Cours: Virtualisation & Cloud Computing

Virtualisation

Objectifs

- Comprendre le concept de la virtualisation ainsi que les motivations et les avantages offerts par cette technologie
- Connaître les différents défis reliés à la gestion des MVs dans les centres de données à large échelle et les infrastructures distribuées

Plan

1. Virtualisation
    1. Définition et concepts
    2. Types d'hyperviseurs
    3. Avantages de la virtualisation
    4. Migration des machines virtuelles
    5. Accès au réseau
2. VM vs. Container
3. Systèmes de gestion de machines virtuelles (Exemple: SANDPIPER)

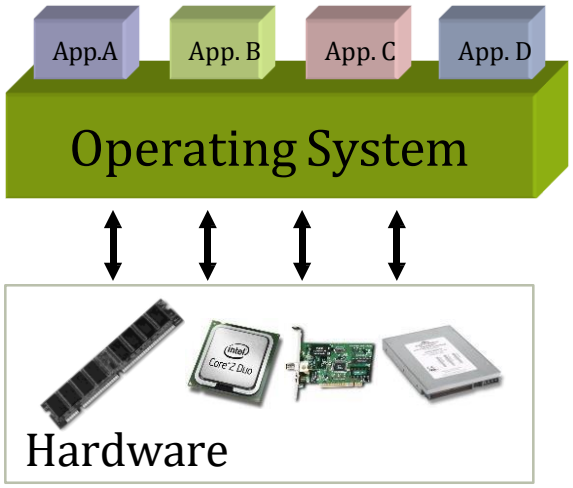
Motivations et défis

- Motivations
  - Des serveurs chers qui sont sous-utilisés (généralement moins de 25%)
  - Consommation d'énergie élevée (un serveur qui ne fait rien consomme 60% de ce qu'il consomme à 100% d'utilisation)
  - ➢ Consolidation de serveurs

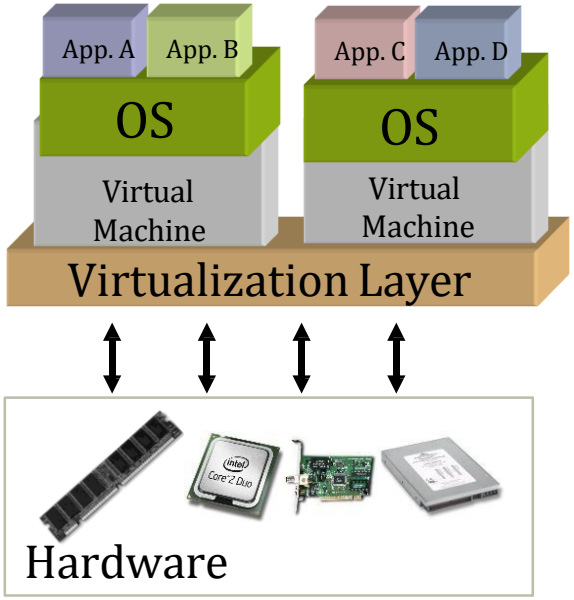
- Défis reliés à la consolidation de serveur
  - Plusieurs applications dans un seul serveur ?
    - Différents systèmes d'exploitation, environnements, versions
    - Différents besoins en termes de ressources
    - Plusieurs risques en termes de sécurité
  - ➢ Besoin de support de plusieurs environnements
  - ➢ Besoin d'isolation entre applications ➔ meilleure sécurité et performance

Qu'est ce que la virtualisation ?

- Idée : découpler le système d'exploitation du hardware
- Une technologie permettant de créer plusieurs environnement simulés appelés "machines virtuelles" (MVs). Chacune agit comme une machine physique avec un système d'exploitation, applications…



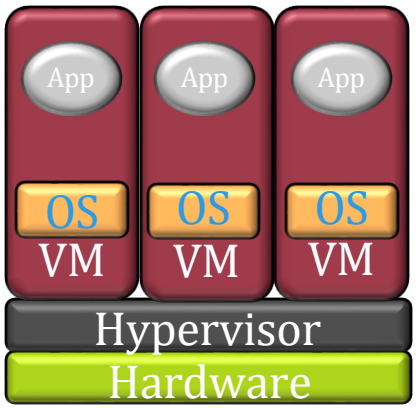
Serveur non virtualisé



Serveur virtualisé

Hyperviseur

- Hyperviseur: (appelé aussi *Virtual Machine Monitor*)
- Un software situé entre les MVs et le hardware
  - Partitionne les ressources physiques, les offrent aux MVs et s'assure de l'isolation entre les MVs
  - Permet aux MVs de partager les ressources de la machine physique
  - Contrôle le flux d'instructions entre les systèmes invités et le *hardware*



- OS_1, OS_2,… OS_n : systèmes d'exploitation invités (hosted)
- Taux de consolidation : Le nombre de MVs placées dans la même machine physique (par ex., 3:1)

OS : Operating System

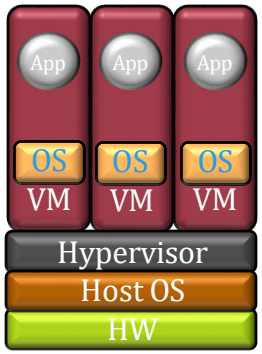
Types d'hyperviseurs

Deux Types d'hyperviseurs :

- Hyperviseur Type 1 (*native* or *bare-metal*) :
  - Installé comme une couche software directement au dessus du hardware
    ➔ contrôle directement le hardware
  - Exemples : Citrix XenServer, VMware ESX/ESXi

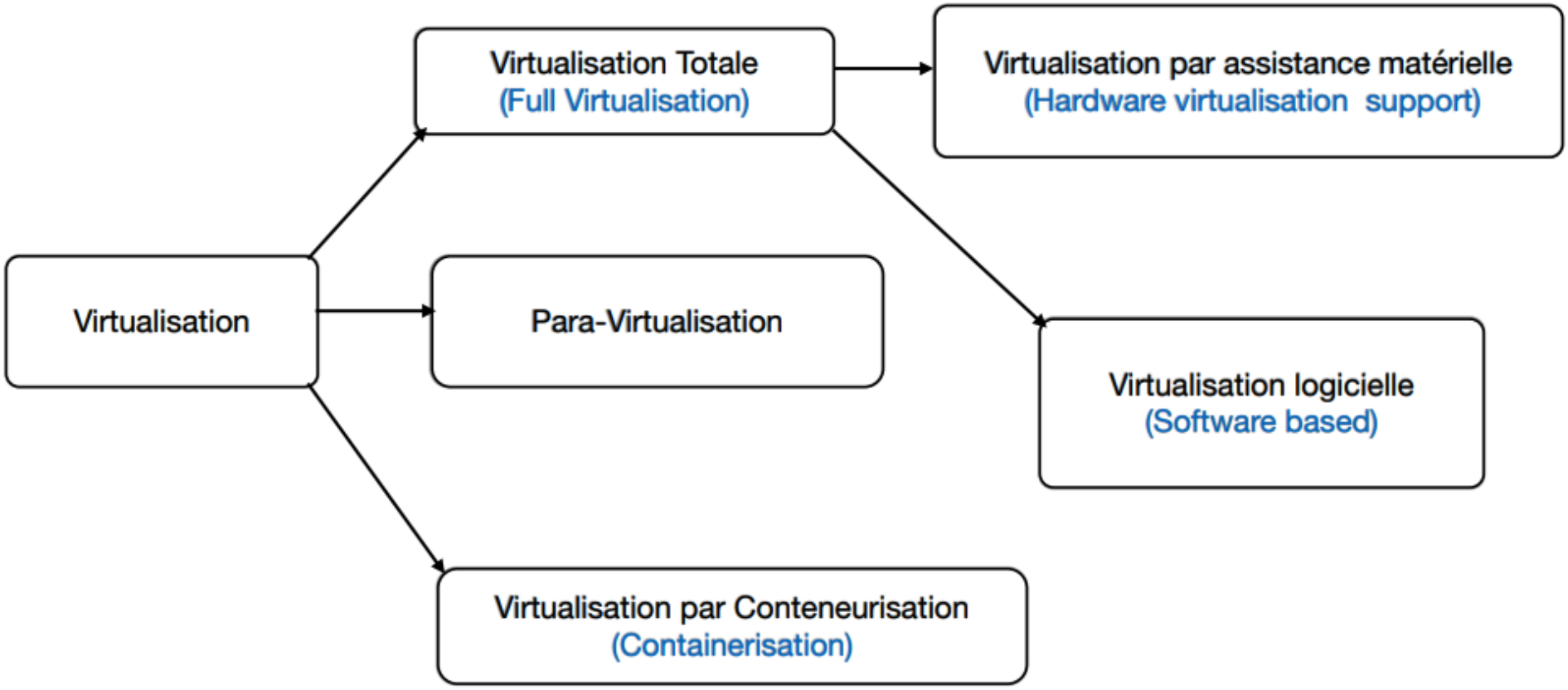
Type 1. Bare Metal

- Hyperviseur Type 2 (*hosted hypervisors*) :
  - Installé au dessus du système d'exploitation
  - Exemples : VMware Workstation, Oracle VM Virtual Box and QEMU
- Type 1 vs. Type 2 : Type 1 offre une meilleure performance + sécurité



Type 2. Non-Bare Metal

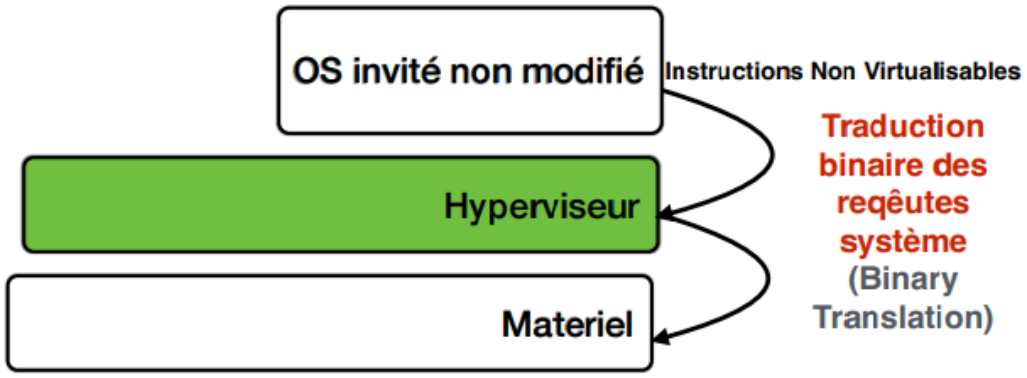
Techniques de Virtualisation



Techniques de Virtualisation

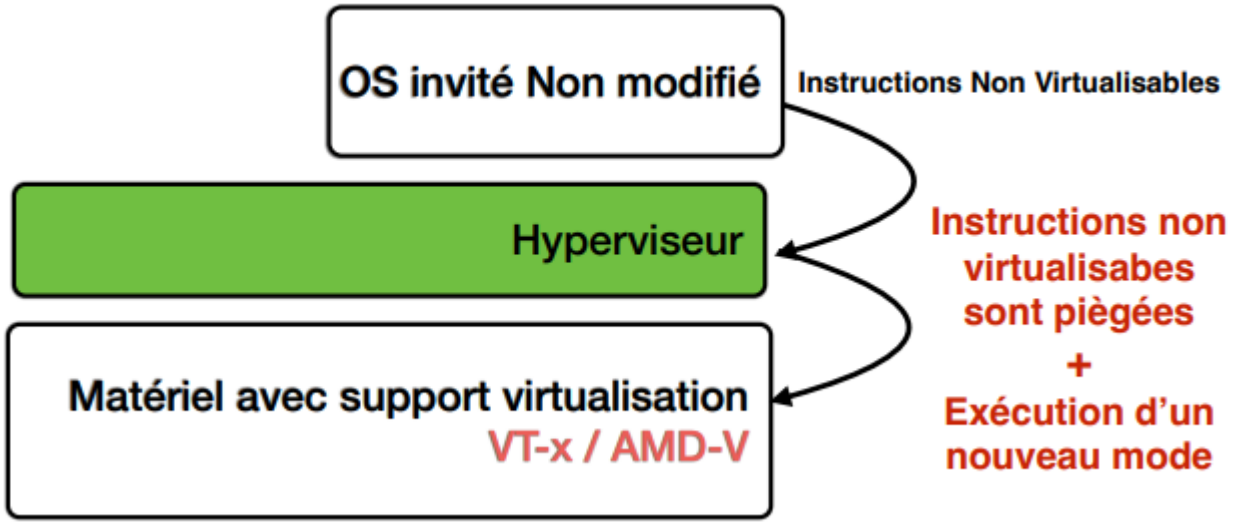
- 1ère Génération : Full virtualization (*binary rewriting*)
  - SE invité non modifié et n'est pas conscient de la virtualisation
  - L'hyperviseur réecrit en temps réel toutes les instructions du SE invité et simule le hardware sous-jacent
  - Meilleure perfomance et isolation
  - Exemples : VMware ESXi et Microsoft Hyper V Server

L'hyperviseur fournit à chaque VM tous les services du système physique y compris le BIOS virtuel, périphériques virtuels, mémoire virtualisée par émulation



Techniques de Virtualisation

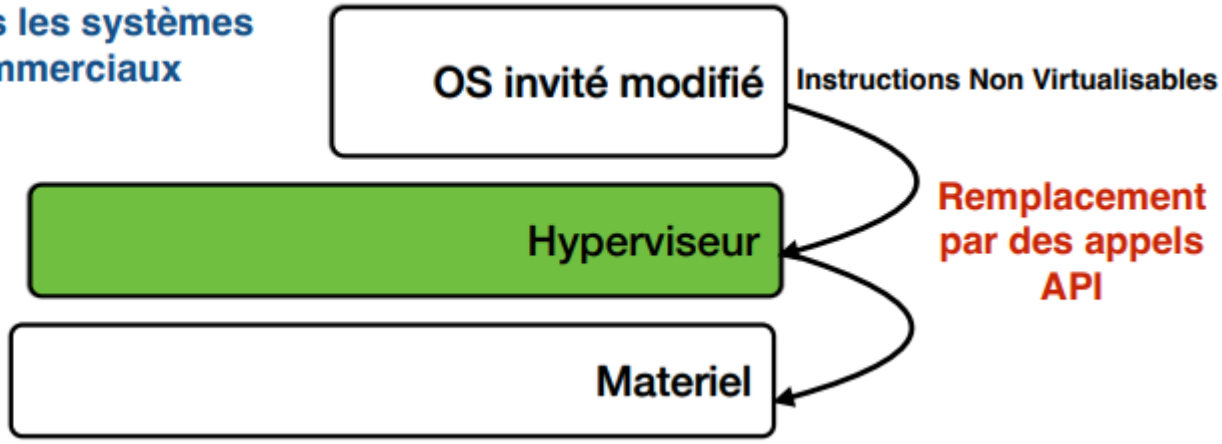
- 1ère Génération : Full virtualization (*assitée par matériel*)
  - SE invité non modifié et n'est pas conscient de la virtualisation
  - Pas besoin de traduire les instructions non virtualisables
  - Un nouveau noyau CPU assure la bonne exécution des instructions critiques
  - Exemples : VMware ESXi et Xen



Techniques de Virtualisation

- 2ème Génération : Paravirtualization (*OS-Assisted Virtualization*)
  - Coopération entre l'hyperviseur et le SE invité
  - Le SE invité est modifié et est conscient de la virtualisation
    - Donc ne supporte pas les SEs non modifiés (par ex., Windows 2000/XP)
  - Exemples : Xen

- **Cette méthode ne supporte pas les systèmes non modifiables (systèmes commerciaux fermés)**



Hyperviseurs disponibles

- KVM: Open source basé sur Linux intégré dans le noyau en 2007. L'hyperviseur le plus utilisé.
  - Type 2, Full Virtualization, Hardware-Assisted Virtualization, Paravirtualization
- XEN : Hyperviseur open source basé sur linux créé en 2003 à Cambridge University et acquis par Citrix Inc en 2007.
  - Type 1, Hardware-Assisted Virtualization, Paravirtualization
- Hyperviseur VMware's ESXi: propriétaire à VMware inc., développé et extrêmement stable
  - Type 1, Full Virtualization, Hardware-Assisted Virtualization
- Hyper-V: commercialisé par Microsoft
  - Type 1 , Full Virtualization, Hardware-Assisted Virtualization

Hyperviseurs disponibles

- Hyperviseur VMware Workstation : propriétaire à VMware inc.
  - Type 2, Full Virtualization

- Virtual Box : Open Source, gratuit, aquis par Oracle Corporation en 2010
  - Type 2, Hardware-Assisted Virtualization, Paravirtualization

Plan

1. Virtualisation
   1. Définitions et concepts
   2. Types d'hyperviseurs
   3. Avantages de la virtualisation
   4. Migration des machines virtuelles
   5. Accès au réseau
2. VM vs. Container
3. Systèmes de gestion de machines virtuelles (Exemple: SANDPIPER)

Avantages de la virtualisation

- Consolidation de serveurs
  - Utiliser moins de serveurs physiques
  - Réduire la consommation d'énergie
  - Optimiser l 'utilisation des ressources : les ressources de chaque serveur sont pleinement utilisées
  - ➢ Réduire les coûts
- Faciliter la gestion des pannes et des désastres
  - Création de MVs de backup
    - Création de snapshot (état de la MV et du disque à un instant particulier)
    - Clonage de MV
  - Déploiement facile de backup

Avantages de la virtualisation (suite)

- Migrer facilement les applications d'un serveur à un autre
  - Quand un serveur est surchargé
  - Quand un serveur doit être remplacé par un nouveau qui est probablement plus puissant
- Augmenter et réduire facilement les ressources dédiées à une MV
- Meilleure sécurité et fiabilité
- Faciliter le test de nouvelles applications sur différents SEs et environnements de développement
- Multi-locataire (*multi-tenant*): plusieurs utilisateurs peuvent partager
  - les mêmes ressources physiques

Avantages de la virtualisation (suite)

- Possibilité de virtualiser les fonctions réseaux (*Network Function Virtualization* – NFV): NFV consisite à exécuter les fonctions réseau (par ex., routeur, commutateur, Firewall, load balancer) dans des machines virtuelles au lieu d'un équipement physique dédié
  - Lancer dynamiquement une fonction réseau
  - Permet de changer le comportement d'une fonction réseau à la volée
  - Offrir une élasticité de la fonction réseau par rapport aux ressources (adaptation dynamique des ressources telles que CPU, mémoire, etc.)
  - Bénéficier d'un mécanisme d'orchestration qui permet de gérer des centaines de fonctions réseau
  - Offrir un environnement multi-locataires (multitenancy) : plusieurs clients

Plan

1. Virtualisation
   1. Définitions et concepts
   2. Types d'hyperviseurs
   3. Avantages de la virtualisation
   4. Migration des machines virtuelles
   5. Accès au réseau
2. VM vs. Container
3. Systèmes de gestion de machines virtuelles (Exemple: SANDPIPER)

Migration des MVs

- *Migration* d'une MV : Le déplacement d'une machine virtuelle d'un serveur vers un autre serveur (qui se trouve éventuellement dans un autre centre de données)

- Avantages de la migration : adapter l'emplacement des MVs aux différents besoins potentiels
  - Consolider les serveurs
  - Équilibrer la charge
  - Améliorer la localité des données et du réseau
  - Réduire la consommation de l'énergie
  - Réduire le coût d'hébergement
  - Faciliter la maintenance et la gestion des pannes

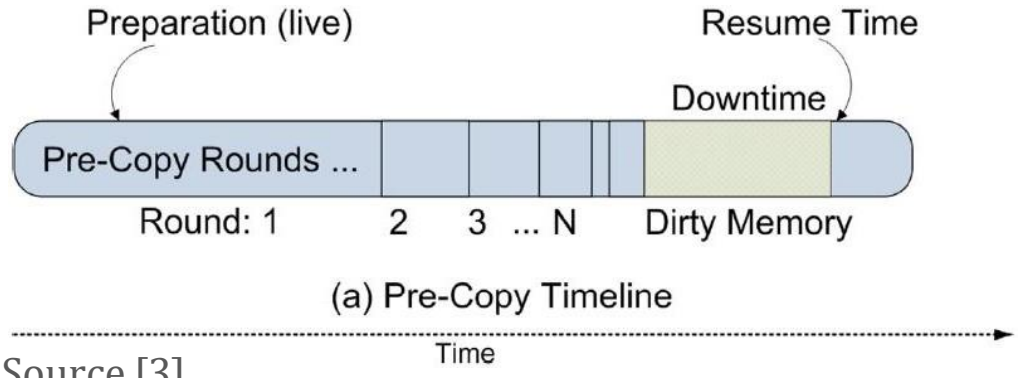
Migration des MVs

- Problèmes :
  - Interruption de service
    - Temps de copier la MV
    - Adressage IP, MAC, VLANs
  - Consommation de ressources
    - Bande passante
    - CPU et mémoire
  - Sécurité
    - Copie de mémoire
    - Vulnérabilité du module de migration

Migration des MVs temps réel

Pre-Copy migration :

- La MV source reste en marche, le contenu de la mémoire est copié
- Une itération : les changements en mémoire (*dirty pages*) sont copiés
- Après plusieurs itérations (jusqu'au moment où le nombre de page "dirty" devient faible ou lorsqu'un seuil de temps est fixé), la VM source est suspendue, les dirty pages sont copiés et la VM destination prend la relève.

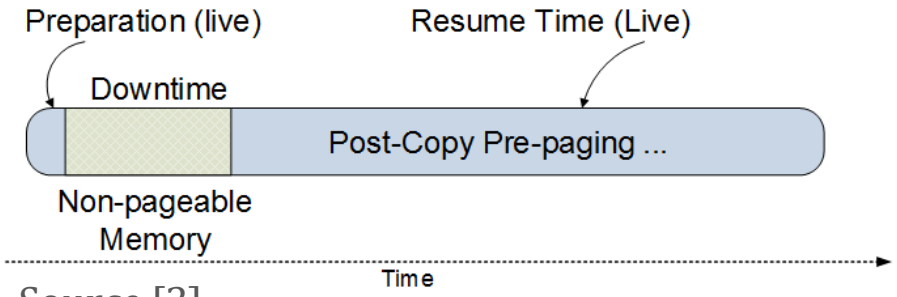


Source [3]

Migration des MVs temps réel

Post-Copy migration :

- La MV source est suspendue, l'état d'exécution de la source (état du CPU, registres et mémoire *non-pageable*) est copié vers la machine destination
- La destination prend la relève
- Le contenu de la mémoire est récupéré progressivement à partir de la source
- Cela peut être amélioré grâce au *pre-paging* : ramener pro-activement les pages mémoire ayant une haute probabilité d'être demandés



Source [3]

(b) Post-Copy Timeline

Migration des MVs temps réel

Pre-Copy vs. Post-Copy migration :

- Pre-Copy :
  - La durée de la migration peut être longue
  - La MV source garde toujours le dernier état ➔ En cas de problème à la destination, on garde la source.
  - Chaque page mémoire peut être transférée plusieurs fois.
  - Le succès de l'opération dépend largement du modèle d'accès à la mémoire
- Post Copy :
  - Durée de la migration moins élevée
  - chaque page mémoire est transférée une seule fois.
  - L'état de la MV est distribué sur les deux MVs pour une certaine durée
  - Le succès de l'opération dépend largement du modèle d'accès à la mémoire

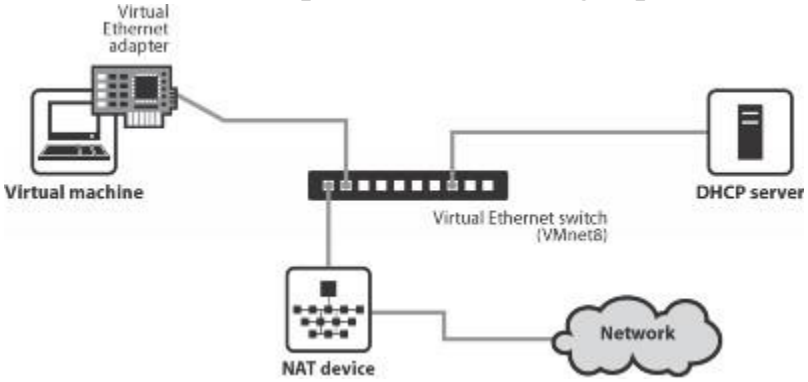
Plan

1. Virtualisation
   1. Définitions et concepts
   2. Types d'hyperviseurs
   3. Avantages de la virtualisation
   4. Migration des machines virtuelles
   5. Accès au réseau
2. VM vs. Container
3. Systèmes de gestion de machines virtuelles (Exemple: SANDPIPER)

Accès au réseau

Trois façons permettant aux MVs d'accéder au réseau :

- *Network Bridging* (pont) : le SE invité a un accès direct à la carte réseau indépendamment du SE hôte donc communique directement avec le réseau
- *Netwrok Address Translation* (NAT) :
  - Le SE invité est donné une carte réseau virtuelle qui est connecté à un NAT simulé par l'hyperviseur.
  - Tout le trafic réseau sortant est envoyé par la carte réseau virtuelle vers le SE hôte à travers le NAT pour être envoyé par la carte réseau



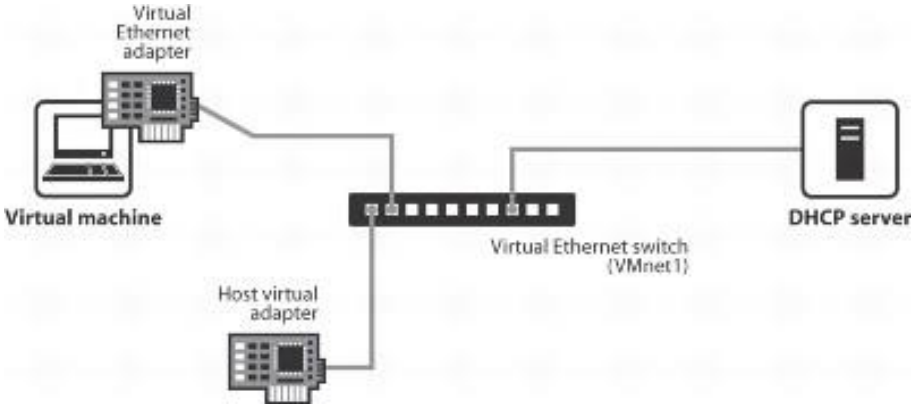
Accès au réseau à travers le NAT [2]

SE : Système d'Exploitation

Accès au réseau (suite)

Trois façons permettant aux MVs d'accéder au réseau :

- *Host-Only Networking* : Le SE invité (MV) a une carte réseau virtuelle qui est seulement connecté au SE dans la machine hôte (un réseau privé entre les deux SEs)



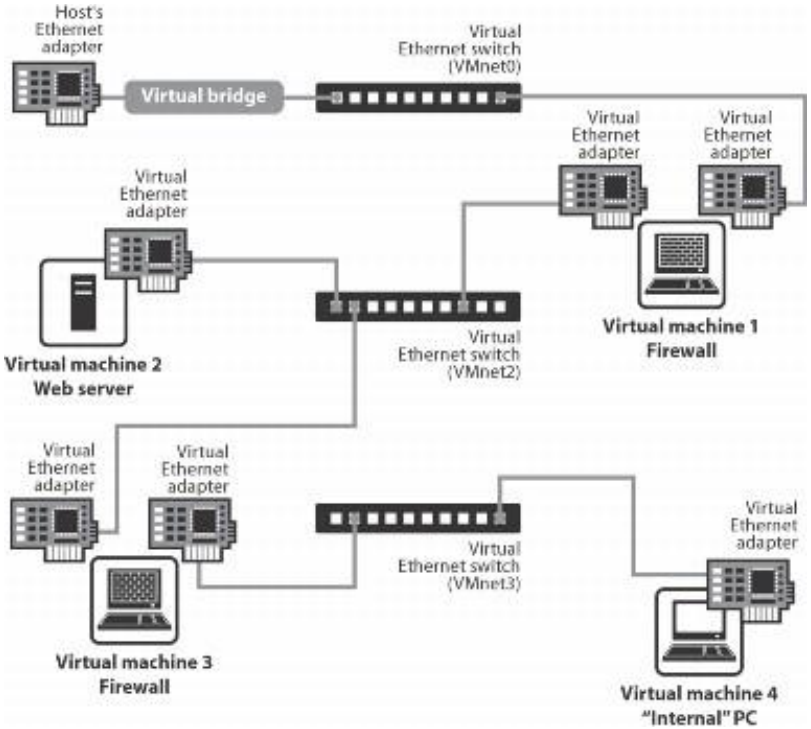
Host Only Networking [2]

Accès au réseau(suite)

- Possibilité de créer des réseaux personnalisés reliant les MVs

Réseau externe

Serveur Web



Infrastructure virtuelle personnalisée [2]

Dans cette configuration, un serveur Web est connecté à travers un pare-feu à un réseau externe. L'administrateur peut se connecter au serveur Web via un deuxième pare-feu.

Administrateur

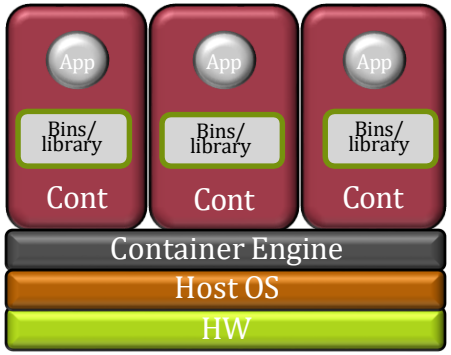
Plan

1. Virtualisation
   1. Définitions et concepts
   2. Types d'hyperviseurs
   3. Avantages de la virtualisation
   4. Migration des machines virtuelles
   5. Accès au réseau
2. VM vs. Container
3. Systèmes de gestion de machines virtuelles
   1. Exemple: SANDPIPER

Containers vs VMs

Avantages des Containers:

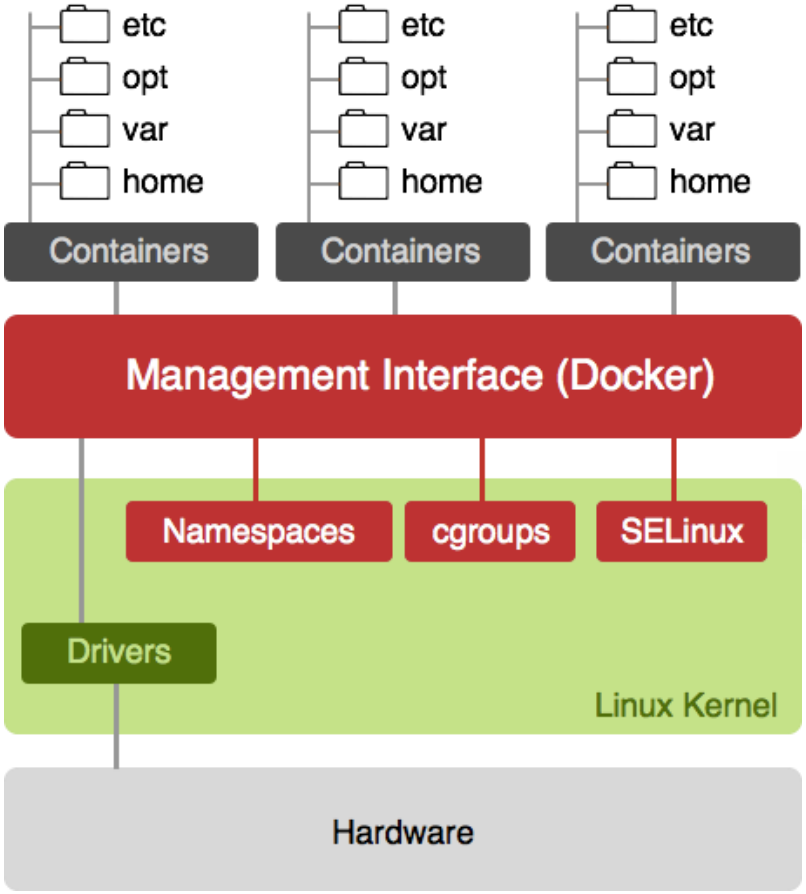
- Plus besoin de créer des MVs sur une machine physique pour exécuter différents processus dans un environnement séparé
- Les processus de l'espace usager s'exécutent sur le même Système d'Exploitation (donc SE partagé), mais sont isolés les uns des autres
- Chaque espace usager (ex. une application et ses dépendances/librairies) peut bénéficier des ressources nécessaires à l'exécution des processus qu'il héberge (vCPU, mémoire et fichier système)
- Meilleure performance et attribution flexible des ressources par rapport à la virtualisation par hyperviseur



Infrastructure pour les *containers*

Infrastructure pour les VMs

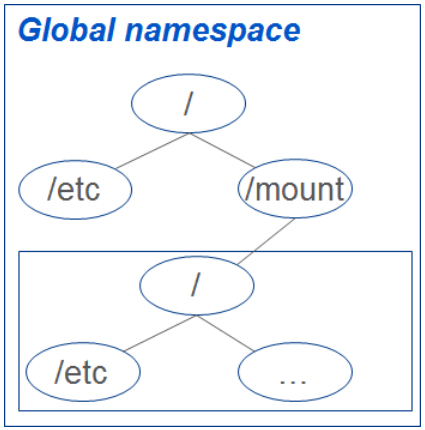
Fonctionnement des containers



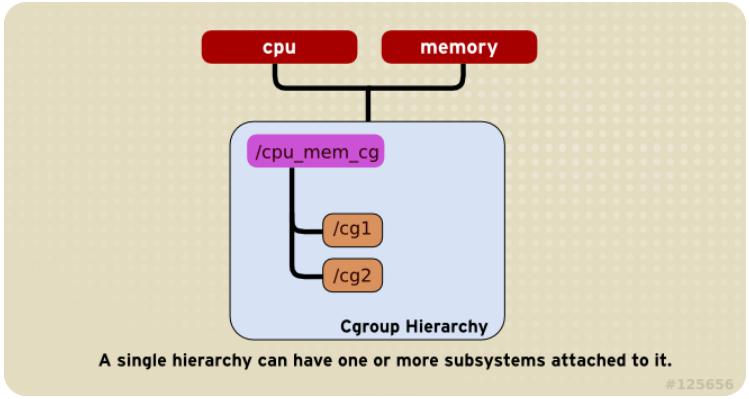
Fonctionnement des containers

Les containers utilisent des fonctionnalité de de Linux:

- Isolation des ressources: NameSpaces

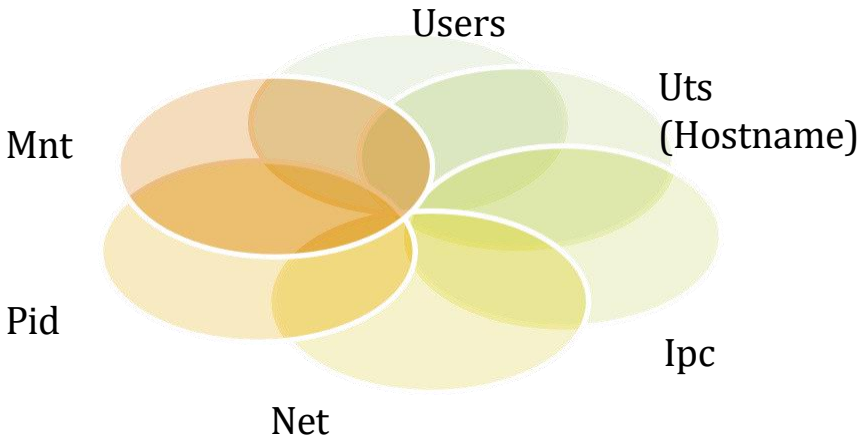


- Gestion des ressources: Control groups (Cgroups)



Namespaces

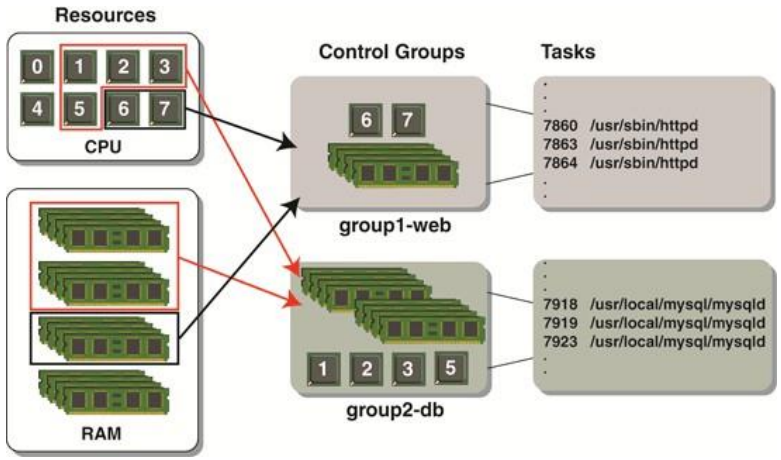
- Namespaces est une fonctionnalité du Kernel Linux permettant de partitionner les ressources tel que chaque groupe de processus peut voir un sous-ensemble de ressources
  - Exemples de telles ressources sont les IDs de processus, les noms d'hôte, les IDs utilisateurs, les noms de fichier et certains noms associés à l'accès au réseau et à la communication interprocessus.
- Chaque processus est associé à des namespaces et ne peut voir ou utiliser que les ressources associées à ces namespaces
- 6 types de namespaces sont implémentés:
  - Mnt (mount points, file systems)
  - Pid (processes)
  - Net (network stack)
  - Ipc (Inter Process Communication)
  - Uts (hostname)
  - User ID (UIDs)



Control groups (cgroups)

Les groupes de contôle (Cgroups) sont une fonctionnalité de Linux Kernel permettant de:

- **Limiter** un ensemble de processus à utiliser seulement une part des ressources (par ex., mémoire)
- **Prioritiser** un groupe pour utiliser une resource (par ex. CPU, disk I/O throughput)
- **Mesurer** l'utilisation d'une resource
- **Contrôler** and geler un groupe de processus, les "checkpointer" (sauvegarder) et les redémarrer



Plan

1. Virtualisation
   1. Définitions et concepts
   2. Types d'hyperviseurs
   3. Avantages de la virtualisation
   4. Migration des machines virtuelles
   5. Accès au réseau
2. VM vs. Container
3. Systèmes de gestion de machines virtuelles (Exemple: SANDPIPER)

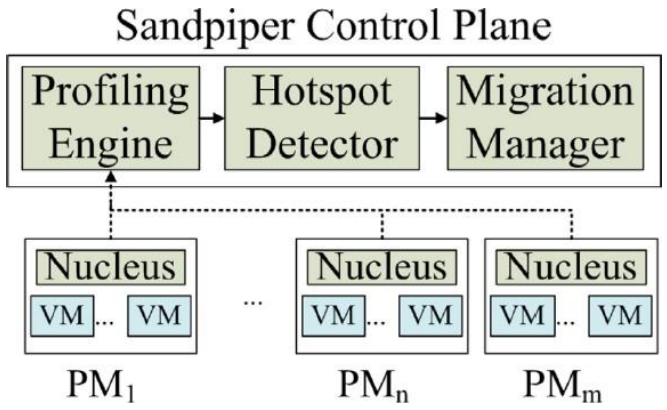
Gestionnaire de MVs

- Fonction : décider automatiquement de l'emplacement et la migration des machines virtuelles dans les infrastructures à large échelle
- Objectifs :
  - Consolider les serveurs
  - Équilibrer la charge
  - Améliorer la localité des données et du réseau
  - Réduire la consommation de l'énergie
  - Réduire les coûts d'hébergement
  - Faciliter la maintenance et la gestion des pannes

Exemple: SANDPIPER

- Objectifs : éviter les surcharges des serveurs (en CPU, mémoire et disque)

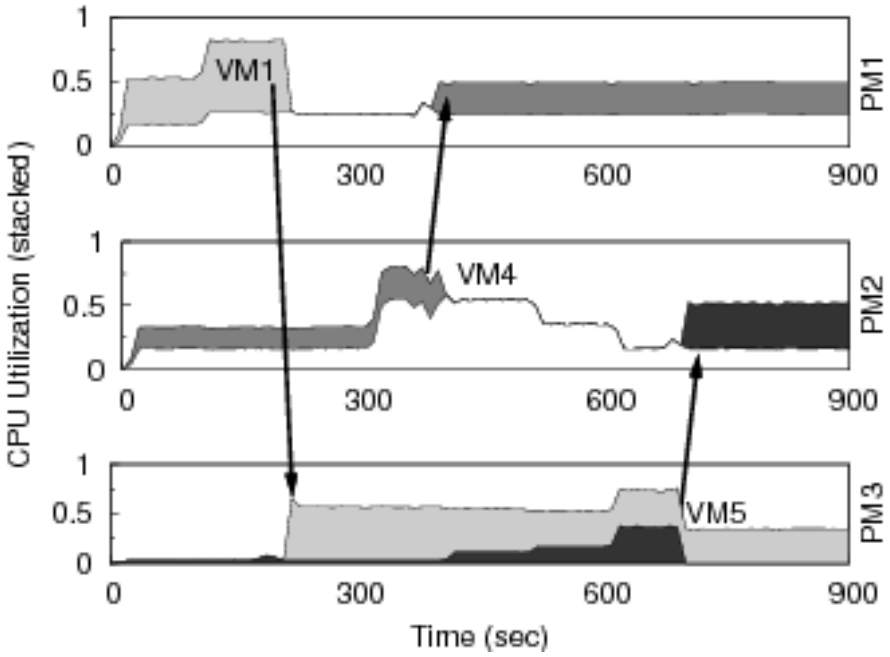
- Exemple: SANDPIPER permet de
  - Détecter les noeuds surchargés (hotspots)
  - Migrer les MVs responsables de la surcharge vers des machines moins chargés



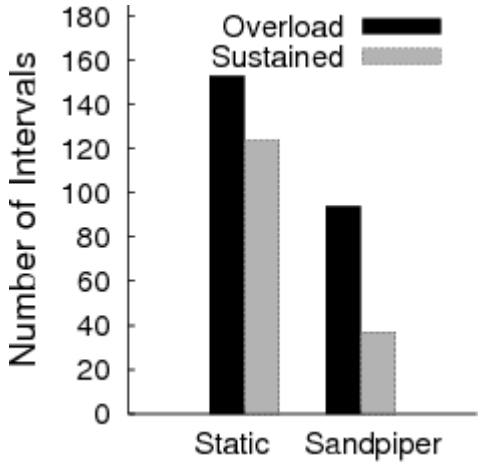
PM : Physical Machine

Exemple: SANDPIPER

Exemple: SANDPIPER



Série de migrations pour résoudre les hotspots.



Exemple: SANDPIPER élimine tous les hotspots et réduit le nombre d'intervalles éprouvant une surcharge continue par 61% (centre de données de 16 serveurs et 35 MVs)

Bibliographie

1 Amit Singh, An Introduction to Virtualization, January 2004 (http://www.kernelthread.com/publications/virtualization/)

2 VMWare Workstation, https://www.vmware.com/support/ws55/doc/index.html

3 R. Boutaba, Q. Zhang, and M. F. Zhani, "Virtual machine migration: Benefits, challenges and approaches," in Communication Infrastructures for Cloud Computing: Design and Applications, Book edit by H. T. Mouftah and B. Kantarci, Eds. USA: IGI-Global, 2013.

4 T. Wood, P. Shenoy, A. Venkataramani, and M. Yousif. Exemple: SANDPIPER: Black-box and gray-box resource management forvirtual machines. Computer Networks, 53:2923–2938, December 2009.

5 M. F. Zhani, Support de cours virtualisation et cloud computing

6 M. Ben Said, Support de cours virtualisation et cloud computing

Namespaces

- 6 types de namespaces sont implémentés:
  - Pays
  - Institut
  - Ville
  - Nom de famille (hostname)
  - Specialité
- Le processus Nadia appartient à
  - Pays : Tunisie (177771)
  - Institut: ESSTHS (88888)
  - Ville : Sousse (03)
  - Nom de famille: Chahed (77663)
  - Specialité: Réseau (0011)
- Le processus Ahmed appartient à
  - Pays : Tunisie (177771)
  - Institut: ESSTHS (88888)
  - Ville : Tunis (01)
  - Nom de famille: Exemple (77668)
  - Specialité: Programmation (0011)