ESGI
Ecole Supérieur de Génie Informatique

Février 2022

Formation Virtualisation des réseaux

Intervenant

- Vincent LAINE
  - Formateur Indépendant – ANDN-Services
  - Ingénieur Systèmes Réseaux

Comprendre son environnement de travail :

Les bases de la virtualisation

Qu’est ce que la virtualisation

- La virtualisation consiste à créer une représentation virtuelle, basée logicielle, d’un objet ou d’une ressource telle qu’un système d’exploitation, un serveur, un système de stockage ou un réseau.

- Ces ressources simulées ou émulées sont en tous points identiques à leur version physique.

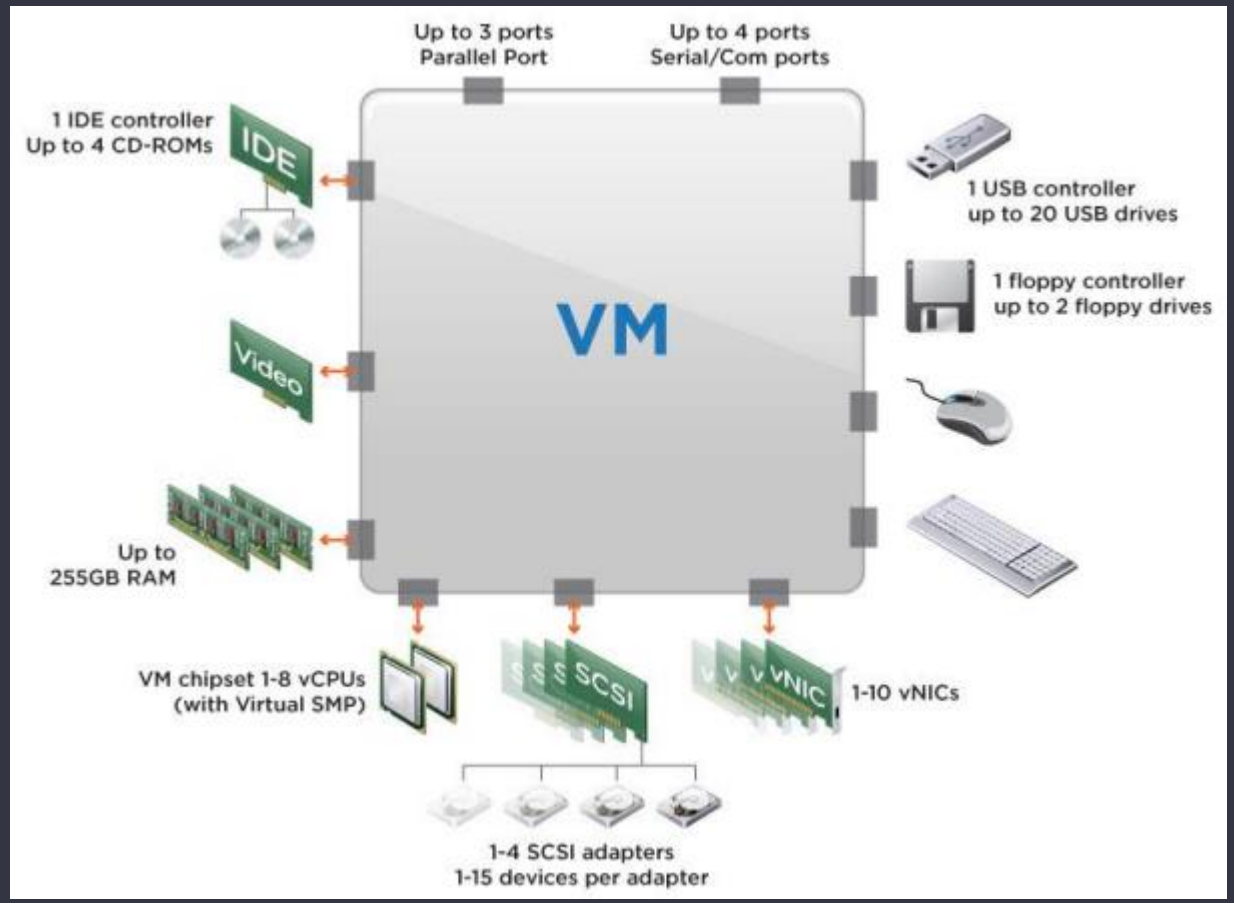
CITRIX®

vmware

VirtualBox

Qu’est ce que la virtualisation

- Un hôte simulé se nomme une machine virtuelle ou VM.
- Une VM permet de virtualiser :
  - Le CPU
  - La RAM
  - Le Stockage
  - Le *Réseau*



Qu’est ce que la virtualisation

- Pour fonctionner, une VM doit être hébergé sur une machine bien précise :
  - Un Hyperviseur
- Il existes plusieurs types d’hyperviseurs :
- Type 1 – Bare Metal
- Type 2 - Hébergé

Qu’est ce que la virtualisation

- Hyperviseur de type 1(ou Bare-metal) :

- Un hyperviseur de type 1, également appelé hyperviseur de système nu ou natif, s'exécute directement sur le matériel de l'hôte pour gérer les systèmes d'exploitation invités.

- Il prend la place du système d'exploitation de l'hôte et planifie directement les ressources des machines virtuelles sur le matériel. (hyper-V, ESXi, kvm…)

Qu’est ce que la virtualisation

- Hyperviseur de type 2 :

- Un hyperviseur de type 2, également appelé hyperviseur hébergé, s'exécute sur un système d'exploitation traditionnel en tant que couche logicielle ou application.

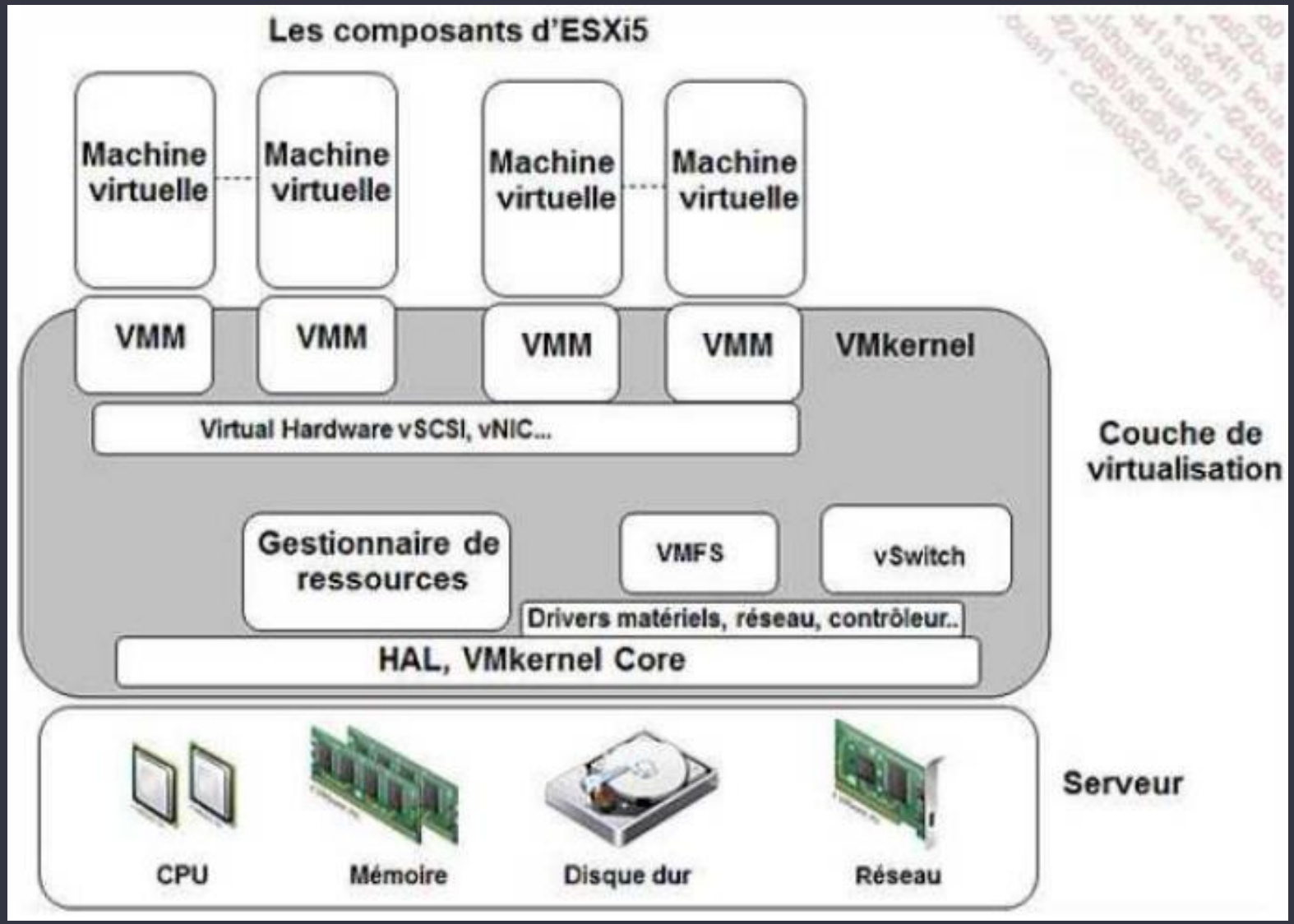
- Il fonctionne en dissociant les systèmes d'exploitation invités du système d'exploitation hôte.

Qu’est ce que la virtualisation

- Les deux acteurs majeurs de la virtualisation des systèmes en entreprises :
- Vmware
  - Vcenter
  - Vsphere
  - ESXi
- Hyper-V (Microsoft)

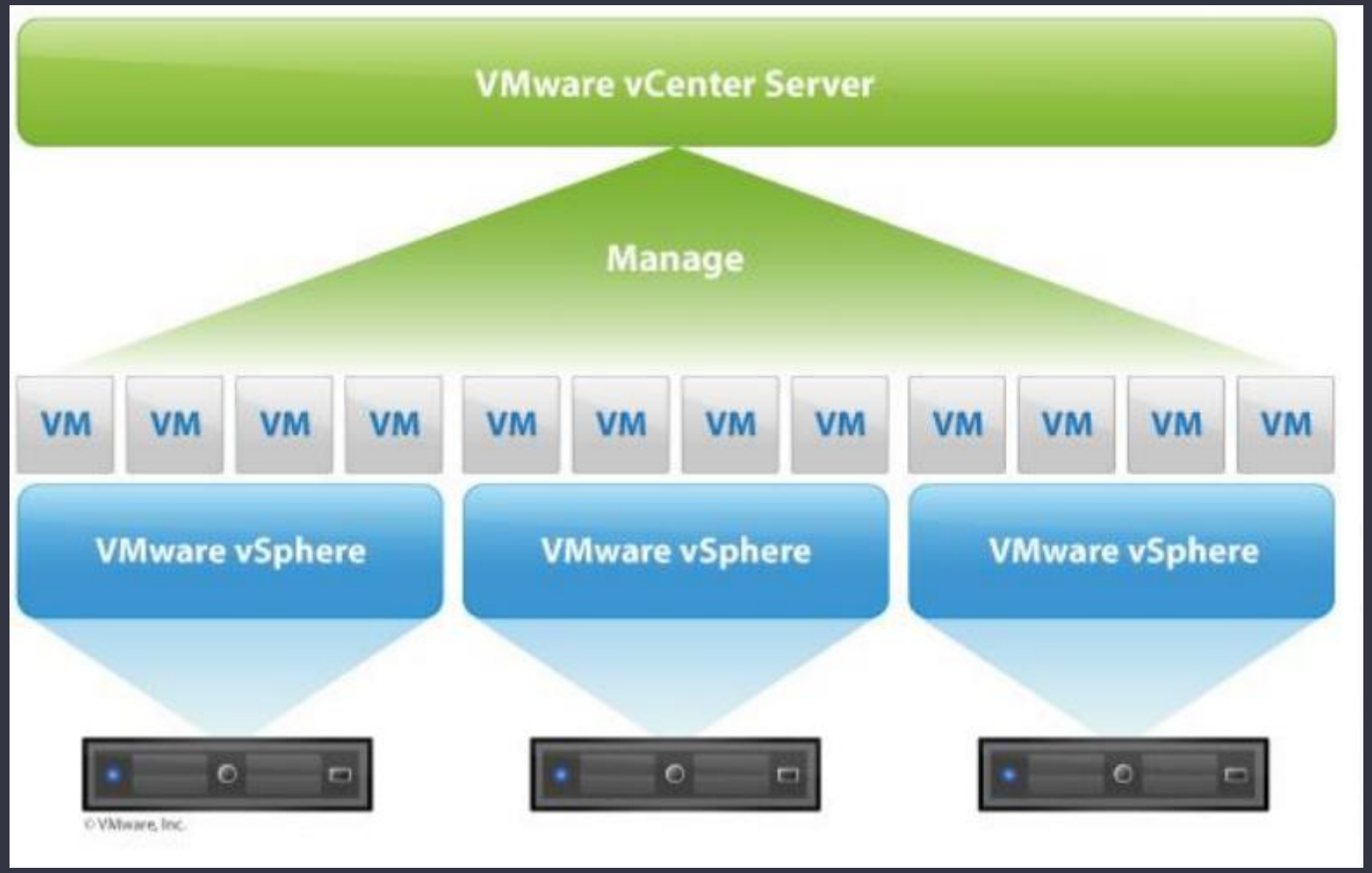
Architecture de Vcenter/ESXi

- Les composants d'un ESXi



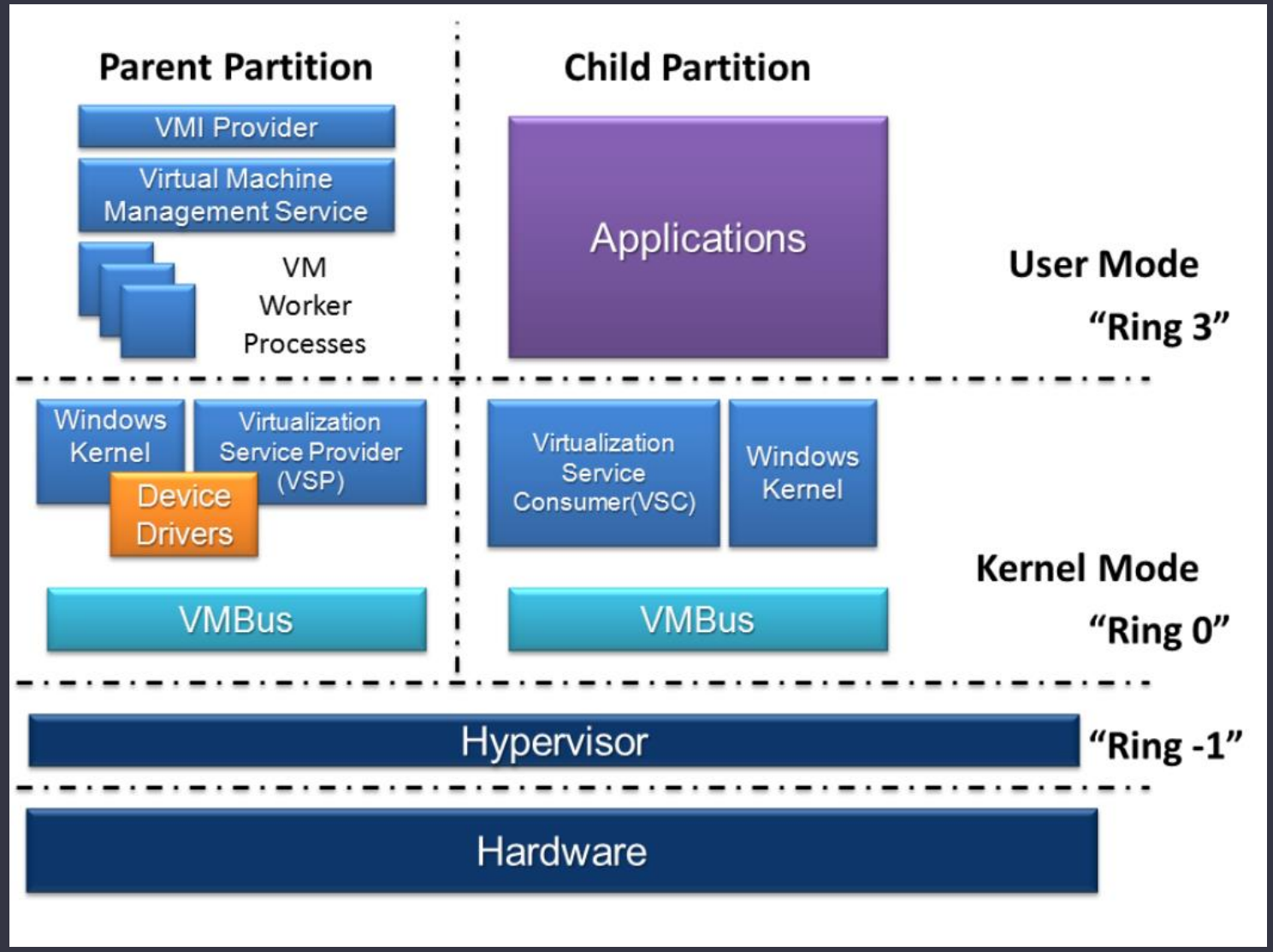
Architecture de Vcenter/ESXi

- Vcenter : Vcenter Serveur permet de gérer et d'administrer des ESXi ainsi que les VM et le RESEAU



Architecture Hyper-V

- Hyper-V fonctionne de la même manières que VMWare.



- Virtualisation
  - TP
  - Installation de VMware Workstation
  - Installation de 2 VM Linux

La Virtualisation réseau

Qu’est qu’un réseau informatique ?

- un réseau est défini par la mise en relation d’au moins deux systèmes informatiques au moyen d’un câble ou sans fil, par liaison radio.
- Le réseau le plus basique comporte deux ordinateurs reliés par un câble.

Pourquoi virtualiser le réseau ?

- La virtualisation des fonctions réseau (NFV : Network Function Virtualization) permet de virtualiser les services réseau (routeurs, pare-feu, modules d'équilibrage de charge, etc.) traditionnellement exécutés sur du matériel propriétaire.

- Grâce à la NFV, les fournisseurs sont en mesure de proposer leurs services plus rapidement et à moindre coût, et de tirer parti de l'automatisation pour s'adapter aux exigences d'évolutivité et d'agilité des clients.

- Une architecture NFV comprend les éléments suivants :

  - Des applications logicielles (VNF: Virtualized Network Function) qui fournissent des fonctions réseau, telles que le partage de fichiers, les services d'annuaire et la configuration d'IP.

  - L'infrastructure de virtualisation des fonctions réseau (NFVi) consiste en un ensemble de composants d'infrastructure (calcul, stockage, réseau) sur une plateforme, qui prend en charge des logiciels, par exemple un hyperviseur tel que KVM, ou une plateforme de gestion de conteneurs, nécessaire pour exécuter des applications réseau.

  - Le composant de gestion, d'automatisation et d'orchestration réseau ou MANO (Management, Automation and Network Orchestration) fournit la structure permettant de gérer l'infrastructure NFV et le provisionnement de nouvelles fonctions réseau virtualisées.

Mise en réseau logicielle (SDN) et virtualisation des fonctions réseau (NFV)

- La SDN (Software Define Network) sépare les fonctions de mise en réseau des fonctions de contrôle du réseau, dans le but de centraliser la gestion et la programmation du réseau.

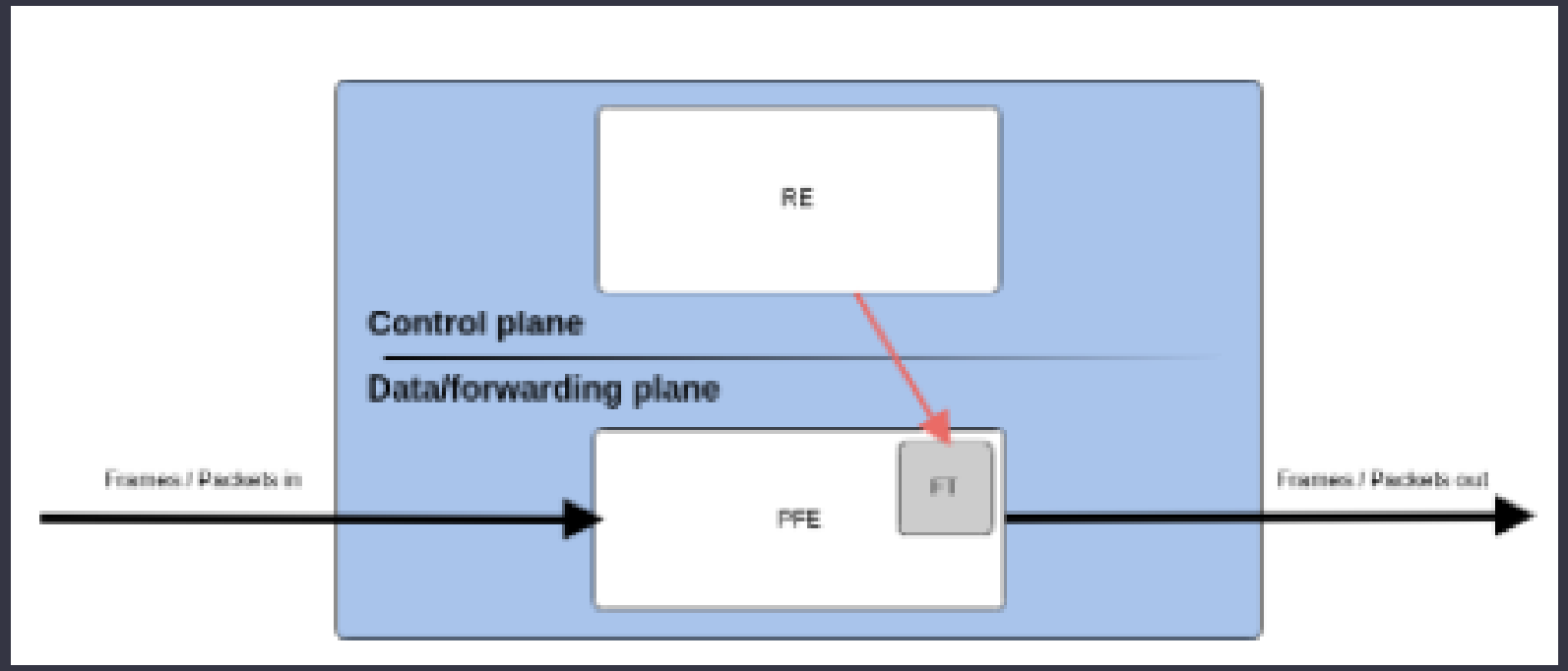
- La NFV permet de dissocier le matériel du logiciel pour les équipements réseau, plusieurs fonctions réseau peuvent par exemple s’exécuter de manière indépendante sur un même matériel générique. Les fonctions réseau peuvent également migrer d’un matériel à un autre.

Comprendre le SDN ?

- Ce que permet le SDN :
  - Gestion plus facile
    - Configuration des équipements
    - Vérification/débogage des contrôleurs plus facile
  - Innovation facile et rapide
    - Moins de dépendances aux équipementiers et aux standards
    - Évolution du logiciel de contrôle indépendamment du hardware
  - Equipement plus simple et moins cher (attention aux tarifs parfois exorbitants des licences !)
    - Logiciel/OS minimaliste

Comprendre le SDN ?

- Un réseau virtuelle ou physique est composé de plusieurs couches :
  - Le plan de contrôle (Couche 3 du modèle OSI)
    - C’est « l’intelligence » de l’équipement : Son CPU
  - Le plan de transmission (couche 2 du modèle OSI)
    - Gère les paquets, trames, toutes les données à traiter



Comprendre le SDN ?

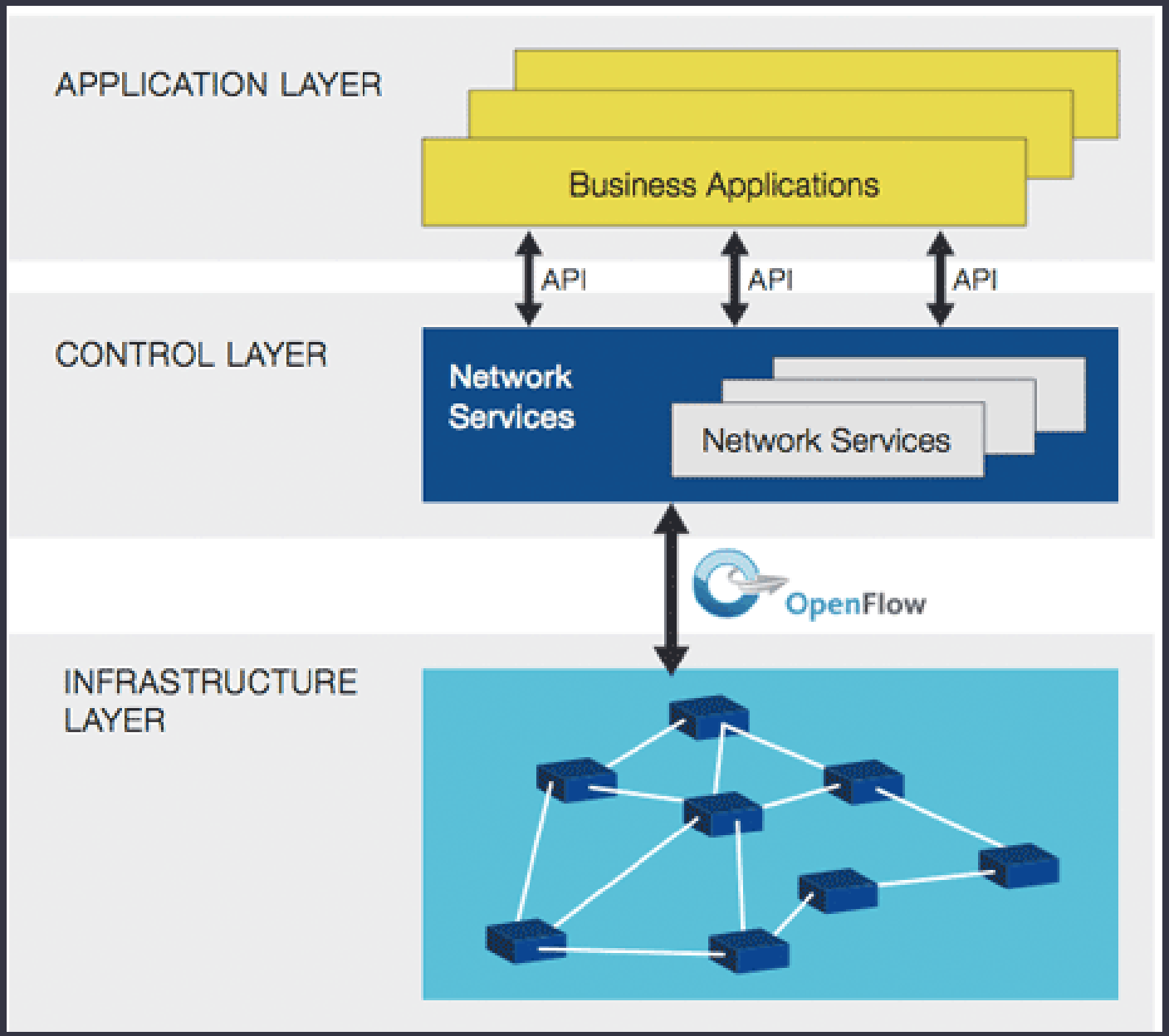
- Dans un réseau « classique », il n’y pas de centralisation du plan de contrôle, chaque routeur peut définir lui-même le chemin à utiliser (avec configuration humaine en amont).

- Les plans de contrôle et les plans de transmission sont donc mutualisés (par exemple équipements de niveau 3)

Le protocole OpenFlow et l'évolution SDN (Software Defined Network).

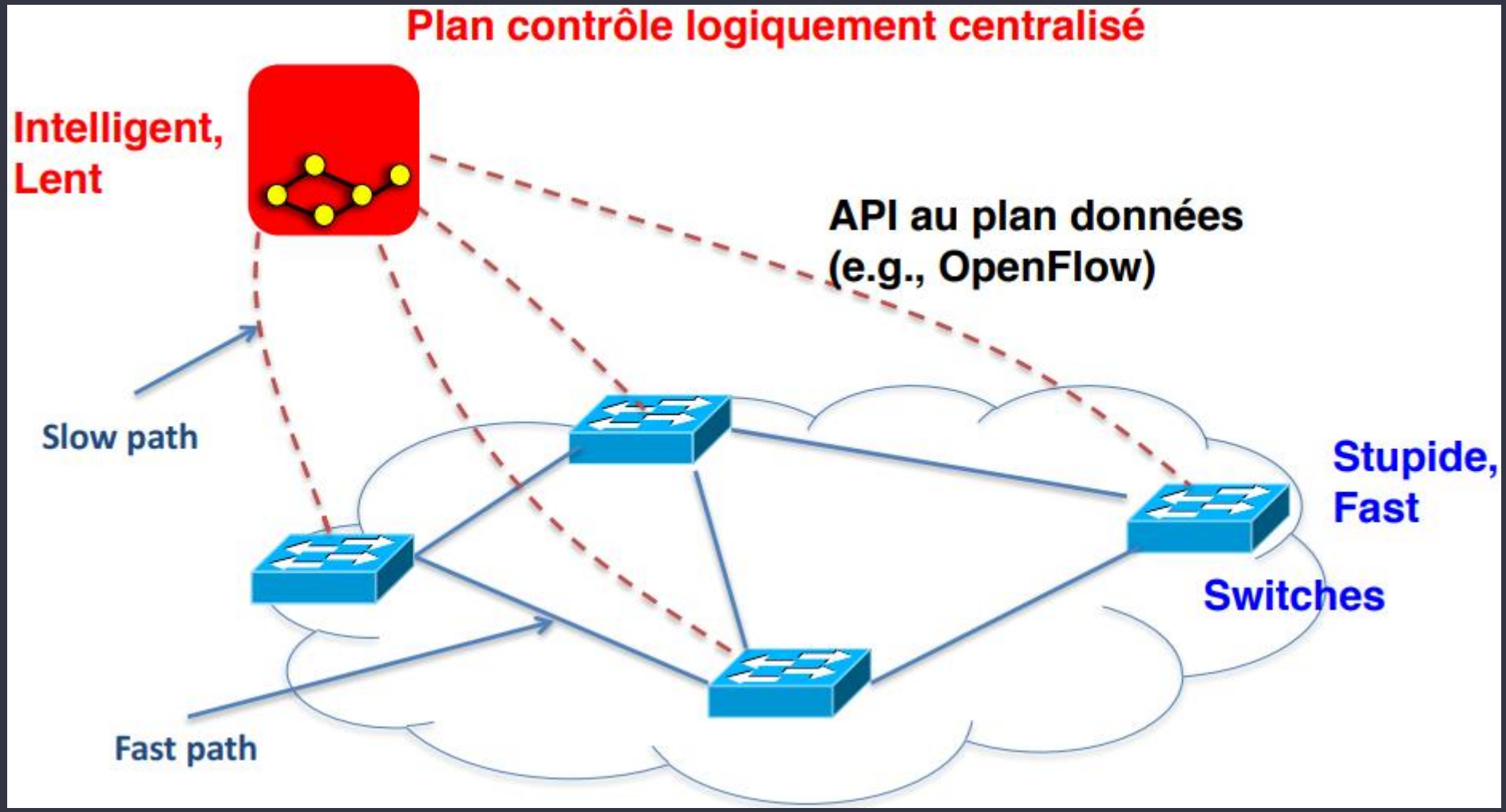
- OpenFlow est un protocole réseau standard qui permet de réaliser une architecture Software-defined networking (SDN). Il est publié par l'Open Networking Foundation.
- C’est un protocole de communication situé entre le plan de contrôle et le plan de données
- Dans OpenFlow, les décisions de routage sont prises par le contrôleur pour chaque flux de données et poussées dans les switches sous forme de simples instructions de commutation.
- OpenFlow contient des mécanismes de QoS, ainsi que le support IPv6.
- Des protocole comme MPLS, OSPF et BGP sont à l’étude pour y être intégrés.

- Comprendre le SDN ?



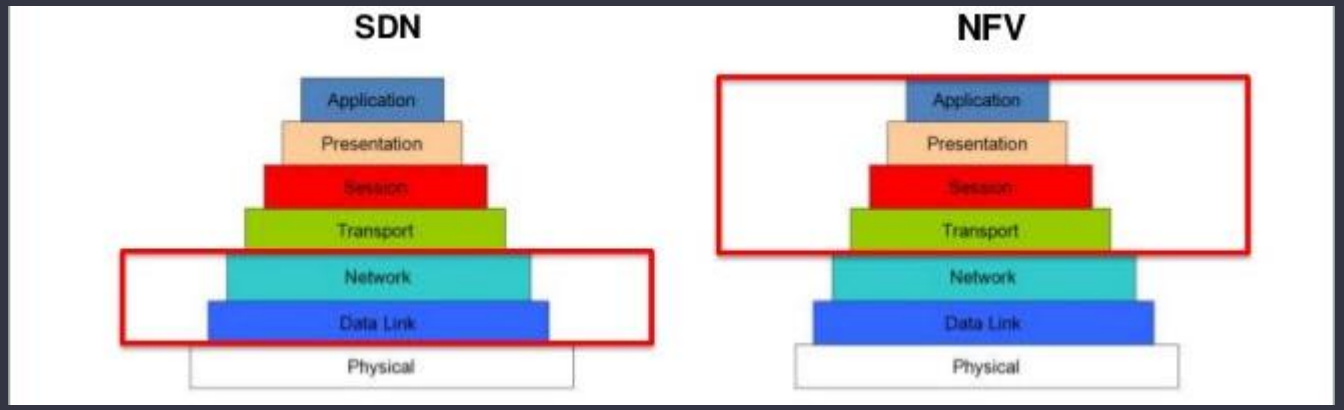
- Le protocole OpenFlow et l'évolution SDN (Software Defined Network).
  - Ce que permet le SDN :
    - Gestion plus facile
      - Configuration des équipements
      - Vérification/débogage des contrôleurs plus facile
    - Innovation facile et rapide
      - Moins de dépendances aux équipementiers et aux standards
      - Évolution du logiciel de contrôle indépendamment du hardware
    - Interopération facile
      - La compatibilité est clé seulement là où il y a des protocoles
    - Equipement plus simple et moins cher
      - Logiciel/OS minimaliste

- Le protocole OpenFlow et l'évolution SDN (Software Defined Network).



Quelles différences entre le SDN et le NFV ?

- Les NFV vont être utilisés sur des matériels standards (VM par exemple) afin de virtualiser le réseau ainsi que ses fonctions (services).
- Ces services VNF incluent le routage, les fonctions pare-feu, l'équilibrage de charge, l'accélération WAN et le cryptage
- Un réseau défini par logiciel consiste à séparer la couche de gestion des flux de la couche de données qui transmet le trafic réseau, l'objectif de cette dissociation étant de créer un réseau centralisé et programmable.
- SDN : Couche 2-3 OSI , NFV Couche 4-7 OSI



La virtualisation réseau

- Dans ce cours, nous n’allons pas utiliser le SDN mais uniquement la virtualisation réseau.
- Logiciel de virtualisation réseau les plus courants :
- GNS 3
- Cisco Netacad Packet Tracer

Différences entre Packet tracer et GNS 3

- GNS 3 :
  - Anciennement Eve-NG
  - Logiciel open source
  - Permet de virtualiser des équipements réseaux avec licences cisco (IOS) ou bien open source sous linux (IOU)

- Packet Tracer
  - Dédié à l’enseignement
  - Propriétaire cisco
  - payant

La virtualisation réseau

Pour du test ou de l’apprentissage,

Vmware Workstation et Virtual Box permettent aussi de faire de la virtualisation réseau.

Workstation peut même être couplé avec GNS3

Il est possible de virtualiser des firewalls, des routeurs et ainsi créer un réseau complexe (avec suffisamment de ressources sur l’hôte)

- TP virtualisation réseau
  - Configurer 2 réseaux d’entreprise connectés sur Internet
  - Relier les 2 réseaux avec un VPN

- # Docker

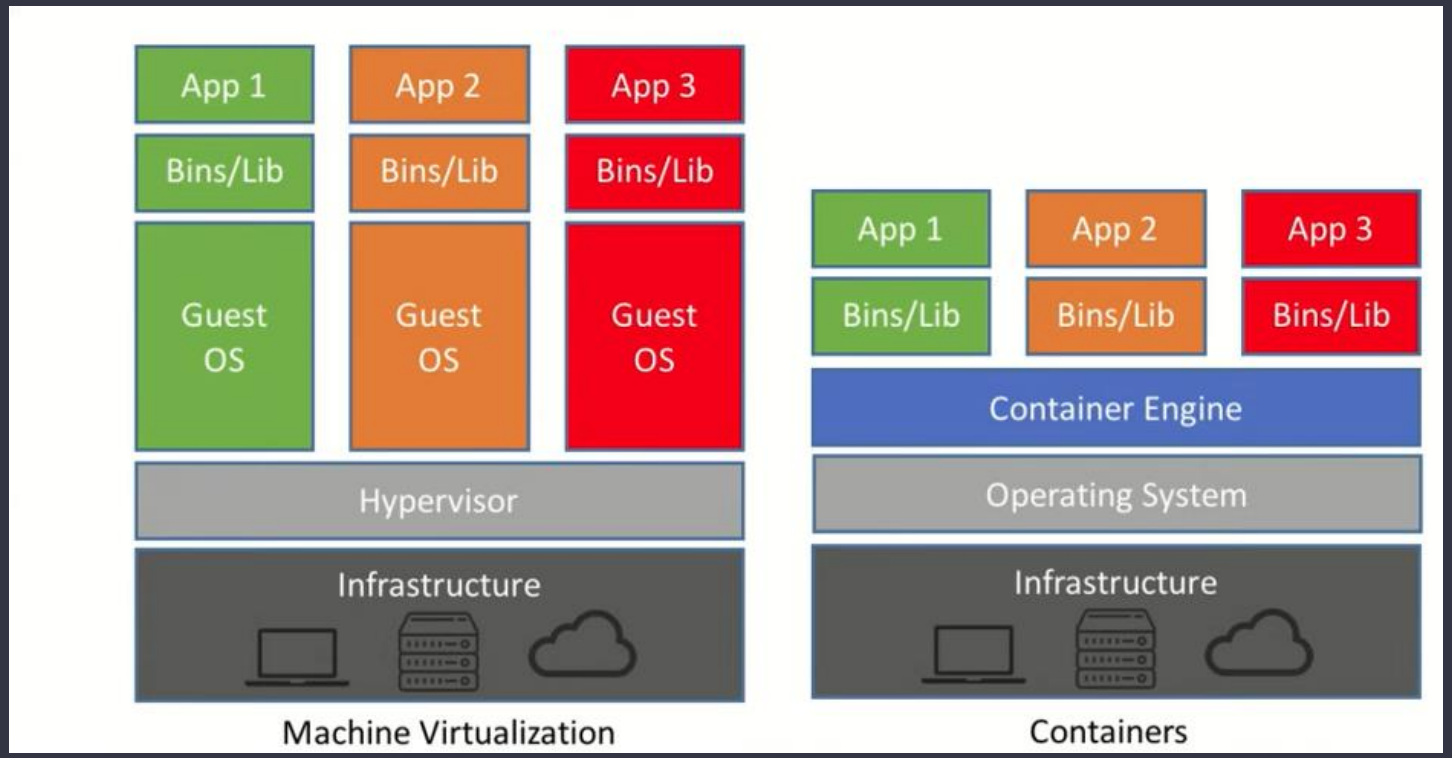
  - Qu’est ce que Docker ?

    Docker permet d'embarquer une application dans un ou plusieurs containers logiciels qui pourra s'exécuter sur n'importe quel serveur machine

    Il permet un déploiement rapide d’application quelque soit le serveur

Docker

- Différence entre Docker et Machine Virtuelle



- Machine Virtuelle
  - Baser sur un Hyperviseur
  - Permet d'émuler des machines complètes : un Os avec une ou plusieurs applications
  - Machine virtuelles lourdes (plusieurs giga)

- Docker
  - Plus léger, baser sur un moteur de conteneur (Docker)
  - Sur la couche supérieure, on retrouve les applications installées dans chacun des conteneurs
  - Un seul système d'exploitation, celui de la machine hôte qui fait bénéficier de ses ressources aux différents conteneurs
  - Conteneur plus léger (rarement au-dessus de 500 Mo)
  - Permet de segmenter une application en micro-service (un docker par service : apache, base de données ...)

- # Docker Hub
  - C’est quoi ?

    https://hub.docker.com/

    Regroupe toutes les images docker disponibles

    Permet de retrouver toutes les images officielles

    Et celle de la communauté

    Tout le monde peut déposer son propre conteneur (inscription obligatoire)

Quelques commandes de base

- Installer Docker : sudo apt-get install docker.io
- Installer une image et la lancer :

docker run ––nom image

docker run -di –nom image (d : detach, I : interactive => permet de garder la main sur l'image)

Lorsqu'on lance un « docker run », docker cherche d'abord l'image en local, si elle n'est pas disponible, il la télécharge depuis Docker Hub (par défaut). Une fois l'image présente en local, le conteneur est lancé.

Quelques commandes de base

- docker ps : permet de lister les processus
- docker ps – a : permet de lister tous les processus docker, même ceux qui sont terminés
- docker exec -tid nomduconteneur sh : Se connecter au conteneur

Exemple avec un serveur web, nginx :

docker run -tid -p 8080:80 --name monnginx nginx:latest

Explication :

Lancer le conteneur nginx, le nommer monnginx, le laisser fonctionner. Rediriger son port 80 sur le 8080 de notre machine locale

Accéder au paramétrage du conteneur :

docker inspect nom du conteneur (permet de voir l'adressage ip entre autre)

Quelques commandes de base

Commande	Explication
docker image ls	Lister les images existantes
docker container ls -a	Lister les conteneurs
docker ps -a	Lister les conteneurs
docker run alpine:latest	Lancer ou créer le conteneur alpine
docker run -di --name alpinetest alpine:latest	Lancer ou créer le conteneur alpine, le nommer alpinetest et le laisser fonctionner (mode Detach Interactive)
docker run -di --name alpinetest alpine:latest sleep infinity	Comme au dessus mais en plus on conserve un process qui tourne sur le conteneur
docker run -tid -p 8080:80 --name monnginx nginx:latest	Lancer le conteneur nginx, le nommer monnginx, le laisser fonctionner. Rediriger son port 80 sur le 8080 de notre machine locale
docker run -tid --name conteneur2 --link conteneur1 alpine	Lancer un conteneur alpine que l'on nomme conteneur2, le linker avec conteneur1 (conteneur2 pourra pinguer conteneur1)
docker start monconteneur	Redémarrer un conteneur arrêté
docker exec -ti monnginx sh	Se connecter au conteneur nommé monnginx avec le shell et le bach
docker stop <nom du conteneur>	Arrêter un conteneur
docker container kill $(docker ps -q)	Tuer tous les conteneurs
docker start <nom du conteneur>	Redémarrer un conteneur
docker rm -f <nom du conteneur>	Supprimer un conteneur (-f permet de forcer, même si le conteneur tourne)
docker rm $(docker ps -a -q)	Supprimer tous les conteneurs
docker inspect <nom du conteneur>	Obtenir les infos sur un conteneur
docker inspect -f "{{.NetworkSettings.IPAddress}}" <nom du conteneur>	Obtenir l'ip d'un conteneur

- TP Docker

Lancer un conteneur NGINX et modifier la page par défaut

Les volumes dans Docker

- Lorsque vous supprimer le conteneur, toutes vos modifications sont perdus car les fichiers sont stockés par défaut dans ce conteneur

- Pour éviter ce problème, il est possible d'utiliser un volume qui sera situé sur la machine hôte

- Vous pourrez supprimer le conteneur, et le recréer en conservant vos fichiers modifiés

Volumes persistant dans Docker

- Lancer un docker avec un volume persistant :

La commande suivante va permettre à docker d'utiliser le répertoire local /srv/data/html/ au lieu du répertoire du docker /usr/share/nginx/html/

Docker run -tid -p 8080 :80 -v /srv/data/html/:/usr/share/nginx/html/ --name nginx nginx:latest

- Autre manière : créer un volume local directement avec docker :

docker volume create nginxlocal
Pour avoir un résumé de la configuration du volume, taper :
docker volume inspect nginxlocal
Le volume est situé ici : /var/lib/docker/nginxlocal/_data
Installer un conteneur nginx utilisant ce volume :
docker run -tid –name nginx -p 8080:80 –
mountsource=nginxlocal,target=/usr/share/nginx/html nginx:latest

Le réseau dans Docker

- Par défaut le réseau utilisé pour tous les conteneurs est le 172.17.0.0/16
- Docker assigne une ip automatiquement et dynamiquement dans ce range pour tout conteneur créé.
- Attention, si un conteneur redémarre, son ip peut changer.
- Par défaut, tous les conteneurs peuvent communiquer entre eux car ils sont créés dans le bridge .
- Il est possible de créer ses propres réseaux afin d'isoler les conteneurs entre eux.
- Les commandes utiles :

docker network ls => liste les réseaux disponibles

docker inspect monréseau => montre le détail du réseau nommé « monréseau »

docker network create -d bridge –subnet 172.18.0.0/24 monréseau => créé le réseau 172.18.0.0/24 nommé monréseau

Merci

Ouvrez les yeux, tout est écrit ☺