aux applications de recherche, à des fins de formation, à un usage personnel ainsi qu'aux entreprises de toutes tailles. On peut accéder facilement aux services de cloud AWS via Internet. Le modèle de cloud computing d'AWS permet, en outre, de payer pour les services à la demande et d'utiliser autant (ou aussi peu) de ressources que nécessaire à chaque instant. Ainsi, on peut remplacer les dépenses de capital initiales liées à l'infrastructure par des coûts variables faibles, en adéquation avec l'évolution de nos besoins.

L'offre "Amazon Web Services" est principalement constituée des services suivants :

- EC2 « Elastic Compute Cloud » : création des MV
- Load Balancing/Elastic IP / Auto Scaling
- EBS « Elastic Block Store »
- S3 « Simple Storage Service »
- Une base de données : ***SimpleDB***
- Une base de données relationnelles MySQL : ***Amazon Relational Database Service (RDS)***
- Un Middleware Orienté Messages : ***Simple Queue Service (SQS)***
- Solution pour construire des cloud Privés virtuels : ***Virtual Private Cloud (VPC)***
- Solution pour gérer un réseau de Cache de contenu (Content Delivery Network) : ***Cloud Front***

4. Microsoft Azure

Au niveau le plus élevé, Azure peut être vu comme une simple plateforme applicative, qui va permettre d'héberger des applications dans un environnement Windows hébergé par Microsoft. Cette plateforme est installée sur plusieurs DataCenters dans le monde, permettant de placer géographiquement les serveurs au plus proche des utilisateurs finaux. Ceci étant dit, il est intéressant d'avoir une vue plus fine de ce qui compose la plateforme en elle-même. Lorsque l'on parle d'Azure, on englobe généralement les éléments suivants :

- Windows Azure : Windows Azure est la plateforme en elle-même. On pourrait faire un parallèle entre Windows Azure et un ensemble de serveurs Windows Server
- SQL Azure : SQL Azure va être le composant de gestion de données de la plateforme. De la même manière que Windows Azure serait Windows Server, on pourrait voir SQL Azure comme SQL Server
- AppFabric : AppFabric est la partie de la plateforme qui va gérer les communications entre les différents éléments internes à Azure et la sécurité. Il fait office de composant de middleware (ou en français, de bus logiciel) pour la plateforme.

  Windows Azure peut à son tour être vu comme un agrégat de trois fonctionnalités comme illustré dans la figure. 12

Figure.12 : Composants de Windows Azure

5. Les solutions libres

5.1 Eucalyptus

Issue d'un projet de recherche de l'université de Californie, cette plate-forme cloud open source est certainement la plus connue, car intégrée dans Ubuntu Server et Debian. Ecrite en C, Java et Python, elle permet de créer des clouds Iaas (Infrastructure as a service) de type privé ou hybride, supporte des machines virtuelles Linux ainsi que les hyperviseurs Xen et KVM. Par ailleurs, elle est compatible avec EC2 d'Amazon. Il existe également une version propriétaire commercialisée par la société Eucalyptus Systems. Il apporte des fonctionnalités supplémentaires comme le support de VMware, celui des machines virtuelles Windows et l'intégration SAN.

Une configuration de cloud fondée sur Eucalyptus se compose de cinq types de composants principaux.

- **Cloud Controller :** c'est l'unique point d'entrée (Front end) pour tous les utilisateurs et les administrateurs d'Eucalyptus. Il est responsable de la gestion de tout le système. Surveiller la disponibilité des ressources sur les différentes composantes de l'infrastructure du cloud.

- **Node Controler :** Le rôle du node est d'héberger 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. Le contrôleur de node fonctionne sur chaque node et il est chargé de vérifier le cycle de vie des instances en cours d'exécution sur le node.

- **Cluster Controller :** Ce contrôleur sert à déployer et gérer les différents contrôleurs de node. Il sert également à gérer la mise en place du réseau entre les instances des différents node. C'est lui qui communique l'ensemble des informations au contrôleur du cloud. Il reçoit les requêtes de déploiement des instances, décide sur quel contrôleur de node les instances seront déployé aussi il contrôle le réseau virtuel entre les instances.

- **Walrus:**

Il assure 3 fonctions principales :

Le stockage des images de machines virtuelles.

Le stockage des images prises en fonctionnement à un instant précis.

Le stockage des fichiers et des services

- **Storage Controller:** ce composant fonctionne avec le composant Walrus et permet de stocker les images des machines virtuelles et les données des utilisateurs.

5.2 OpenStack

Créé en juillet 2010 par la NASA et l'hébergeur américain Rackspace, OpenStack est une offre d'IaaS 100% open-source encore en développement qui a livré son code source récemment et qui permet aux sociétés de développer leurs propres solutions d'infrastructure du Cloud Computing.

Plus que trente fournisseurs soutiennent ce projet tels que : AMD, Intel, Dell et Citrix. Il comprend le logiciel OpenStack Compute pour la création automatique et la gestion de grands groupes de serveurs privés virtuels et le logiciel OpenStack Stockage pour optimiser la gestion de stockage, répliquer le contenu sur différents serveurs et le mettre à disposition pour une utilisation massive des données.

OpenStack s'organise autour de trois composants et des API qui leur permettent de communiquer :

- ***OpenStack Nova*** : Il fournit les fonctionnalités de gestion du cycle de vie des VM (via le sous composant nova-compute), du réseau (via nova-network) et des authentifications.
- ***OpenStack Swift*** : permet de créer un service de stockage dans une architecture de cloud computing.
- ***OpenStack Imaging Service*** : OpenStack Imaging Service est un système de récupération et de recherche d'images de machines virtuelles.

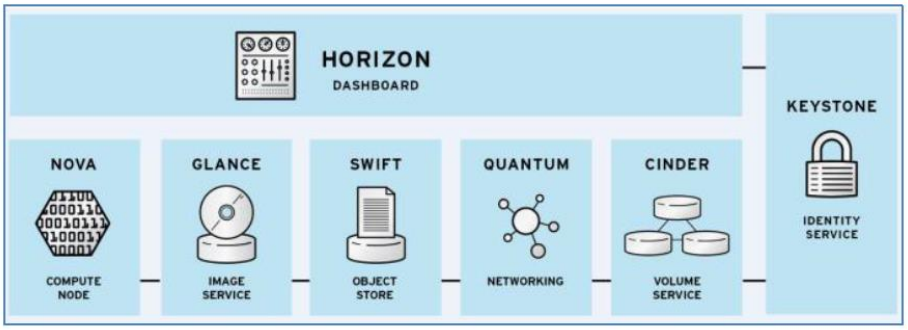

Figure. 13 Architecture de la solution Openstack

5.3 OpenNebula

OpenNebula voit le jour en 2005 à l'université Complutense de Madrid dans le cadre du projet européen open source RESERVOIR. Son objectif dans le cadre de ce projet est l'administration des IaaS virtualisés. Autrement dit, il fournit des services permettant de déployer et d'exécuter dans un environnement matériel virtualisé des

VM. Notons qu'une version commerciale d'OpenNebula (OpenNebulaPro) est disponible depuis 2010.

OpenNebula est capable de prendre en compte simultanément dans l'IaaS des hyperviseurs Xen, kvm et VMware. Il organise l'IaaS sous forme de clusters et de VLAN (réseaux virtuels). Un cluster contient un ensemble de machines physiques tandis qu'un VLAN est défini pour un ensemble de VM. Lors de la création d'une VM, le client choisit la machine et le VLAN dans lequel il souhaite l'exécuter. Notons que dans l'esprit du cloud, il ne revient pas au client de choisir la machine sur laquelle il souhaite exécuter sa VM.

Toutes les opérations d'administration sont coordonnées à partir d'une unique machine de l'IaaS appelée Frontend.

Les composants d'Open Nebula peuvent être divisés en trois couches :

- ***Tools*** : c'est l'ensemble des outils de gestion pour OpenNebula ;
- ***Core***: il se compose d'un ensemble de composants pour contrôler les machines virtuelles, le stockage et le réseau virtuel ;
- ***Drivers*** : l'interaction entre OpenNebula et l'infrastructure de Cloud est effectuée par des pilotes spécifiques qui sont les drivers.

Les machines Front end et Node sont reliés entre eux à travers un réseau privé.

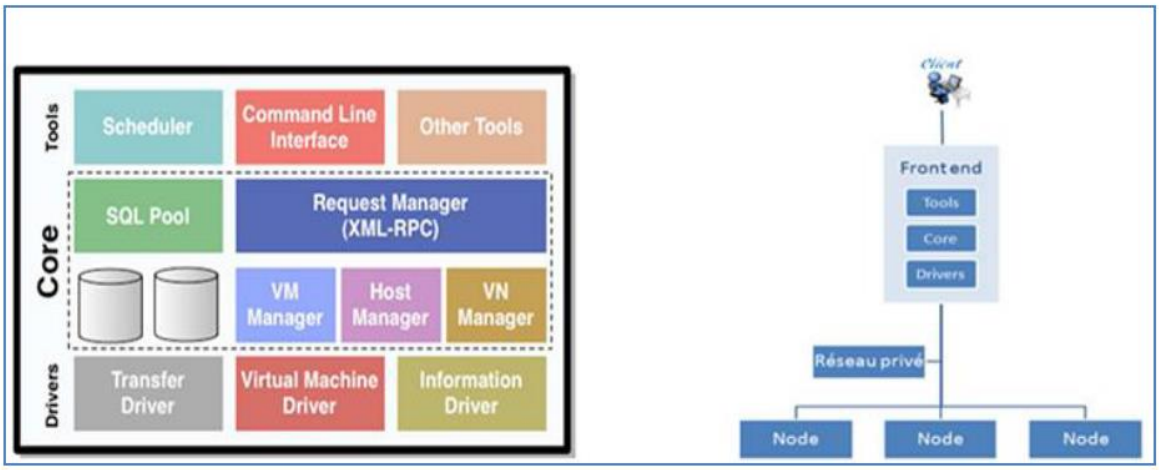

Figure.14 Composants de la solution OpenNebula

Chapitre 5 Environnement Matériel et Logiciel d'un Cloud

1. Architecture physique

L'infrastructure physique 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.

La couche IaaS du Cloud Computing comprend trois parties essentielles :

- La partie réseau qui regroupe des routeurs, des switchs et des firewalls.
- La partie stockage SAN (Storage Area Network) qui comprend principalement des baies.
- La partie compute qui est constituée des châssis regroupant des serveurs blades.

a. Partie de Stockage

Le **SAN** est une technologie de stockage en réseau qui fournit l'espace disque rapide et fiable. 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 FC. Cette technologie est basée sur le protocole Fibre Channel, qui autorise le transfert de données entre périphériques sans surcharger les serveurs.

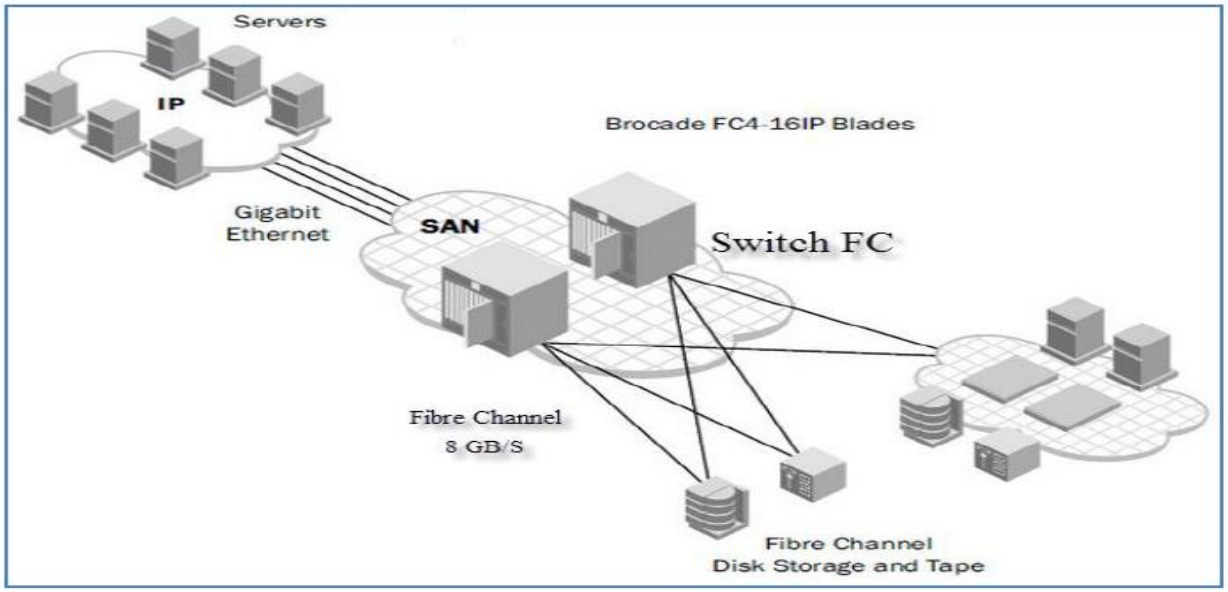

Figure15. Technologie de stockage SAN

Une **baie de stockage** est un équipement de sauvegarde de données informatiques qui comporte principalement un ensemble de disques, permettent d'emmagasiner et de gérer de grandes quantités de données généralement à travers un réseau de stockage dite SAN.

Les baies de stockage utilisent différentes techniques d'agrégat de disques, nommées RAID qui gèrent la cohérence et la répartition des données sur plusieurs disques durs. Les disques qui existent sur le marché sont : FC, SATA, SAS mais le meilleur c'est le FC. Les baies utilisent aussi des protocoles de stockage comme iSCSI ou FC. Mais ce dernier est le plus performant et il peut aller jusqu'à 10GB/s.

b. **Partie de Calcul (compute)**

Un serveur lame ou 'blade' est un serveur de la taille d'une carte d'extension PCI, intégrant processeur, mémoire vive, interface réseau et disque dur, dont la compacité simplifie la gestion de l'espace, économise la consommation d'énergie, et autorise l'installation d'un grand nombre de serveurs. Tenant sur une simple carte PCI, il nous permet de ranger dans un seul châssis des dizaines de serveurs. Chaque lame est un serveur à part entière, souvent dédié à une seule application.

En effet, chaque lame à six connectiques Réseau (sur le châssis) :

- Une carte pour l'administration des blades : une path sur ETH 1 et l'autre sur ETH 2 de châssis.
- Une carte pour le LAN : une path sur ETH 3 et l'autre sur ETH 4.
- Une Carte pour le stockage : une path sur FC 1 et l'autre sur FC 2.

Le châssis est un équipement qui héberge un ensemble de serveurs lames et fournit une source d'alimentation électrique unique pour ces serveurs en mutualisant plusieurs unités d'alimentations électriques, assurant ainsi une redondance et permettant une tolérance aux pannes. Les connexions réseau sont incluses dans le châssis. Cela permet de connecter un serveur lame à différents supports physiques (paire torsadée ou fibre optique) et de mettre en place des configurations avancées (agrégation de ports). Chaque châssis peut contenir un certain nombre de switchs internes. Mais généralement il intègre six switchs : quatre switchs ETH et deux switchs FC.

Figure .16 un exemple de châssis pour serveurs lames : HPE blade system

c. **Partie réseau**

L'architecture réseau d'un cloud privé est composé généralement de:

- Un serveur qui a comme rôle la gestion du nuage (dashboard)
- Des serveurs (contenant des MV) reliés entre eux par des switchs
- Des postes pour les utilisateurs (edge points)
- Une infrastructure destinée aux communications entre les machines
- Des équipements pour la sécurité et le contrôle d'accès (Firewalls)
- Des équipements pour le maintien et la climatisation des équipements.

2. Environnement logiciel

Une fois choisie, une plate-forme de cloud privé ne peut pas être remplacée facilement par une autre. L'expertise de sa mise en place, son administration, les développements engendrés nécessitent un investissement conséquent. Le choix d'une solution de mise en œuvre et d'exploitation d'un cloud reste une tâche difficile pour le décideur en réseaux, néanmoins la tendance vers les solutions libres reste toujours le garant de la sécurité et de l'indépendance.

OpenNebula , Nimbus, Cloudstack, Stratuslab, Eucalyptus sont d'autres solutions libres de cloud privé. Elles sont toutes éclipsées par OpenStack. En quelques années, OpenStack s'est imposé comme un standard de fait. Dans la suite

on va représenter succinctement les différentes fonctionnalités offertes par ce logiciel.

OpenStack est un logiciel modulaire où chaque module assure une fonctionnalité nécessaire (voire complémentaire) pour rendre la gestion du cloud aussi facile que possible. Les différents modules (liste non exhaustive) sont les suivants.

a. Horizon (interface web de management)

Horizon est l'application web faisant office de panneau de contrôle d'OpenStack. Les utilisateurs se connectent à cette interface pour gérer leurs machines virtuelles, leurs images, leurs réseaux, etc. Mais Horizon est plus qu'une simple interface, c'est aussi un framework, que nous pouvons utiliser pour modifier facilement l'apparence du site et ses possibilités. Jusqu'à présent, la grande majorité des utilisateurs se basent sur Horizon pour gérer leurs infrastructures virtuelles. Cette interface, même si elle est moderne et ergonomique, ne dispose pas de toutes les possibilités qu'offrent les API d'OpenStack. Par exemple, elle ne permet pas de partager une image avec des utilisateurs particuliers, ni de connecter une machine virtuelle à plusieurs réseaux. A l'inverse, cette interface fournit des possibilités qui vont bien au-delà des besoins d'une grande partie des utilisateurs, comme celles relatives aux réseaux virtuels.

b. Keystone (gestion des identités)

Keystone est la pierre angulaire de la sécurité. Il réalise ou sous-traite l'authentification et assure les autorisations d'accès.

Il se base principalement sur trois « entités » :

– l'utilisateur

– le rôle

– le projet

On peut simplifier la relation entre ces trois entités en une phrase : un utilisateur a un certain rôle dans un certain projet.

Une ressource virtuelle (instance de machine, réseau, etc.) ne peut pas être associée directement à un utilisateur. Elle est associée soit à un projet, soit à un couple utilisateur-projet.

Des jetons à usage limité dans le temps, générés par Keystone lors de l'authentification, permettent d'identifier les utilisateurs et de vérifier leurs droits

d'accès. Ces droits d'accès sont configurés dans des fichiers qui associent chaque action à des rôles. Par exemple, on peut définir que seul le rôle *admin* dispose du droit de créer un nouveau projet.

Le composant keystone inclut un backend LDAP, ce qui rend la configuration relativement simple.

c. Glance (gestion des images)

Glance est le composant qui gère les images de machines virtuelles. Autrement dit les modèles qui servent à créer des instances de machines virtuelles. La configuration de Glance est triviale et n'a posé aucun problème particulier. Chaque utilisateur peut, selon un certain quota, transférer sur le cloud des images de machines virtuelles, qui sont alors stockées sur la baie de disques et éventuellement partagées avec d'autres utilisateurs. Des images adaptées à une utilisation avec OpenStack de la plupart des systèmes d'exploitation et distributions Linux sont fournies par les éditeurs. Mais il est également possible pour l'utilisateur de préparer une image à une utilisation dans le cloud avec des outils comme cloud-init.

d. Nova (gestion des instances)

Nova constitue le coeur du cloud OpenStack. Glance lui fournit des images qu'il « transforme » en instances de machines virtuelles exécutées sur les différents noeuds de calculs.

La plupart des hyperviseurs actuels peuvent être utilisés avec Nova. KVM, est le plus fiable, relativement performant et offre une bonne isolation des instances, et surtout, il fournit le plus grand support de la part de la communauté OpenStack.

Les agrégats OpenStack permettent d'utiliser, côte à côte, des hyperviseurs différents répartis sur des groupes de noeuds de calculs. Ces agrégats permettent d'ajouter, en plus des clusters de noeuds KVM , des clusters VMware ou Xen utilisés dans d'autres infrastructures OpenStack.

e. Neutron (gestion des réseaux)

Deux composants, au choix, peuvent être utilisés pour fournir un accès réseau aux machines virtuelles.

— Nova-network est le composant historique. Ses possibilités sont limitées.

— Neutron est la solution d'avenir. Il fournit aux utilisateurs la possibilité de gérer ses propres ressources virtuelles (réseaux, routeurs, pare-feu, etc.)

Neutron offre deux alternatives à ce protocole : GRE et VXLAN.

f. Autres outils

Au fur et à mesure l'architecture d'OpenStack devient de plus en plus large en intégrant d'autres modules comme ceilometer (facturation), Quotas,…etc. Il faut mentionner aussi que pour un fonctionnement en production, OpenStack a besoin de nombreux outils tiers. Entre autres, on peut citer :

— MySQL + Galera pour la création et la réplication des bases de données en temps réel

— OpenLDAP pour le cache des comptes utilisateurs et le stockage des projets et des rôles

— Openvswitch pour la gestion des commutateurs virtuels

— Nagios et Cacti pour la supervision

— …etc.