Virtualisation de serveurs
Solutions Open Source

Alain Devarieux
TSRITE2009 FOAD

Table des matières

1.Les principes de la virtualisation....3
1.1.Partage d'un serveur....3
1.2.Objectif de la virtualisation....4
1.3.Bref historique....4
1.4.Hyperviseur....5
1.5. Performances et rendement....6
1.6.Sécurité....6
1.7.Administration....7
1.8.Contrôle des ressources....7
2.État de l'art....9
2.1.L'isolation....10
2.2.Virtualisation complète....11
3.Les Principales Solutions....12
3.1. OpenVZ....12
3.2.Xen....12
4.Domaines d'application....14
4.1.Hébergement VDS....14
4.2.Plateforme de validation de développement....14
4.3.Haute disponibilité....14
a)Répartition de charge....14
b)Reprise automatique....16
5.Le stockage....17
5.1.Différents besoins....17
5.2.Stockage en réseau....17
6.Conclusion....19

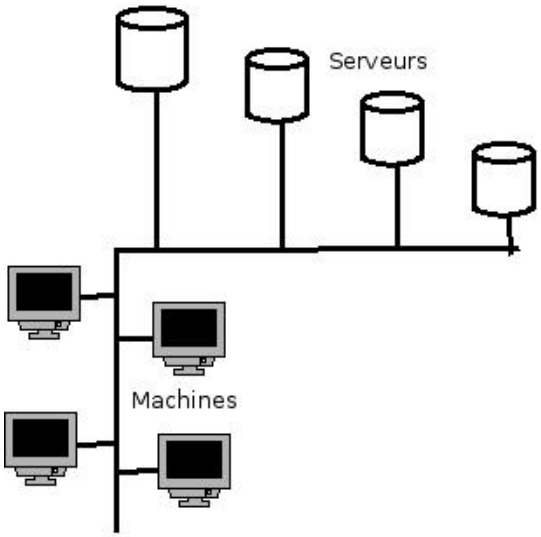
1. Les principes de la virtualisation

1.1. Partage d'un serveur

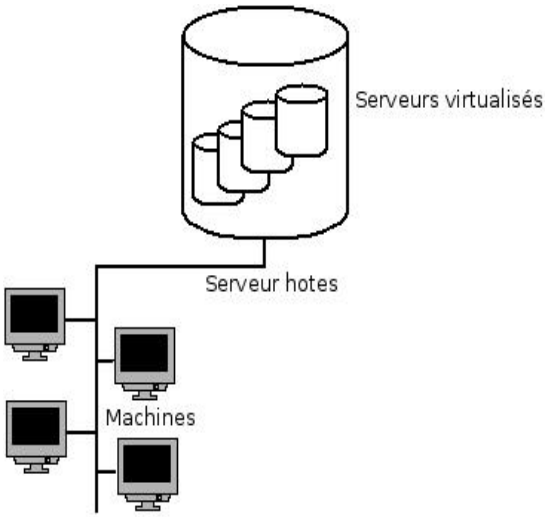
La virtualisation de serveurs est un ensemble de techniques et d'outils permettant de faire tourner plusieurs systèmes d'exploitation sur un même serveur physique. Le principe de la virtualisation est donc un principe de partage : les différents systèmes d'exploitation se partagent les ressources du système.

De manière opérationnelle, pour être utile, la virtualisation doit respecter 2 choses :

- **La cloisonnement** : chaque système d'exploitation est indépendant et ne doit pas interférer avec les autres.
- **La transparence** : le fait de fonctionner en mode virtualiser ne doit rien changer au fonctionnement du système d'exploitation ni à celle des applications.



Architecture traditionnelle



Architecture virtualisée

1.2. Objectif de la virtualisation

La premier objectif de la virtualisation est économique.

Partager les ressources physique d'un serveur entre différents serveurs virtuels permet de ne pas acheter plusieurs serveurs physiques. Le constat qu'on peut en tirer est que les serveurs physiques sont souvent sous exploités. On estime ainsi que dans un data center privé, le taux d'utilisation moyen est de 10%. Celui ci passe à 35% sur une architecture virtualisée. De plus, les serveurs commercialisés correspondent à des normes contemporaines, notamment en terme de puissance de calcul. On peut ajuster à l'achat la quantité de mémoire souhaiter, mais on peut ne nécessité que d'1/10ème de processeur. Il faut donc acheter un processeur entier avec un serveur entier qui sera sous utilisé.

Éviter de multiplier les serveurs physiques permet de faire des économie en *coût d'acquisition*, mais aussi en *coût de possession*, tant au niveau de l'hébergement (rack, électricité, câblage, interface réseau) que de l'exploitation.

La virtualisation apporte aussi indirectement d'autres avantages. Elle permet par exemple de déplacer une VM d'un hôte à un autre de manière très simple, même sur du matériel différents de celui d'origine. On peut donc agencer et répartir les serveurs virtuels sur les serveurs physiques ce qui est une véritable révolution dans l'administration de parcs de serveurs.

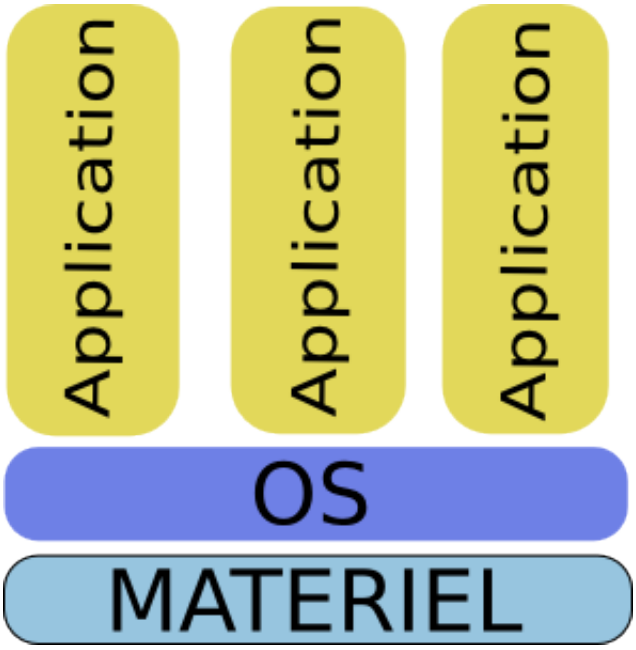
1.3. Bref historique

IBM fut le pionnier de cette technologie dans les années 60 avec son hyperviseur CM/CMS puis VM/CMS dans les années 70. Dans la seconde moitié des années 90, VMware développe et popularise le produit du même nom sur les machines x86. Dans le monde des logiciels libre, il faut citer Qemu qui en fut le précurseur, suivi par Xen et KVM..

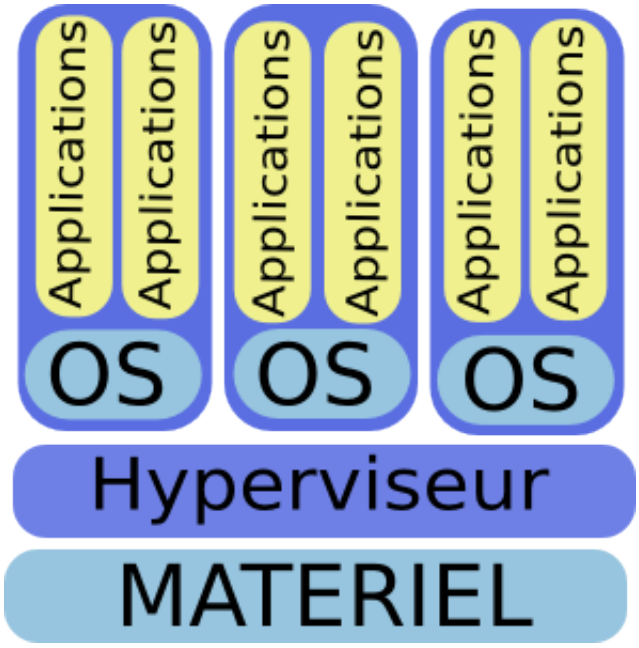
Pour répondre au défis de la virtualisation en terme de performances, AMD et Intel ont implémenter dans leur processeurs des instructions spécifique améliorant les possibilité de virtualisation avec un rendement proche de 100%.

1.4. Hyperviseur

L'hyperviseur est la couche logicielle qui s'insère entre le matériel et les différents systèmes d'exploitation . Voici un schéma de base d'un serveur, en distinguant le matériel, le système d'exploitation et les applications.

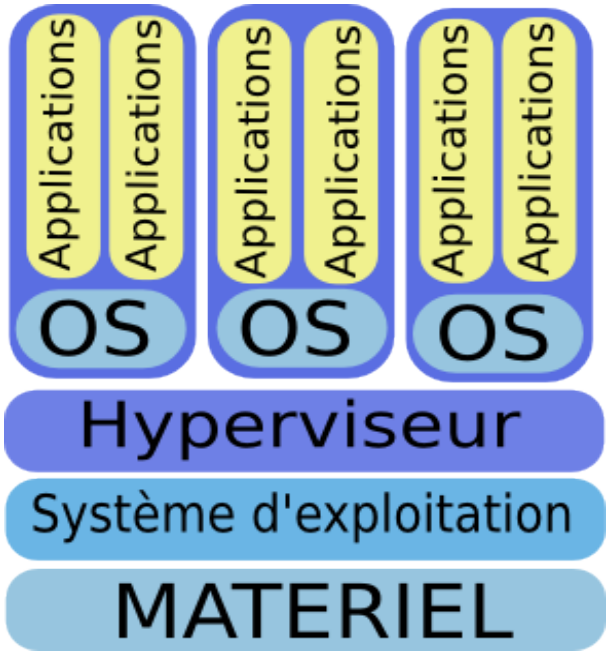


L'hyperviseur vient s'insérer entre le matériel et les systèmes d'exploitation.



Hyperviseur de type1

Il existe aussi des hyperviseurs d'un autre type. Ceux ci nécessite une couche logiciel entre eux même et le matériel, à savoir, un système d'exploitation complet



Hyperviseur de type 2

1.5. Performances et rendement

En observant les fonctionnement d'un hyperviseur, on se rend bien compte que chaque environnement virtuel dispose de ressources plus limitées que s'il disposait d'un serveur physique dédié. La vrai question est : « *La somme des ressources allouées aux différents environnements virtuels est-elle égal au ressources physiques disponibles?* » ou « *Quel est le surcout de la virtualisation?* ».

Les solutions de virtualisation actuelles , appuyées sur des processeurs disposant d'instructions spécifiques permettent un surcout en performance qui est aujourd'hui négligeable. C'est à dire que le rendement est pratiquement égal à 1.

1.6. Sécurité

La sécurité du serveur physique est critique. Un accès console sur ce serveur peut compromettre l'hyperviseur et les machines virtuelles. Il faut donc avoir un très haut niveau de sécurité et distinguer l'administration du serveur physique et de l'hyperviseur de l'administration des environnements virtualisés.

A l'inverse, un accès root sur un des environnements virtualisés ne livre aucun accès au serveur physique et à l'hyperviseur, ni aux autre machines virtuelles.

La bonne pratique consiste à placer les serveurs physiques dans des réseaux différents de ceux des environnements virtuelles, ce qui les rend inaccessibles depuis l'extérieur.

1.7. Administration

Si la virtualisation est transparente pour les utilisateurs, pour les machines clientes et les applications, elle ne l'est évidemment pas pour l'administrateur qui en a la charge. Pour un administrateur système de bon niveau, maitriser une solution de virtualisation complète demandera quelques jours de formation et quelques semaines de pratique.

Les solutions open source de virtualisation ne fournissent pas d'outils graphique aussi abouti que leur concurrents propriétaire. Mais même si la maitrise des outils administratifs en ligne de commande nécessite un certains niveau de formation, ils permettent une maitrise plus importante et une plus grande souplesse d'utilisation.

1.8. Contrôle des ressources

Le contrôle du partage des ressources est une des plus grande problématique de la virtualisation. Si par exemple 10 serveurs virtuels se partagent un serveur physique et qu'un de ces serveurs connait un pic d'utilisation, on peut souhaiter le laisser prendre 90% du temps CPU tant que les autres serveurs virtuels n'en ont pas l'usage. À l'inverse, si un hébergeur de site web a vendu 1/10ème de serveur à un de ses clients, il doit être en mesure de garantir que le client aura toujours son quota quelle que soit la demande des autres clients.

Les différents produits de virtualisation fournissent des outils permettant ce contrôle des ressources. Les 4 ressources que l'on souhaite contrôler sont :

- **CPU** : La plupart des technologies permettent d'attribuer des poids selon l'importance et les besoin d'un serveur virtuel. On peut ainsi privilégier un serveur par rapport à un autre pour lui assurer un minimum de temps CPU.

- **Mémoire** : La mémoire est la ressource la mieux gérée par les hyperviseurs. La mémoire que l'on souhaite attribuer à un serveur virtuelle est souvent réservée à la création de la VM.

- **Stockage** : Les différents produit de virtualisation peuvent tous utiliser différents type de stockages, tel qu'un simple répertoire local, un fichier image, un disque dur entier ou un répertoire logique sur un SAN.

- **Réseau** : C'est la ressource la moins bien gérée par les technologies actuelles de virtualisation. Dans les produits présentés ici, aucune limitation de bande passante réseau n'est possible. En revanche, contrairement aux autres ressources, il est possible de contrôler le réseau en amont, au moyen d'un routeur implémentant des technologies de Qualité de Service.

2. État de l'art.

Il existe différentes techniques de virtualisation. Toutes ne possèdent pas le même niveau d'abstraction. Nous avons, dans l'ordre croissant d'abstraction :

- L'isolation.
- La para virtualisation.
- La virtualisation complète, ou machine virtuelle.
- La partitionnement matériel.

L'isolation consiste à mettre en place sur un même noyau de système d'exploitation, une séparation forte entre différents contextes logiciels. Il s'agit de la technique de virtualisation la plus légère qui existe.

La para virtualisation présente aux systèmes d'exploitations virtualisés une machine générique spécial, avec des interfaces spéciales, intégrées au système invité sous la forme de driver ou de modifications du noyau.

Dans la virtualisation complète, l'hyperviseur intercepte de manière transparente tous les appels que le système d'exploitation peut faire au matériel et supporte donc des systèmes invités non modifiés.

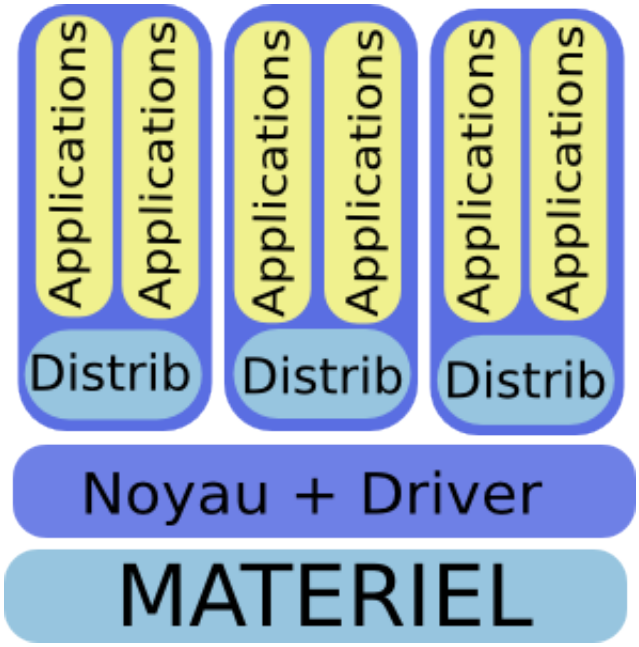
La partitionnement matériel est la technique utilisée sur les gros systèmes. Elle consiste à séparer les ressources matérielles directement sur la carte mère de la machine. Cette technique est surtout répandue dans les serveurs haut de gamme de type SPARC. Elle est assez rare dans le monde x86.

Je présenterai par la suite seulement les 2 techniques les plus utilisées dans le monde x86 : L'isolation et la virtualisation complète.

2.1. L'isolation

Aussi appelé cloisonnement, cette technique se met en œuvre au sein d'un même système d'exploitation. Elle permet de séparer un système en plusieurs contextes. Chacun de ces contextes en géré par un seul OS mais les programmes de chaque contexte ne peuvent communiquer qu'avec les processus et les ressources associés à leur contexte.

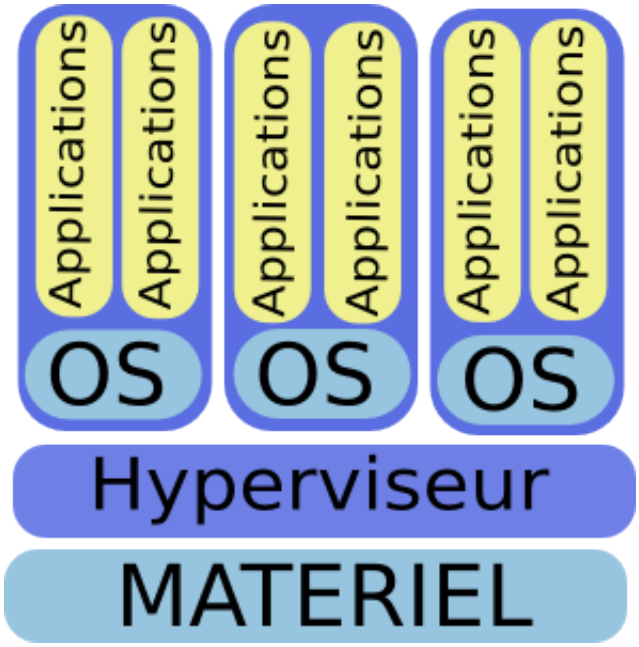
L'isolation est utilisé depuis longtemps sous Unix pour protéger les systèmes via des mécanismes comme chroot qui rend possible l'exécution d'applications dans un environnement qui n'est pas celui du système hôte, mais un mini système qui ne contient que ce dont l'application a besoin. L'espace noyau n'est pas différencié. Ainsi, on peut faire tourner différentes distributions à condition qu'elles partagent le même noyau. L'unicité de ce noyau reste la principal limitation d'une part en terme de robustesse puisqu'un plantage noyau entraine un plantage de tous les environnements et d'autre part dans les utilisations possible puisque ce mécanisme ne peut pas convenir pour une validation d'une nouvelle version du noyau.



2.2. Virtualisation complète

Comme son nom l'indique, la virtualisation complète consiste à simuler un ordinateur complet de façon à ce que le système d'exploitation ne se rende pas compte qu'il est virtualisé. On parle aussi de machines virtuelles, ou VM. On peut donc faire fonctionner plusieurs systèmes d'exploitations différents sur un même serveur physique. Le matériel du serveur physique est rendu abstrait et remplacé, du point de vue des serveurs virtuels par du matériel générique. Ce matériel est soit émulé pour ressembler à un matériel réel, soit para-virtualisé, c'est à dire qu'il nécessite un pilote particulier dans le système invité pour fonctionner.

Les première solutions de virtualisation complète étaient basées sur des émulateurs. Cette émulateur réinterprétait chaque opération demandée par le système virtuel pour les adapter au matériel physique réel, au prix d'une grande perte de performances. Suite à cela, les techniques issues de la para-virtualisation se sont greffées aux hyperviseurs. En effet, la présence de matériels émulés ralentissaient les entrées/sorties de la machine virtuelle, en particulier les accès disques et le trafic réseau. Un pilote de périphérique spécial, nativement capable de dialoguer avec l'hyperviseur permet d'obtenir des performances quasi native pour les E/S. Dans les solutions de virtualisation, cela prend la forme d'un driver s'installant dans l'OS invité. Par exemple les « VMware tools » ou « VirtualBox Guest Additions » pour les plus connus. Ce pilote permet une meilleur interaction entre l'hyperviseur et le système invité, notamment de commander l'extinction de l'OS invité à partir de l'hyperviseur.



3. Les Principales Solutions

3.1. OpenVZ

Une des solutions les plus avancées et matures dans le domaine de l'isolation est OpenVZ. Ce produit se présente sous la forme d'un patch pour le noyau Linux, et d'un ensemble d'outils d'administration. Le patch du noyau permet à un un système GNU/Linux de gérer des contextes virtualisés.

OpenVZ utilise la technologie d'isolation qui ne permet donc d'exécuter que des serveurs virtuels Linux, même si ces serveurs peuvent être des distributions différentes. Cette solution peut être mis en place en quelques secondes sur la majorité des distributions via le système de package.

Le patch noyau de ce projet open-source permet d'y ajouter un ensemble d'outils pour isoler les contextes. Ceci permet de faire tourner plusieurs instances de l'OS sur la même machines. Toutefois, si le noyau se retrouve en défaut, tous les contextes sont impactés.

Sa limitation réside dans le fait que, tous les serveurs utilisant le même noyau, OpenVZ est incapable de faire tourner autre chose que des distributions GNU/Linux. En revanche, cela lui confère un rendement proche des performances natives.

Enfin, OpenVZ permet de mettre en place des quotas d'utilisation d'espace disque ou de mettre des priorités d'accès au CPU et aux disques.

3.2. Xen

Xen quant à lui est une solution de virtualisation complète activement sponsorisée par Citrix, doté d'une interface d'administration avancée et d'un accès au support technique. IBM a également grandement contribué au développement de Xen.

Chaque système s'exécutant sous cet hyperviseur s'appelle un domaine et dispose d'une interface d'accès aux ressources qui permet de régler une limite mémoire, une limite d'utilisation du CPU ainsi qu'une priorité d'utilisation du temps CPU disponible. Ceci permet de donner une priorité plus importante aux serveurs virtuels considérés comme « critiques ».

Un de ces domaines possède un rôle particulier en sein de XEN, c'est le *domaine zero*. C'est le premier OS lancé par l'hyperviseur au démarrage du serveur physique. Ce domaine zero permet de lancer, arreter et prendre le contrôle des domaines virtuels executés. Il est donc important d'y apporter une attention particulière concernant sa sécurité, notamment en l'isolant du réseau.

Un des grand avantage de Xen est sa souplesse d'utilisation. Un grande liberté est offerte concernant le mode de stockage et le configuration du réseau en particulier. De plus, la migration d'un domaine d'un serveur physique à un autre se fait quasiment sans interruption en utilisant un mécanisme de sauvegarde de la RAM.

Le principal reproche qui peut être fait à Xen est le manque d'ergonomie de la version libre, qui ne dispose pas d'interface graphique comme la solution payante. La documentation n'est pas souvent mise à jour et de nombreuses possibilités intéressante sont peu documentées.

4. Domaines d'application

4.1. Hébergement VDS

Traditionnellement, les offres d'hébergement étaient distinguées en 2 catégories : Hébergement dédié et hébergement mutualisé.

Dans un hébergement dédié, le fournisseur met à disposition de son client un ou plusieurs serveurs physiques. Le client a le contrôle totale sur ce serveur et dispose de la totalité des ressources ce celui ci. Dans l'hébergement mutualisé, le fournisseur utilise un même serveur pour plusieurs de ces clients avec différentes solutions de cloisonnement. Les ressources du serveurs sont donc partagées entre plusieurs clients.

Entre ces 2 modes d'hébergement, la virtualisation a permis un mode combinant l'avantage de l'un et de l'autre : Le partage de ressources et l'autonomie. C'est le mode que l'on appelle VDS pour « Virtual Dedicated Server ». Ceci permet donc, en mettant en œuvre une des solutions décrite plus haut, d'allouer un serveur virtuel à chaque client.

4.2. Plateforme de validation de développement

La compatibilité des applications avec les différentes configurations informatique est un enjeu majeur, en particulier pour les progiciels. Pour garantir cette compatibilité, il est nécessaire de tester les produits sur un large ensemble de plateforme. Certains grands éditeurs utilisent pour cela des fermes de validation comportant plusieurs centaines de serveurs.
La virtualisation permet ici d'alléger ces infrastructure de validation, leur coût matériel, mais aussi leur exploitation.

Grâce à la virtualisation complète, on peut tester la compilation des applications sur un grand nombre d'OS, alors qu'une solution d'émulation permet de simuler un processeur différent.

4.3. Haute disponibilité

La virtualisation apporte également son lot d'avantages pour la répartition de charge (load blancing) et le reprise automatique sur incident (failover).

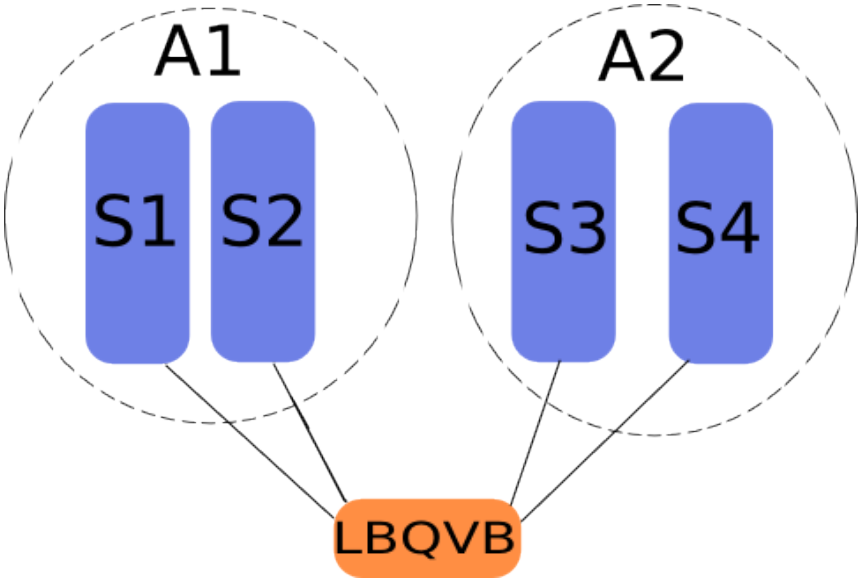
a) Répartition de charge

A la base, la répartition de charge est un moyen d'augmenter la capacité maximale d'une application en l'hébergeant sur plusieurs serveurs qui se partagent les visiteurs. En général, la répartition de charge est gérée en amont par un boitier spécialisé (LBQVB) qui dirige les requêtes des visiteurs sur les différents serveurs. Ces boitiers savent en général détecter la panne d'un serveur et ne plus lui affecter de trafic. Ainsi *Load balancing* et *failover* vont souvent de pair.

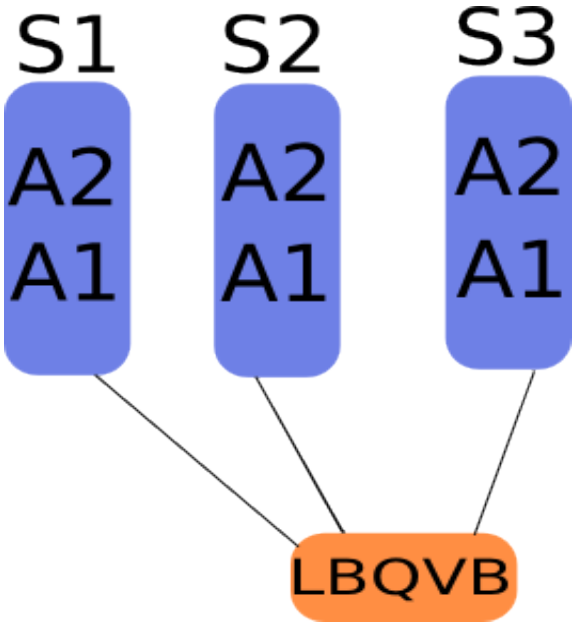
Si l'on est en présence de plusieurs applications de ce type, la virtualisation peut apporter une meilleur mutualisation des moyens.

Supposons que l'on exploite 2 applications critiques A1 et A2. Chacune de ses applications dispose de 2 serveurs physiques entre lesquels le trafic est réparti. Supposons maintenant, ce qui arrive souvent, que ces serveurs ne soient pas utilisés à pleine capacité. Une bonne alternative d'architecture consiste à réunir les 2 applications sur 2 serveurs, chacun partagé en 2 VM, l'une pour A1 l'autre pour A2.

Avant :



Après:



b) Reprise automatique

Un autre usage de la virtualisation dans une optique de haute disponibilité de service peut consister à avoir sur plusieurs serveurs physiques les mêmes environnements virtuels (synchronisés régulièrement).

Les différents serveurs physiques se partagent les différents serveurs virtuels, et si un des serveurs physiques tombe en panne, les machines dont il avait la responsabilité sont relancées sur les autres serveurs. Cela permet d'assurer un temps d'indisponibilité minimum, et une continuité de service malgré des performances amoindries. On peut ainsi travailler plus sereinement à la remise en route du serveur en panne.

5. Le stockage

Tout projet de virtualisation doit, à un moment ou un autre, se poser la question du stockage. Pour cela, les technologies de virtualisations nous offrent plusieurs solutions.

5.1. Différents besoins

Quelque soit la technologie utilisée, une machine virtuelle se compose de 2 choses :

- Des ressources : CPU, RAM, cartes réseau
- Des données : comme un serveur normal, on doit disposer d'un OS, de bibliothèques, d'applications et de leurs données.

Le stockage peut se faire sous différentes formes :

- Sur un fichier
- Sur une partition
- Sur un disque dur
- Sur un serveur distant
- Sur un réseau de stockage

L'utilisation d'un fichier ou d'un disque local est la plus avantageuse en terme de performance et de facilité d'administration Cependant, l'utilisation d'un réseau de stockage permet d'apporter de nouvelles fonctionnalités.

5.2. Stockage en réseau

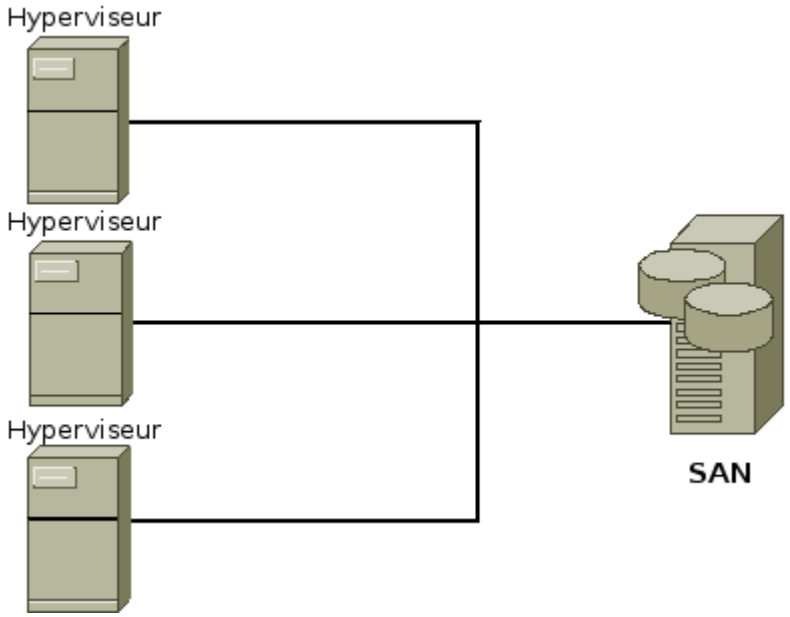
Les pleines capacités des hyperviseurs modernes ne peuvent s'exprimer qu'au travers d'un stockage en réseau, en effet les hyperviseurs sont généralement gérés sous forme de «pools», formant une «force de travail» globale qui se partageront les machines virtuelles à exécuter. Cette vision n'est possible que si le stockage est lui aussi unifié : sans cela chaque hyperviseur ne peut faire tourner que les serveurs virtuels présents sur son disque local, et n'est donc pas interchangeable.

Disposant d'un réseau de stockage, chaque hyperviseur a accès à toute les machines virtuelles, et peut donc exécuter n'importe laquelle, et la transférer sans interruption à un autre hyperviseur en fonction de sa charge. Je vais vous présenter ici la solution la plus avantageuse mais également la plus couteuse : le SAN (Storage Area Network)

Un SAN, ou réseau de stockage, est un réseau sur lequel circulent les données entre un système et son stockage. Cette technique permet de déporter tout le stockage interne d'une machine vers un équipement dédié.

Les SAN sont des équipements dédiés, qui ne travaillent qu'aux plus basses couches du

stockage, la notion de fichier leur est inconnue ; ils travaillent simplement sur des blocs de données et les fournit par le réseau à des serveurs qui eux sauront les utiliser. Cependant, les SAN les plus hauts de gamme sont dotés de capacités avancées, tel que la prise de snapshots ou encore la copie rapide de volumes.



6. Conclusion

Désormais bien installées dans le monde des serveurs, la virtualisation s'attaque de plus en plus aujourd'hui au poste de travail, où elle vise à régler tout les problèmes de déploiement et de maintenance. L'engouement est similaire à ce qu'il à été pour les serveurs, mais la problématique est plus complexe, car les contraintes d'utilisation sont plus fortes : il faut assurer une faible latence, des capacités graphiques à la hauteur du confort d'utilisation d'un poste dédié.

Côté serveur, les technologies de virtualisation ont très rapidement tenu leurs promesses en termes de réduction de coût d'acquisition et de possession des parcs informatiques. En quelques années, elles se sont répandues, et même généralisées. De plus en plus d'administrateurs système préfèrent mettre en place un environnement virtualisé même s'il n'y a dans un premier temps qu'un seul serveur virtuel. D'une part cela permettra ultérieurement de mettre en œuvre un partage des ressources, et d'autre part cela permet de bénéficier des services qui ne sont pas liés à ce partage, par exemple la sauvegarde et reprise de l'environnement.

Les solutions open source apportent exactement le même niveau de service que les solutions commerciales en termes de robustesse, de performances et de pérennité. Il leur reste seulement à combler un petit retard en termes d'ergonomie des interfaces. Mais pour des administrateurs système chevronnés, elles sont le plus souvent privilégiées.