

A. Virtualisation vs Conteneurisation

B. Virtualisation de stockage

Faculté des Sciences de Rabat / Master IPS 2020-2021

Virtualisation vs Conteneurisation

Virtualisation VS Conteneurisation

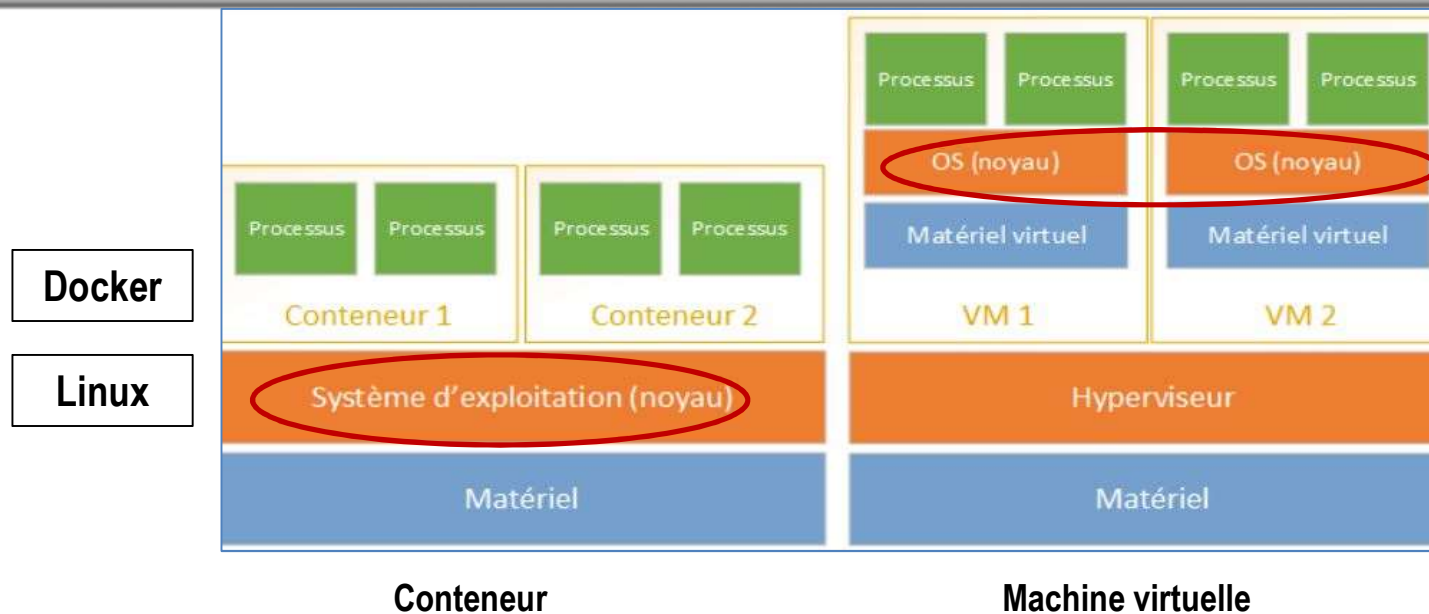
- **En virtualisation**, l'isolation des VMs se fait au niveau **matérielles (CPU/RAM/DISK)** avec un accès virtuel aux ressources de l'hôte via un hyperviseur, ce qui permet d'exécuter plusieurs systèmes d'exploitation en parallèle.
- **En conteneurisation**, l'isolation est réalisée **par le noyau du système d'exploitation hôte**.
- Un conteneur est un concept dont le principe est **d'isoler l'exécution d'ensembles de processus sur un même hôte**. Le noyau et l'accès au matériel reste partagé (à la différence d'une VM).
- Tous les conteneurs et l'hôte se **partage un même noyau Linux** dans lequel des **espaces de nommages** (**namespace** : ce qu'un processus peut voir / limiter un processus à son environnement) sont créés.

Virtualisation vs Conteneurisation

- Sous **Linux**, un conteneur s'exécute de manière **native sur leur système d'exploitation** et **partage le noyau de la machine hôte (même kernel)** avec d'autres conteneurs (vu comme processus normal de l'extérieur mais il se comporte comme une VM de l'intérieur)
- A la différence d'un conteneur qui **se lance à travers l'OS**, une VM à besoin d'un hyperviseur pour se lancer
- La conteneurisation est une méthode qui permet de **virtualiser, à l'intérieur d'un conteneur, les ressources matérielles** (systèmes de fichiers, réseau, processeur, mémoire vive, etc) dont une application a besoin pour être exécutée et **PAS LE SYSTÈME ENTIER DANS LE CAS DE LA VIRTUALISATION.**
- La **containérisation** permet **d'empaqueter une application et toutes ses dépendances ensemble sous la forme** d'un **container** pour assurer que cette application **fonctionne de manière identique dans tous les environnements .**

- ☐ Les développeurs ont ainsi la garantie que ce qui fonctionne sur **leur ordinateur portable** fonctionnera aussi dans **l'environnement de production**.
- ☐ Un conteneur Linux mobilise **moins de ressources** qu'une machine virtuelle.
- ☐ Il propose une interface standard (démarrage, arrêt, variables d'environnement, etc.), assure l'isolation des applications et peut être géré plus facilement en tant que module d'une application plus importante (plusieurs conteneurs)
- ☐ Plutôt que de **virtualiser le hardware comme l'hyperviseur**, le container **virtualise le système d'exploitation**. Il est donc nettement **plus efficient qu'un hyperviseur en termes de consommation des ressources système**.

Virtualisation VS Conteneurisation



- Les conteneurs reposent sur le **noyau du système d'exploitation** qui les accueille.
- Les VMs utilisent le **noyau du système d'exploitation 'OS (noyau)' à l'intérieur de la VM**. Celui-ci n'a pas directement accès aux ressources physiques et utilise **l'hyperviseur pour accéder aux ressources** physiques (processeur, RAM, disque dur)
- L'**isolation** est réalisée à l'aide d'**espaces de nommage** qui sont présents nativement dans le **noyau Linux** sous l'appellation "**cgroups** » (ce qu'un processus peut utiliser en terme de ressources). Cette fonctionnalité est exploitée par différentes technologies de conteneurisation comme Docker, LXC,....

Comment fonctionne la conteneurisation ?

- Dans un **conteneur**, les ressources matérielles (systèmes de fichiers, réseau, processeur, mémoire vive, etc) nécessaires à l'exécution d'une application sont **virtualisées** .
 - Dans ce **conteneur** sont aussi stockées toutes les **dépendances des applications** : **fichiers, bibliothèques, etc**
 - Pour déplacer les applications virtuelles d'un système d'exploitation à un autre, **le conteneur se connecte à leur noyau (*kernel*)** afin d'exploiter les ressources.
- Il n'est donc pas nécessaire comme pour les machines virtuelle, d'installer un nouveau système d'exploitation.
- Le noyau gèrent les ressources de l'ordinateur et permet au différents composants matériels et logiciels de communiquer entre eux.

Conteneurisation : Exemples

Exemple :

- Créer conteneur contenant Ubuntu
- Fonctionnalités proposés par le noyau linux prétendent quelque chose qui ressemble à une VM avec OS Ubuntu
- En réalité , ce n'est pas une VM mais un **processus isolé** s'exécutant sur le même noyau linux

Exemple :

Je suis un développeur, je développe une **application web en PHP**.

Une fois l'application terminée, le service exploitation chargé de déployer mon application en production rencontre un problème: J'ai développé l'application en **PHP version 5.3** et la production tourne en **PHP version 5.1**

Solution 1 :

Mettre à jour le serveur de production

Risque: les autres applications web PHP ne soient pas compatibles avec la mise à jour.

Solution 2 :

Conteneuriser un environnement de développement avec toute ses dépendances (**DOCKER**)

Virtualisation et Conteneurisation : Comparaison

Virtualisation	Conteneurisation
Un seul OS Hôte et plusieurs OS invités	Partage d'un seul et unique système d'exploitation >> Echange de données entre les conteneurs est plus simple et plus rapides que pour les VM.
Chaque VM contient son propre OS	Chaque conteneur ne contient pas de système d'exploitation propre à lui.
	Réduits et prennent moins de place et moins de ressource Serveur
Evolution des OS de plusieurs VM doit être faite manuellement sur chaque Machine.	Facilité de l'évolution technique car toute l'infra repose sur un seul système d'exploitation
	Diversification des systèmes d'exploitation n'est pas possibles car les conteurs reposent sur un seul OS
L'existence d'une couche d'OS rend le déplacement de la VM entièrement de serveurs en serveurs plus lent .	Le déplacement des conteneurs d'un environnement à l'autre est très rapide du fait qu'il se base sur le partage des fichiers de config légers

Conteneurs : Outils

➤ LXC (LinuX Containers) sur Linux (2008) :

Le noyau de linux à la possibilité de créer des conteneurs.

Les conteneurs Linux ont une isolation des systèmes de fichier, des identifiant réseau et utilisateur.

Et également une isolation des ressources (processeur, mémoire, etc).

Ce système de virtualisation est vraiment la base de la conteneurisation

➤ Docker (logiciel libre 2013) :

Outil de conteneurisation célèbre qui permet aux utilisateurs de gérer facilement leurs conteneurs avec une interface une Interface de programmation « Libcontainer » pour démarrer, gérer et arrêter des conteneurs. Il est basée sur le fonctionnement de LXC

➤ RKT « rocket » :

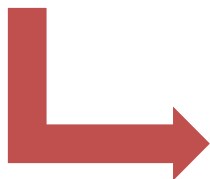
Cet outil est édité par CoreOS est le concurrent de Docker.

Il prend en charge les Images Docker et le format ACI (App Container Images)

virtualisation de stockage

virtualisation de stockage

- 1) Le volume de données croît sans cesse,
- 2) Les applications exigent plus de performances
- 3) Le nouveau matériel est difficile à intégrer.



Stockage des données peut poser problème,

- 4) Les environnements de stockage typiques utilisent **plusieurs périphériques de différents fabricants** mais ils **ne peuvent pas communiquer entre eux**.
- 5) Il est alors **compliqué de les gérer** de manière **centralisée**

SOLUTION : virtualisation du stockage

Définition de la virtualisation de stockage

- ❑ La virtualisation du stockage est un concept qui permet à plusieurs **périphériques de stockage physique** (disques durs, mémoire flash ou lecteurs de bandes) qui sont **regroupés, d'apparaître comme un seul et unique périphérique de stockage**

- ❑ Cet espace de stockage sera présenté aux hôtes qui les verront comme **un pool de stockage** lié (data Store)(**leur propre disque**)

- ❑ Exemple :
Partitionner un disque dur en plusieurs partitions est un exemple de cette virtualisation.

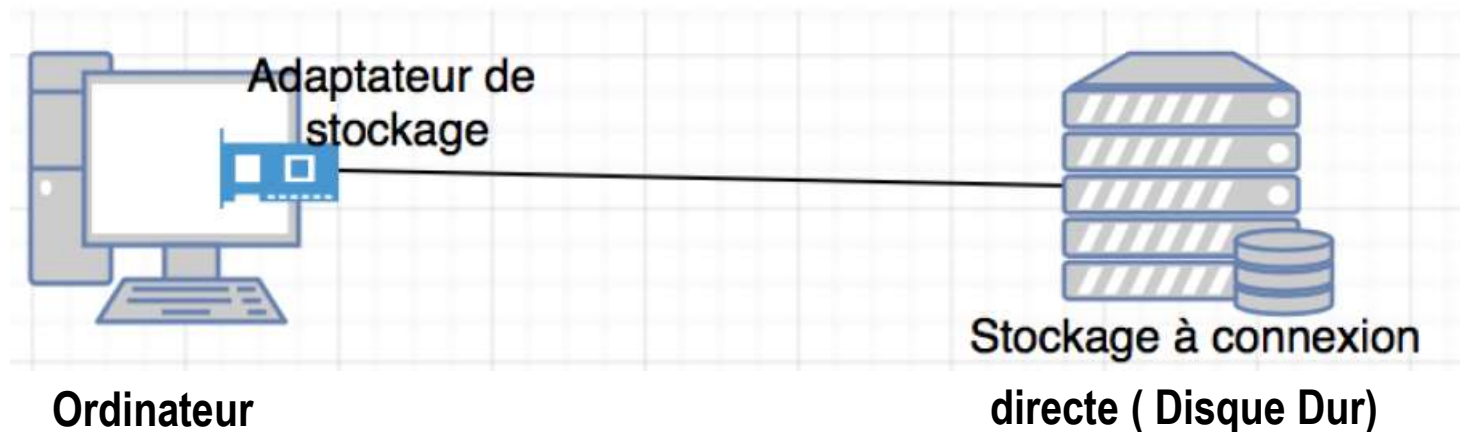
Avantages de la virtualisation de stockage

- ☐ **Homogénéisation** du stockage sur des périphériques de stockage de capacité et de vitesses différentes,
- ☐ Temps d'arrêt réduits
- ☐ **Meilleure optimisation** des performances et de la vitesse avec **disponibilité très élevée (High Availability)**
- ☐ Migration automatique et à chaud les données vers une autre machine virtuelle (**VMotion**)
- ☐ Automatisation l'extension de la capacité de stockage par réduction le besoin de provisionnement manuel

Architectures de stockage : DAS (Direct Attached Storage)

DAS (Direct Attached Storage):

Direct Attached Storage ou **Stockage à connexion directe**, fait référence à un système de stockage informatique qui **est directement connecté au serveur ou ordinateur** au lieu de transiter par un réseau.



- 1) SCSI (Small Computer System Interface),
- 2) SATA (Serial Advanced Technology Attachment)
- 3) SAS (Serial Attached SCSI)

DAS (Direct Attached Storage) : Avantages/Inconvénients

AVANTAGES :

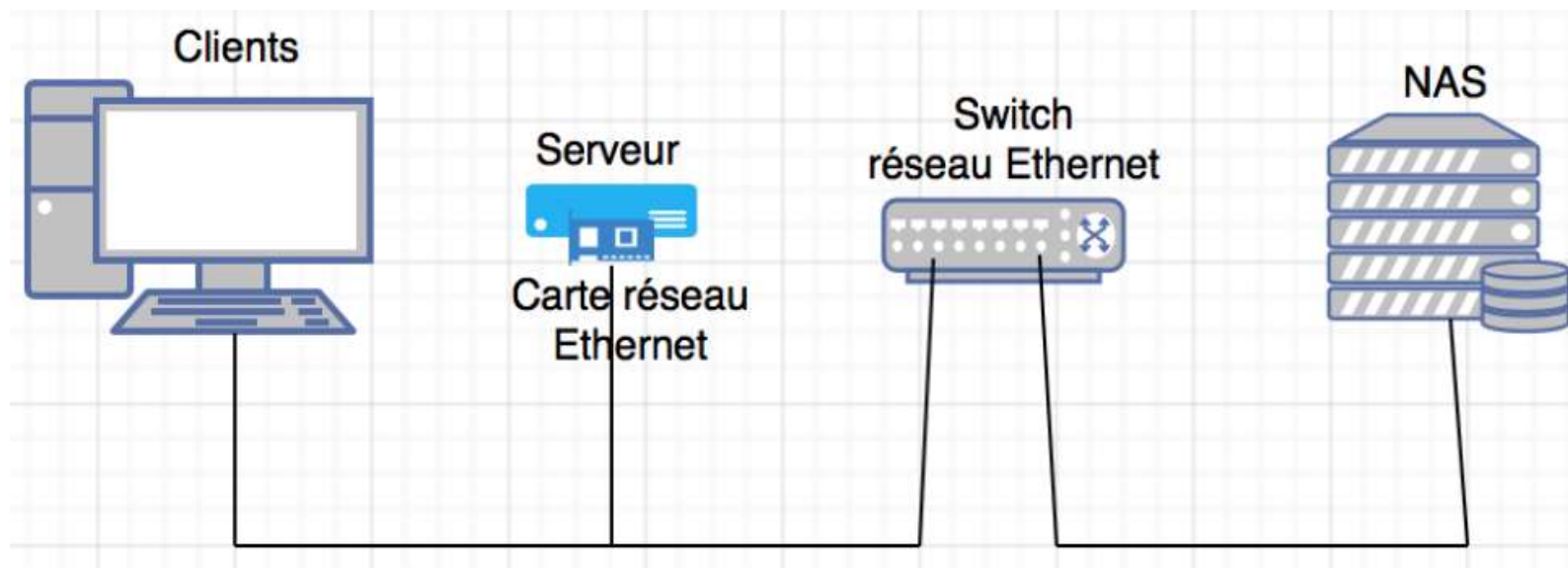
- 1) Solution la plus économique, très simple et rapide à mettre en place
- 2) Architecture est envisageable en environnement de production mais il faut penser à une solution de sauvegarde adaptée
- 3) Solution idéale pour des petites et moyennes entreprises.

INCONVENEINTS :

- 1) Les fonctionnalités évoluées telles que VMotion, HA, ne sont pas disponibles
- 2) Pas de gestion centralisée des VM

NAS (Network Attached Storage)

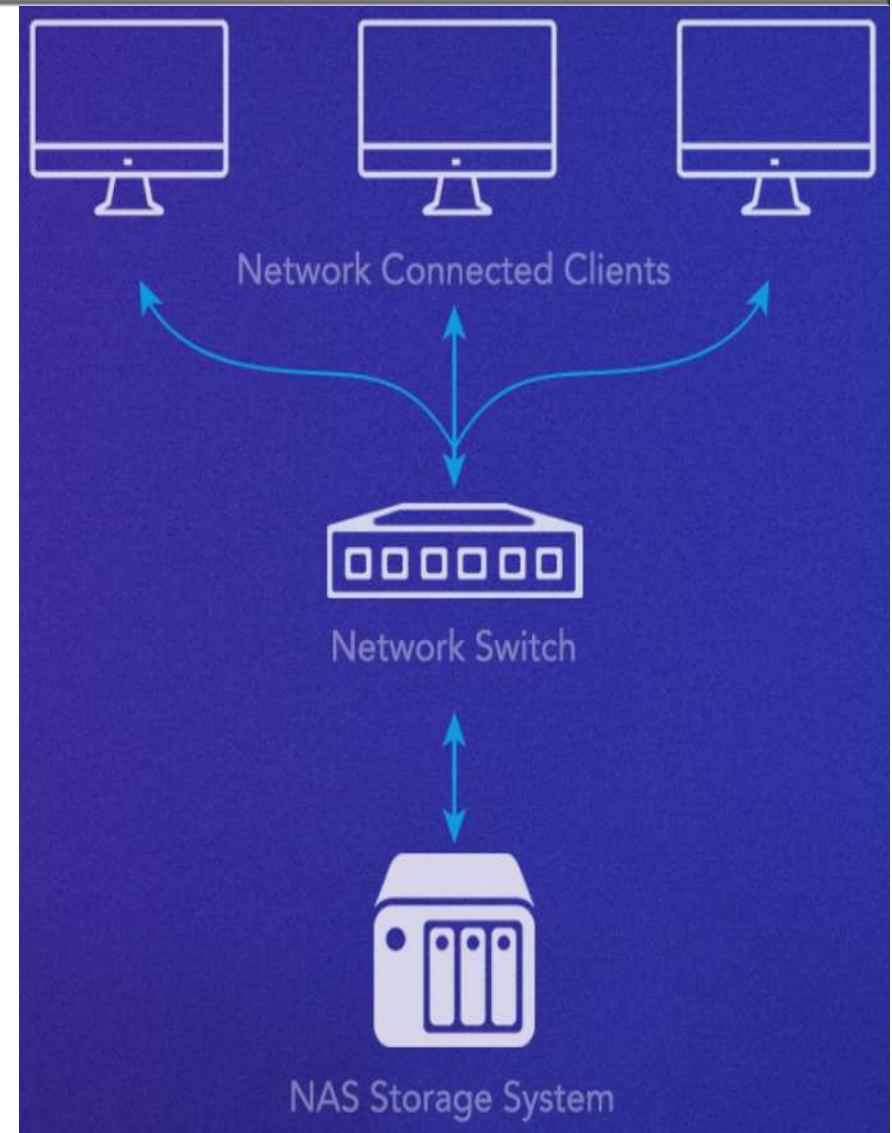
Le stockage en réseau (NAS) est un type de périphérique de stockage de fichiers dédié qui fournit aux noeuds de réseau local (LAN) un **stockage partagé basé sur des fichiers** via une connexion Ethernet standard



- ☐ Les périphériques NAS sont généralement dépourvus de clavier ou d'écran, ils sont configurés et gérés avec un programme utilitaire basé sur un navigateur
- ☐ Chaque NAS réside sur le réseau local en tant que noeud de réseau indépendant et possède sa propre adresse IP

NAS (Network Attached Storage)

- ❑ Les périphériques NAS intègrent le support de multiples systèmes de fichiers réseau, tels que :
 - **Common Internet File System (CIFS)** protocole de partage de Microsoft et de Samba
 - **Network File System (NFS)** qui est un protocole de partage de fichiers Unix,
 - **AFP (AppleShare File Protocol)** qui est l'équivalent pour la technologie Apple.



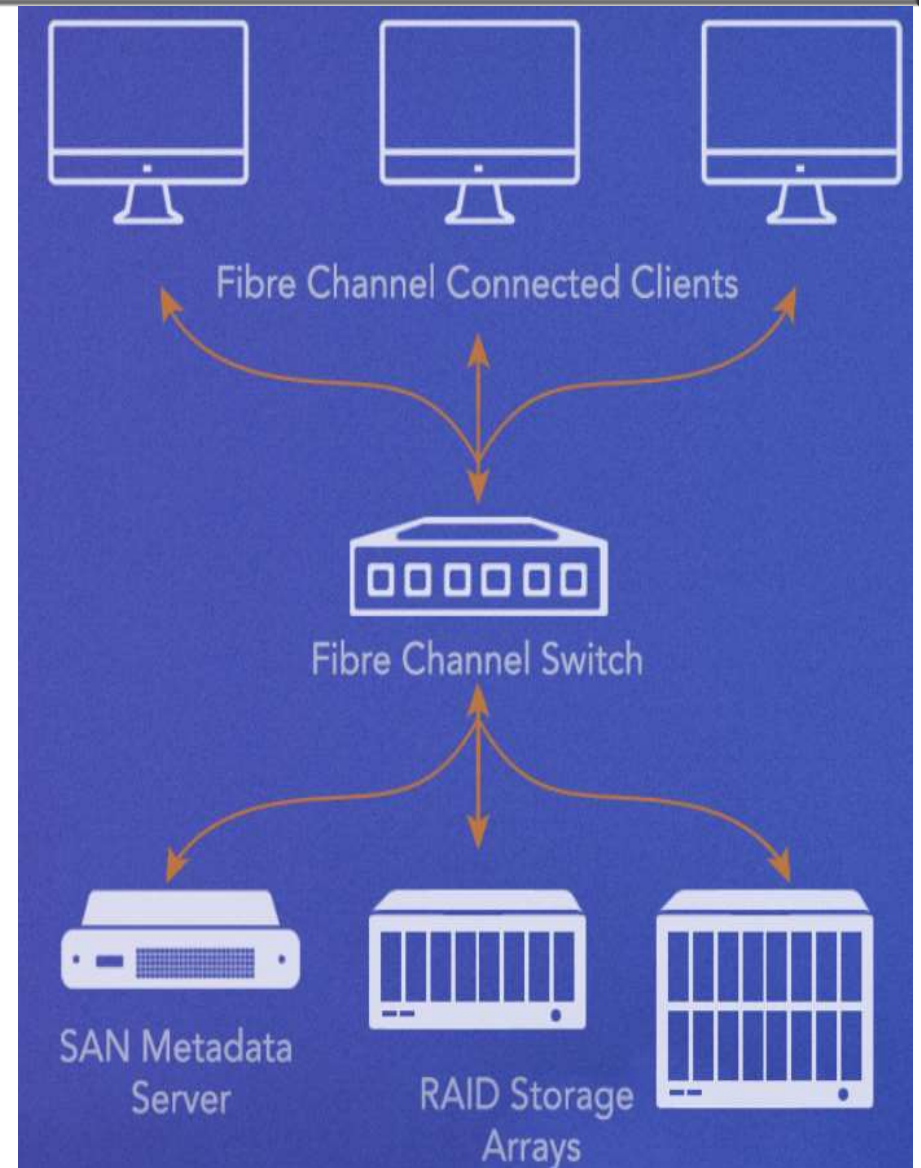
Exemple : Synology



NAS with eight drive bays for 3.5" disk drives.

SAN (Storage Area Network)

- ❑ Un réseau de stockage (SAN) est un réseau dédié permettant de mutualiser les ressources de stockage.
- ❑ VMs sont stockées sur la baie de stockage et sont accessibles par plusieurs serveurs.
- ❑ Si un serveur tombe en panne, un autre serveur connecté au stockage peut accéder aux VM et les **remettre rapidement et automatiquement en production** sans avoir à attendre que le serveur en panne soit de nouveau opérationnel .



SAN (Storage Area Network)

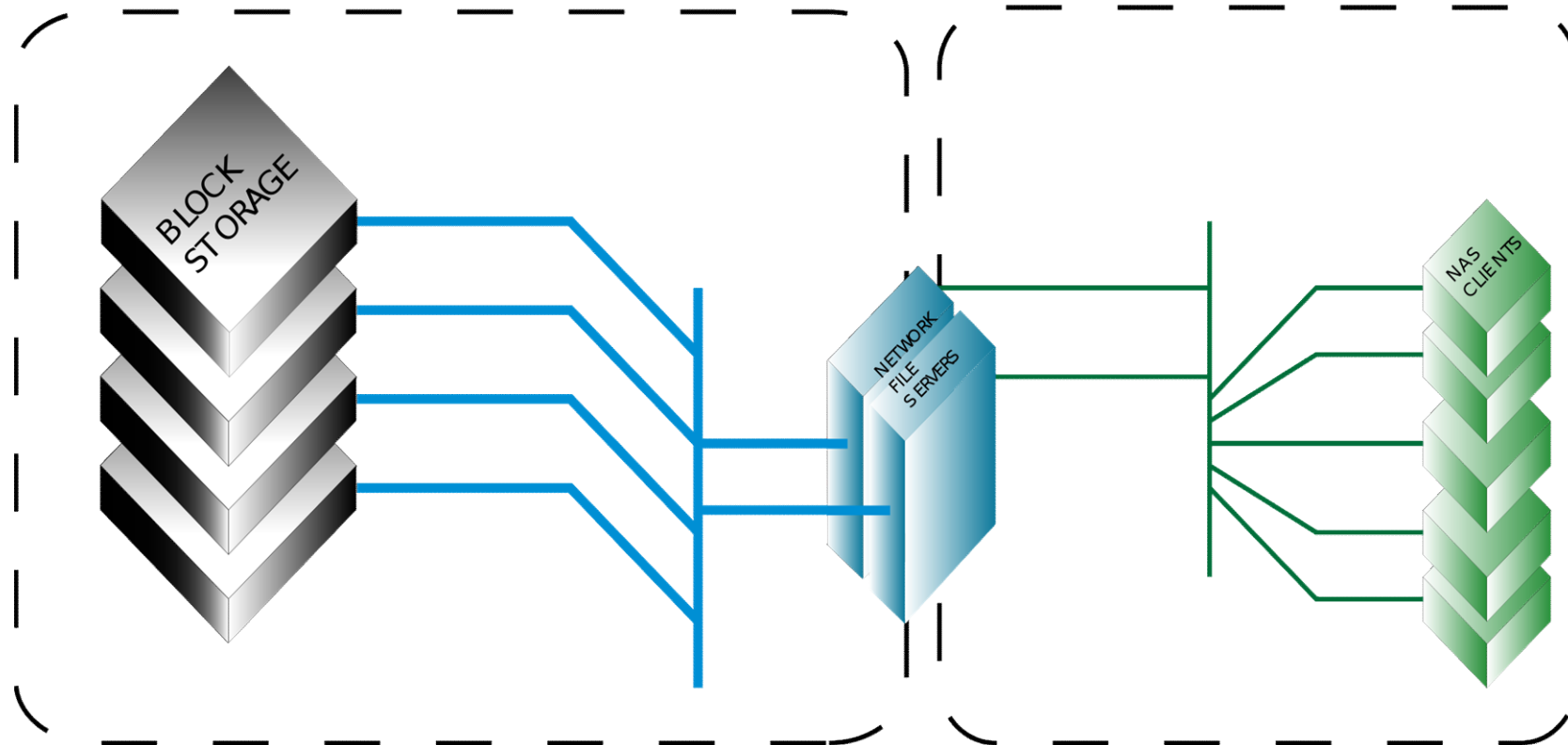
- ☐ Il se différencie des autres systèmes de stockage par un accès aux disques en **mode bloc**. Les baies de stockage **n'apparaissent pas comme des volumes partagés sur le réseau**
- ☐ Les baies sont directement accessibles en mode bloc par le système de fichiers des serveurs par introduction d'un niveau d'abstraction entre le serveur et le système de stockage
- ☐ Chaque serveur voit l'espace disque d'une baie SAN auquel il a accès comme son propre disque dur
- ☐ Des protocoles les utilisés pour le transit des données sur le réseau SAN sont Fibre Channel et iSCSI (Internet Small Computer System Interface)

SAN

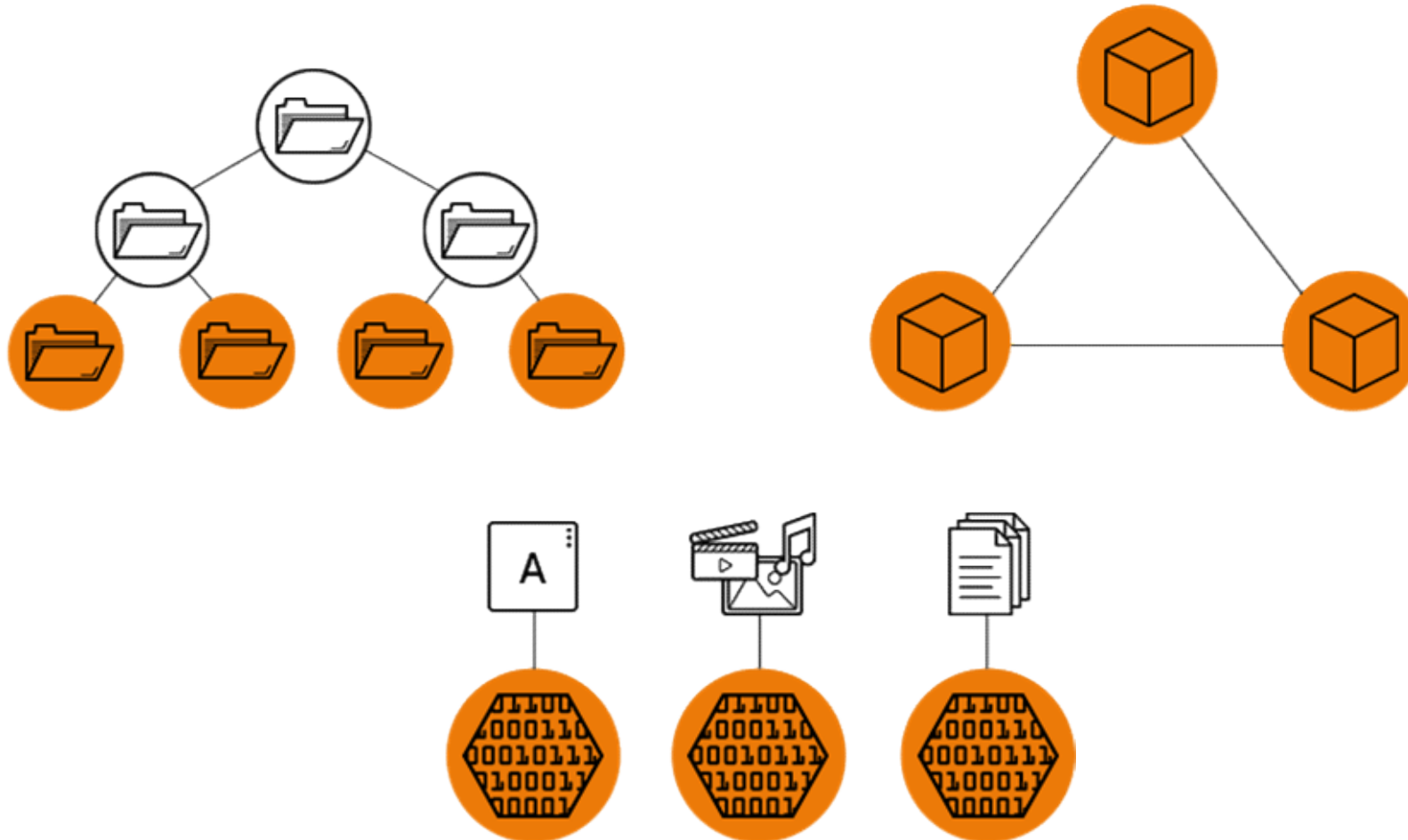
Fibre Channel, iSCSI, or AoE

NAS

SMB, NFS, AFS



Mode de stockage



TP N°2
CONFIGURATION ET PRISE EN MAIN DES MACHINES
VIRTUELLES SOUS VMWARE WORKSTATION Pro