

# Cours : Architecture Logicielle

## Chapitre : Couche infrastructure

### Introduction :

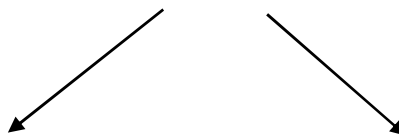
Après le traitement de la couche applicative, il est temps de regarder de plus près les infrastructures permettant de supporter les flux, gisements de données et autres middlewares identifiés.

### Qu'est-ce que la couche infrastructure ?

La **couche infrastructure** traite des **mécanismes sous-jacents** qui proposent des ressources et garantissent la **performance**, la **disponibilité** et la **protection** des données. On parle ici de **serveurs**, de **réseau**, de **stockage**, de **virtualisation**, de **sauvegarde**, d'**archivage**, de cloud bien sûr.

Les interlocuteurs principaux seront ici les **responsables des infrastructures** et les **développeurs**.

On a deux cas d'infrastructure



#### Les infra mutualisées

Utilisées par le système que vous êtes en train de bâtir, mais aussi par d'autres systèmes, sur lesquels on va zoomer au niveau des spécificités permettant de répondre aux exigences identifiées dans l'analyse fonctionnelle.

#### Les infra dédiées

Uniquement utilisées par le système considéré, qui doivent être pensées et décrites complètement dans le cadre de cette méthode.

On va se concentrer sur le premier cas, **les infrastructures mutualisées**, car il est le plus représentatif du quotidien. En effet, on monte rarement, par exemple, un réseau dédié pour un projet. Le Cloud renforce cette tendance en nous poussant de plus en plus à utiliser des **services unitaires** sans parfois pouvoir expliciter comment sont construits les systèmes sous-jacents.

#### → Le calcul

La partie calcul : serveur, machine virtuelle, conteneur.

Au niveau infrastructure, on va s'attacher à décrire 3 éléments :

- **Le dimensionnement**—quel niveau de performance requis ? Nombre de cœurs versus fréquence GHz ? Taille mémoire ? Nombre de ports réseau et/ou SAN (Storage Area Network) ?
- **L'emplacement**—Quel Datacenter ? Quel fournisseur Cloud ?
- **L'identifiant**—Qui permet de faire le lien avec la couche applicative.

On peut également mettre en avant les mécanismes de **disponibilité** (clustering OS, VMware HA, zone de disponibilité cloud...)

De nos jours, avec l'auto-scaling, on ne parle plus de dimensionnement, c'est fonctionnalité de cloud qui permet aux utilisateurs de faire évoluer automatiquement les services cloud. Comme les machines virtuelles (VM) et les capacités des serveurs

## → Le réseau

Dans un deuxième temps, il peut être pertinent d'analyser la partie réseau :

- Comment mes composants sont-ils **interconnectés** ?
- Comment mes clients **communiquent**-ils avec les composants (que ce soit de façon physique ou virtuelle) ?
- Quel **débit** (1G, 10G ?), quelle latence ?
- Quel **type de lien** (RJ45, SFP, vAdapter ?) ?
- Quelle **redondance** (LACP, interfaces multiples...) ?

On générale on travaille sur des architectures **hybrides**, utilisant à la fois des **infrastructures** du **Datacenter** et du **cloud**. Dans ce cas, la méthode **d'interconnexion** est capitale—que ce soit via internet ou des liens dédiés type Azure **ExpressRoute**(est un service Azure qui vous permet de créer des connexions privées entre les centres de données Microsoft et une infrastructure locale) ou **AWS**(Amazon Web Services) **DirectConnect** (un service cloud qui associe votre réseau directement à AWS pour fournir des performances constantes et à faible latence.), il faut bien

prendre en compte les débits et latences générés. Cela peut permettre d'affiner l'architecture, en choisissant par exemple de placer tel ou tel service plus sensible aux contraintes réseau d'un côté ou de l'autre.

## → Le stockage

Un dernier élément qu'il faut analyser est la **partie stockage**.

- Comment **j'accède à mon stockage** (SAN ? NAS ? FC ? IP ? attachement direct ? Local ?)
- Quelle **forme** de stockage (bloc, fichier, objet) ?
- Quel **type de disque** utilisé (SSD vs HDD, Full Flash vs Hybride) ?
- Plus généralement les **performances** délivrées (latence, débit, nombre d'E/S..) ?

La **localisation** du stockage va également être un enjeu important afin de pouvoir évaluer les risques de sinistre, dans une approche de PRA (Plan de Reprise d'Activité).

C'est également l'occasion de détailler les **fonctionnalités** particulières : **virtualisation** (permet d'accélérer la mise à disposition des serveurs, d'augmenter la disponibilité, d'améliorer la reprise après incident.), **chiffrement** (c'est par exemple une fonctionnalité des Azure Storage Account, une solution de stockage cloud de Microsoft), réplication synchrone et asynchrone, ...

On trouve aujourd'hui une tendance à la création de **silos de stockages dédiés** à des cas d'usages bien précis (big data capacitif, application critiques haute performances...)

La **couche infrastructure** est aussi l'occasion d'analyser et de détailler les différentes **machines physiques ou virtuelles** dédiées à des usages spécifiques, comme par exemple un système d'archivage (support physique à écriture unique (WORM)).

Pour conclure **La couche infrastructure** permet à un système logiciel d'interagir avec des **systèmes externes** en recevant, **stockant et fournissant des données** sur demande.

La couche d'infrastructure est **une couche importante, bien que facultative**, dans un système logiciel, en particulier pour les systèmes qui ont besoin **d'interagir** régulièrement avec des **systèmes externes**.