



SYSTÈMES ET RÉSEAUX

LICENCE MATHÉMATIQUE
2022 - 2023

UNIVERSITÉ PARIS 8

OUMAIMA EL JOUBARI

ELJOUBARI.OUN@GMAIL.COM

PLAN DU COURS

I. Réseaux informatiques:

- 1) Introduction
- 2) Fonctionnement des réseaux
- 3) Couche physique
- 4) Couche Liaison de données
- 5) Couche réseau
- 6) Couche transport
- 7) Couches supérieures

II. Programmation réseau:

- 1) Adresses IP & MAC
- 2) Protocoles
- 3) Services et ports
- 4) Threads et Sockets

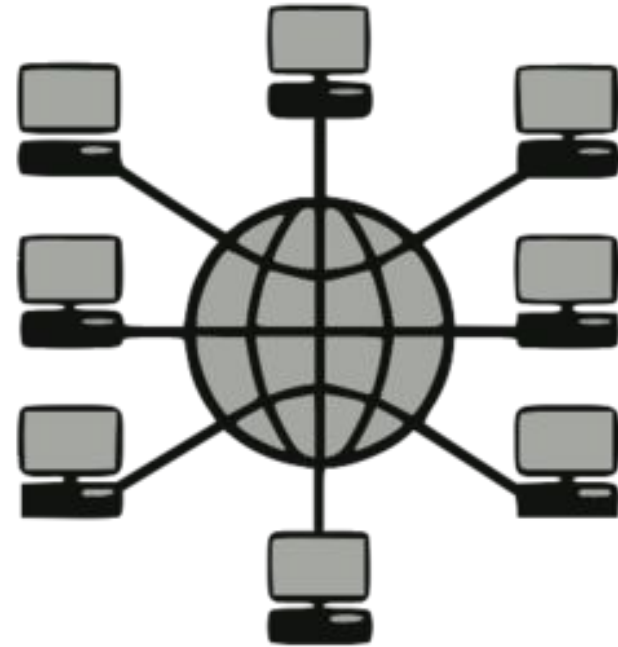
I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

A. Qu'est ce qu'un réseau de communication ?

B. Quelles sont les utilisations des réseaux ?

C. Quels sont les types des réseaux ?



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

A. Qu'est ce qu'un réseau de communication ?

Un réseau informatique (Computer network) est un ensemble de matériels et de logiciels permettant à plusieurs machines (ordinateurs au sens large) de communiquer entre elles.

B. Quelles sont les utilisations des réseaux?

- Faciliter la communication (email, messages instantanés, téléphone...);
- Partage des ressources matérielles (imprimantes, scanners...);
- Partage des fichiers, données et informations;
- Partage des logiciels.

I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

La classification des réseaux informatiques se base sur quatre critères:

- La couverture géographique du réseau
- Le support utilisé pour transporter les données
- La topologie du réseau
- L'architecture du réseau (méthode de gestion)

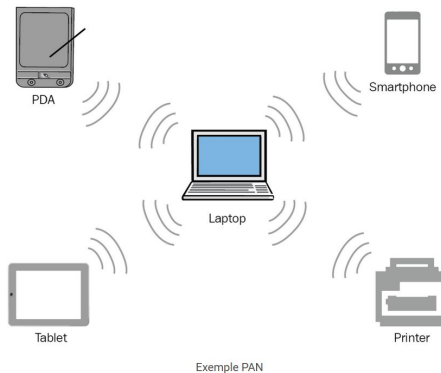
I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

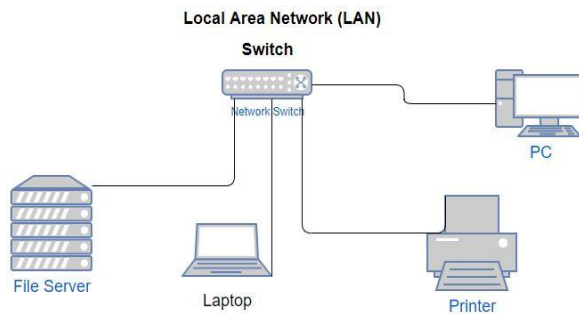
C. Quels sont les types des réseaux?

- La classification par étendue de la couverture géographique:

PAN (Personal Area Network)



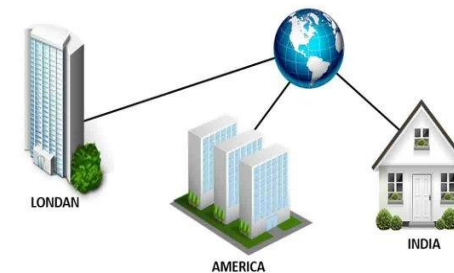
LAN (Local Area Network)



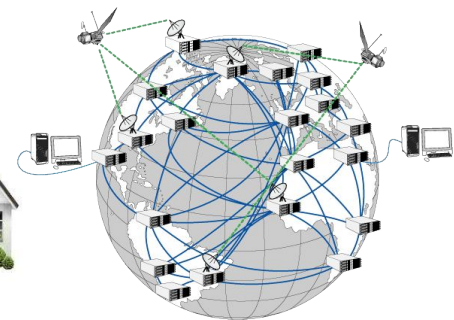
MAN (Metropolitan Area Network)



WAN (Wide Area Network)



Internet



Réseaux de petite taille:

- Interconnexion de quelques équipements,
- Distance de quelques mètres.

Réseaux de moyenne taille:

- Interconnexion de quelques machines,
- Les équipements se situent dans le même bâtiment/certaines mètres.

Réseaux de grande taille:

- Interconnexion de plusieurs LANs,
- Les équipements se situent dans différents bâtiments.

Réseaux de très grande taille:

- Interconnexion de plusieurs LANs,
- Interconnexion à l'échelle d'une ville, un pays, ou le monde entier.

Réseau universel:

- Plus grand WAN.

I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

■ D'autres réseaux:

Intranet

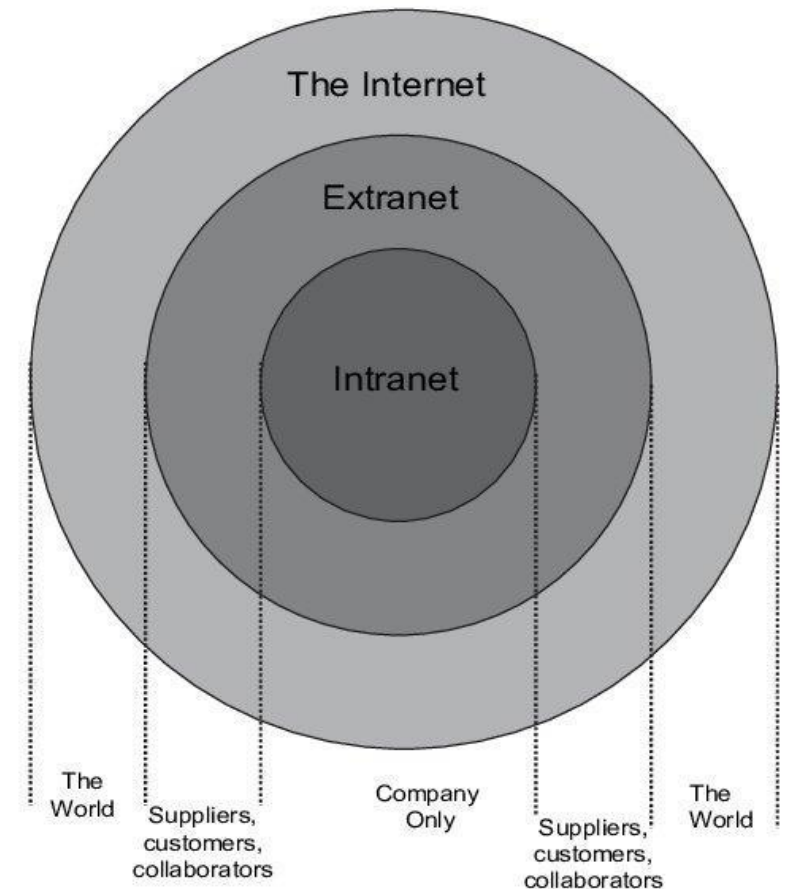
Ensemble d'ordinateurs privés reliés entre eux via un réseau de communication.

Ex: Intranet de l'université Paris 8

Extranet

un réseau qui permet un accès contrôlé à des systèmes ou utilisateurs externes.

Ex: Services bancaires sur Internet



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

- La classification selon le support utilisé pour transporter les données:

Transmission filaire

- Câbles réseaux;
- Fibre optique.

Transmission sans fil

- Infrarouge (IR);
- Ondes;
- WiFi;
- Bluetooth;
- Etc.

I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

■ La classification par topologie:

- Bus
- Étoile
- Anneau
- Maillée
- Arbre
- Hybride

I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

■ La classification par topologie:

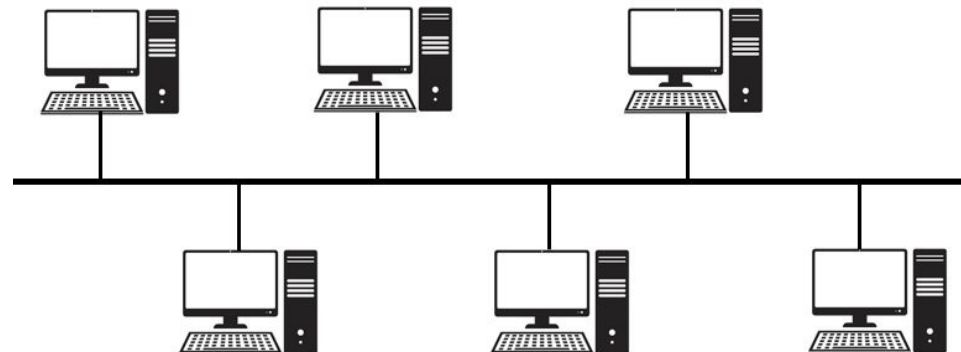
- Bus
- Étoile
- Anneau
- Maillée
- Arbre
- Hybride

Topologie en bus

Cette topologie consiste à relier chaque ordinateur à un bus par l'intermédiaire d'un câble coaxial.

Avantages: simple à déployer.

Défauts : Vulnérabilité en cas de panne, réseau très lent.



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

■ La classification par topologie:

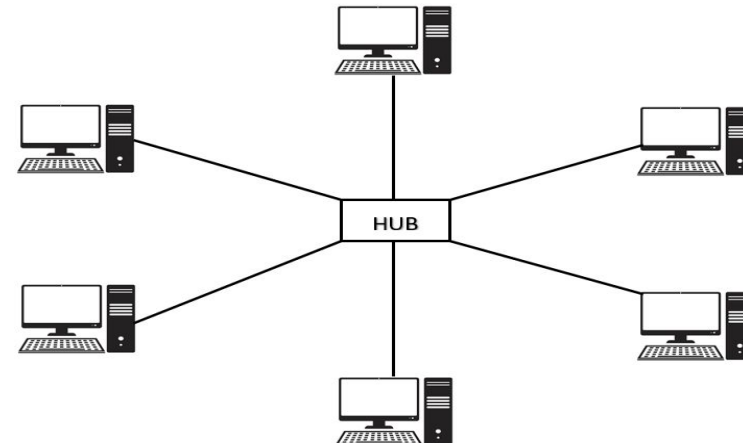
- Bus
- Étoile
- Anneau
- Maillée
- Arbre
- Hybride

Topologie en étoile

Tous les nœuds du réseau sont connectés à un point central.

Avantages: chaque nœud est indépendant, réseau extensible.

Défauts : Coût de câblage plus élevé, la défaillance du nœud central impacte tout le réseau .



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

■ La classification par topologie:

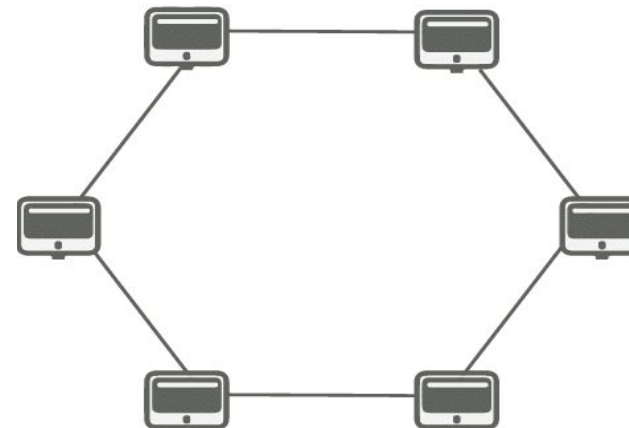
- Bus
- Étoile
- Anneau
- Maillée
- Arbre
- Hybride

Topologie en anneau

Tous les nœuds sont reliés en formant un circuit en boucle.

Avantages: Une rupture de câble peut être facilement contournée dans le cas d'un signal bidirectionnel.

Défauts : Si une machine envoie un message à une autre éteinte le réseau sera bloqué.



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

■ La classification par topologie:

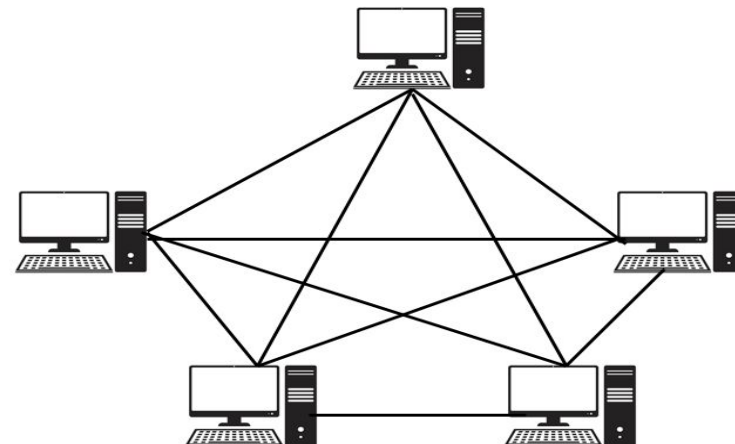
- Bus
- Étoile
- Anneau
- Maillée
- Arbre
- Hybride

Topologie maillée

Chaque nœud est directement lié à plusieurs machines (ou toutes les machines).

Avantages: Machines indépendantes, réseau robuste face aux pannes.

Défauts : Coût de câblage très élevé.



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

■ La classification par topologie:

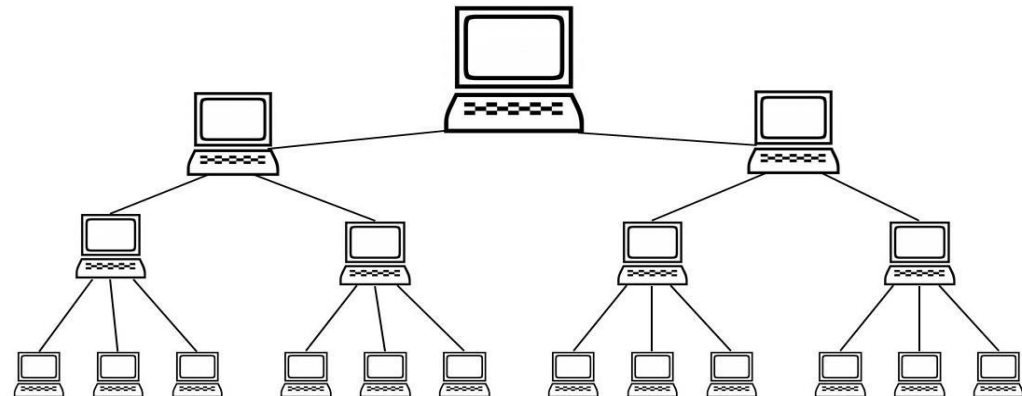
- Bus
- Étoile
- Anneau
- Maillée
- Arbre
- Hybride

Topologie en arbre

Une architecture hiérarchisée où les données remontent l'arborescence puis redescendent.

Avantages: facilite la gestion des grands réseaux.

Défauts: Une panne sur une partie du réseaux touche les nœuds en dessous.



I. RÉSEAUX INFORMATIQUES

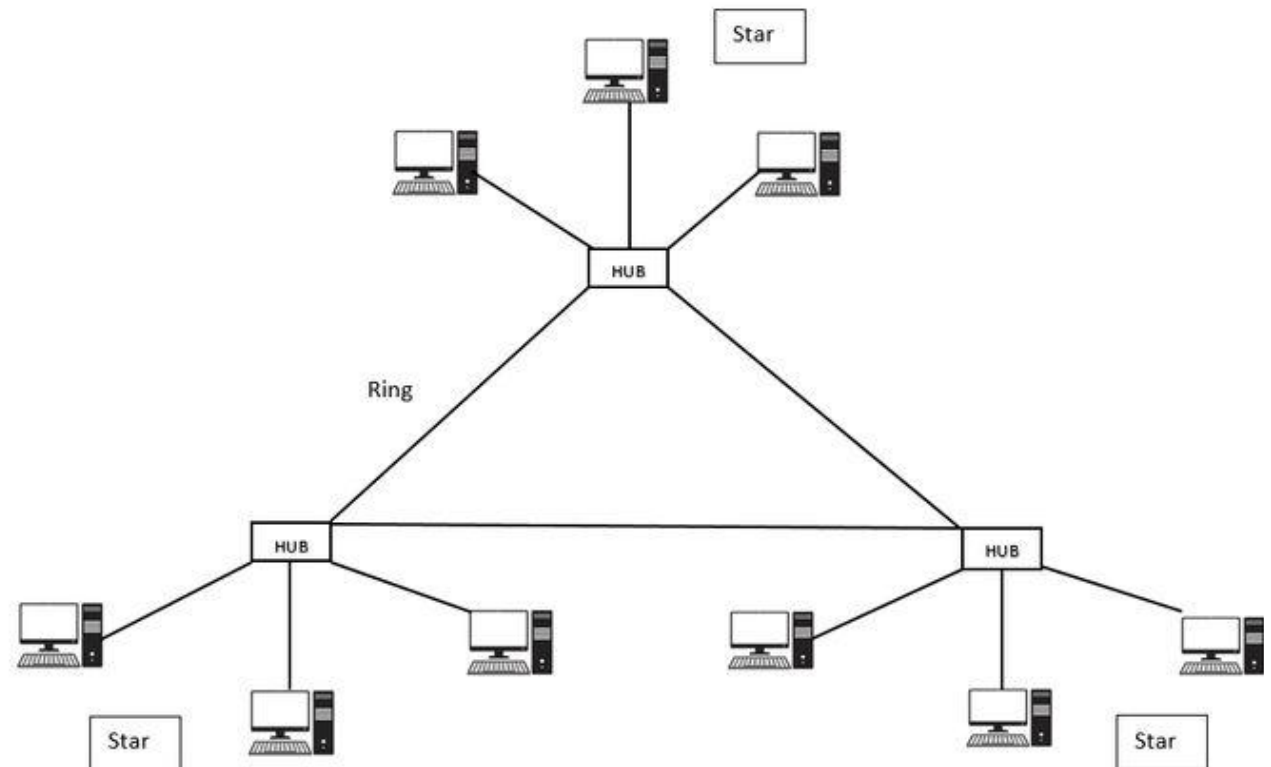
I) INTRODUCTION

C. Quels sont les types des réseaux?

■ La classification par topologie:

- Bus
- Étoile
- Anneau
- Maillée
- Arbre
- Hybride

Topologie hybride



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

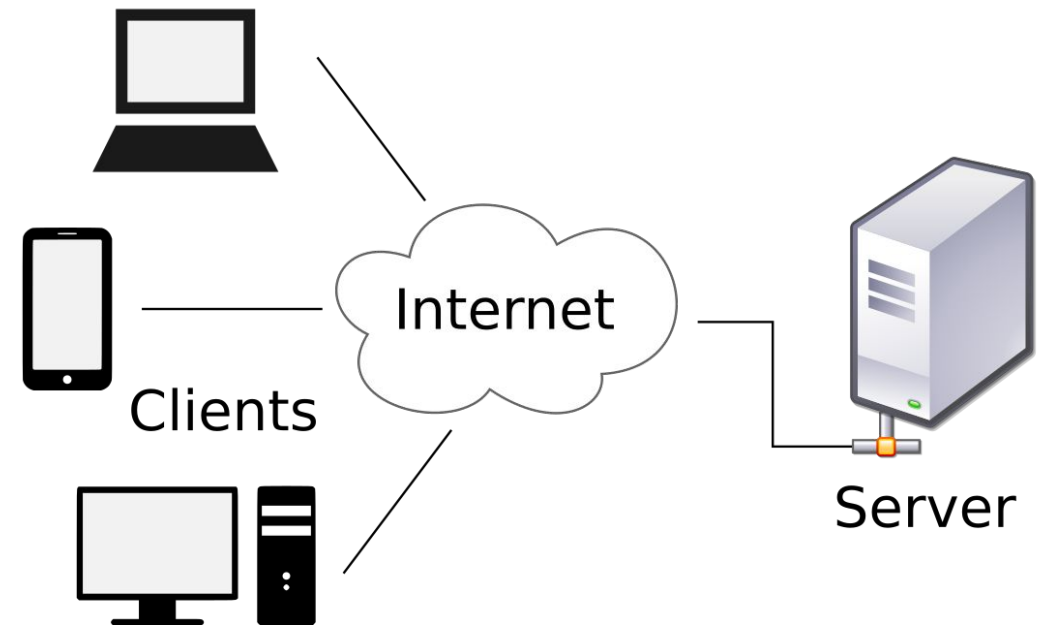
■ La classification par architecture:

○ Client-serveur (Client-server):

Cette architecture décrit le fonctionnement coopératif entre un serveur et un client.

Serveur: une machine qui offre un service sur le réseau. Le serveur reçoit des requêtes, les traite et renvoie une réponse à l'utilisateur.

Client: une machine qui utilise le service offert par un serveur. Le client envoie une requête et reçoit une réponse.



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

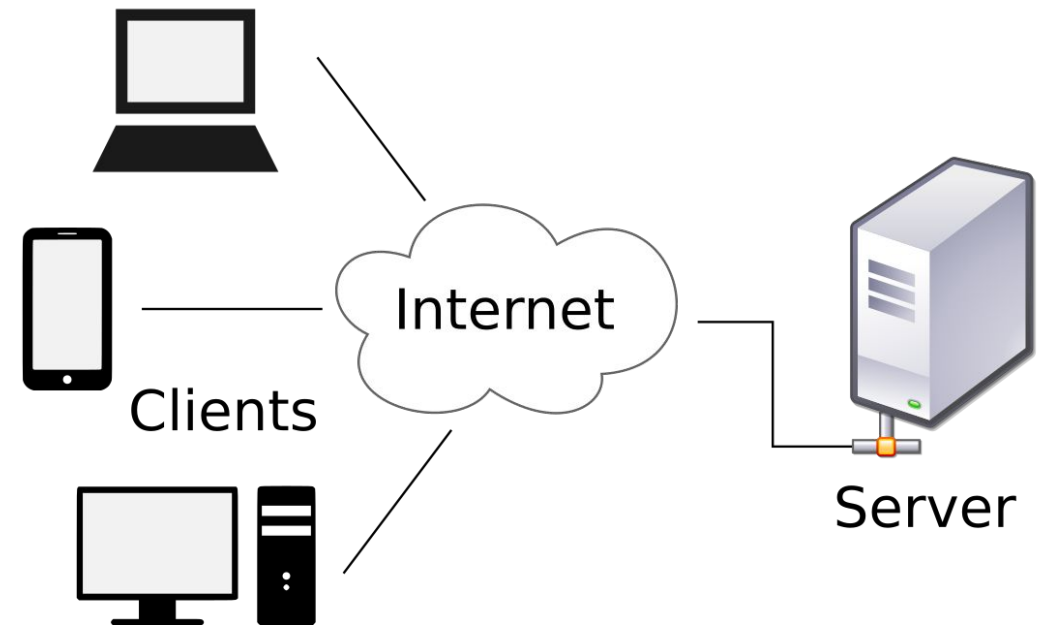
C. Quels sont les types des réseaux?

■ La classification par architecture:

○ Client-serveur (Client-server):

Avantages: Simplicité de fonctionnement, disponibilité immédiate des données, meilleure sécurité.

Inconvénients: Coût élevé.



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

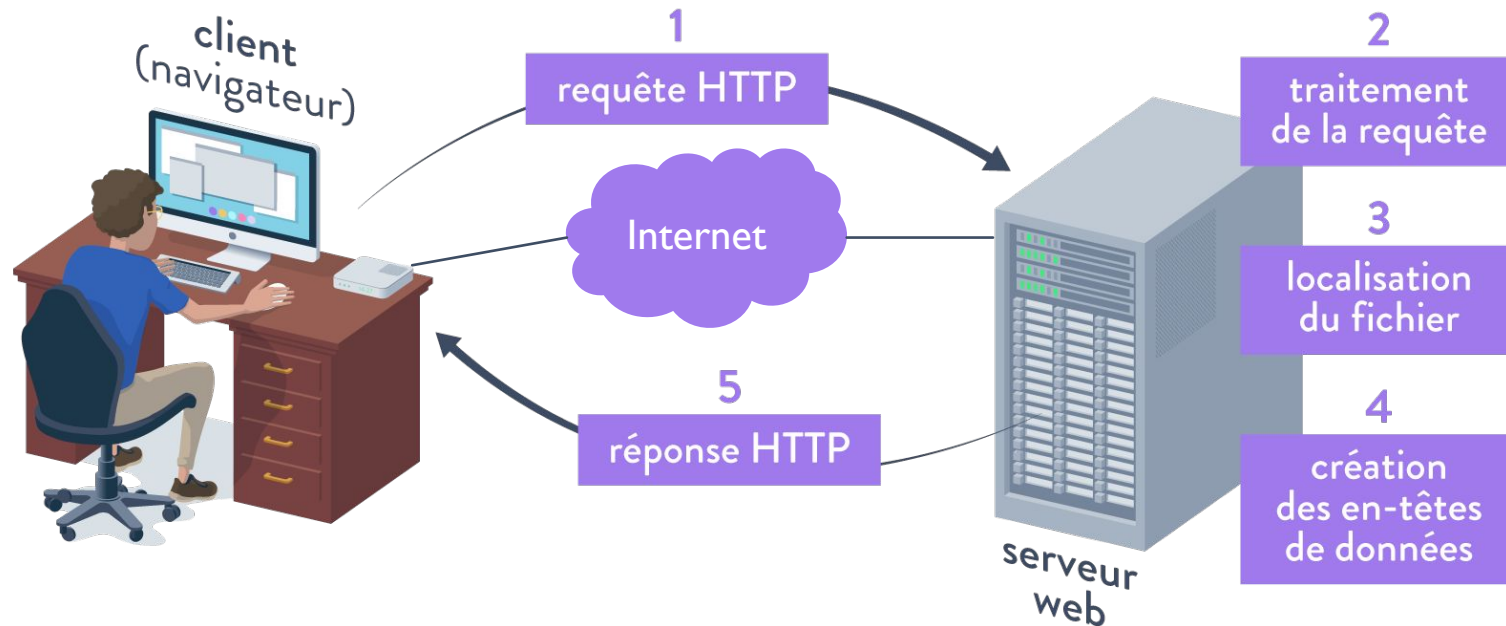
■ La classification par architecture:

○ Client-serveur (Client-server):

Vue de l'utilisateur:

- On clique sur un lien;
- Une page web s'affiche.

Une requête sur le web



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

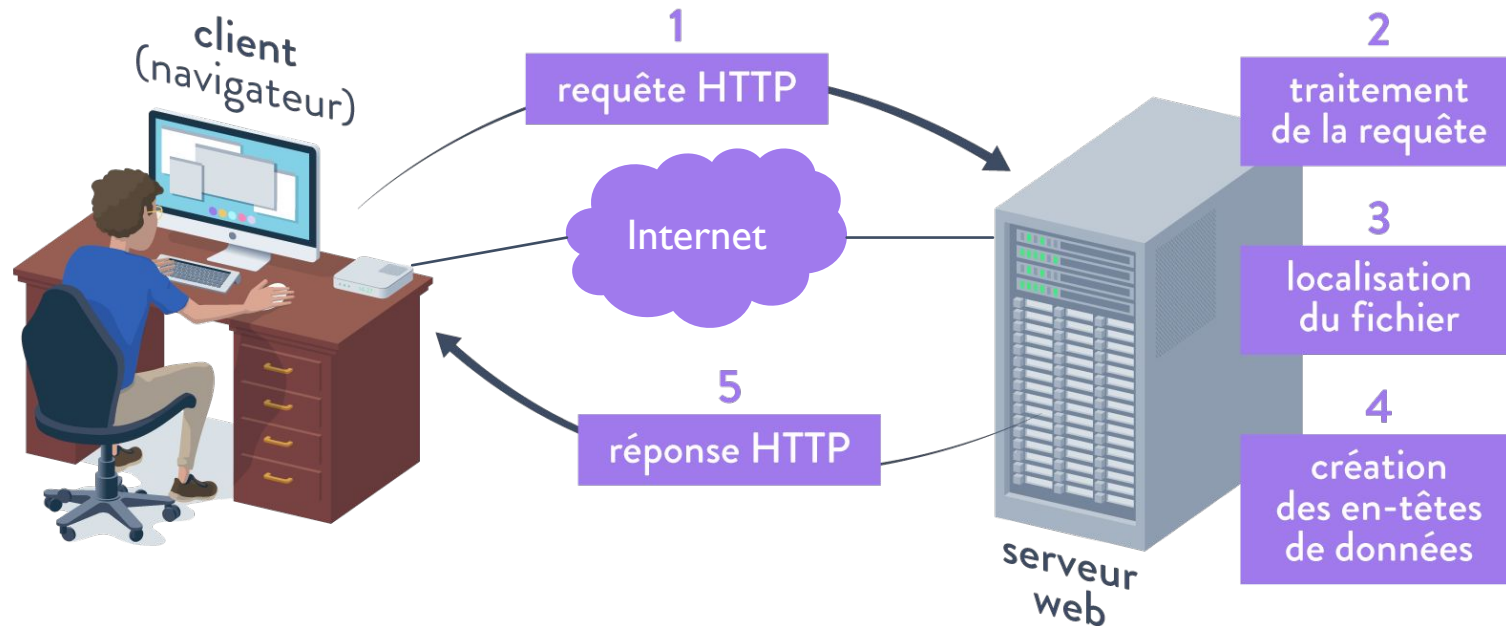
■ La classification par architecture:

○ Client-serveur (Client-server):

Sur la station client:

- Le programme navigateur envoie la requête au serveur associé au lien;
- Le programme “navigateur” reçoit un fichier HTML qu’il affiche sur l’écran.

Une requête sur le web



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

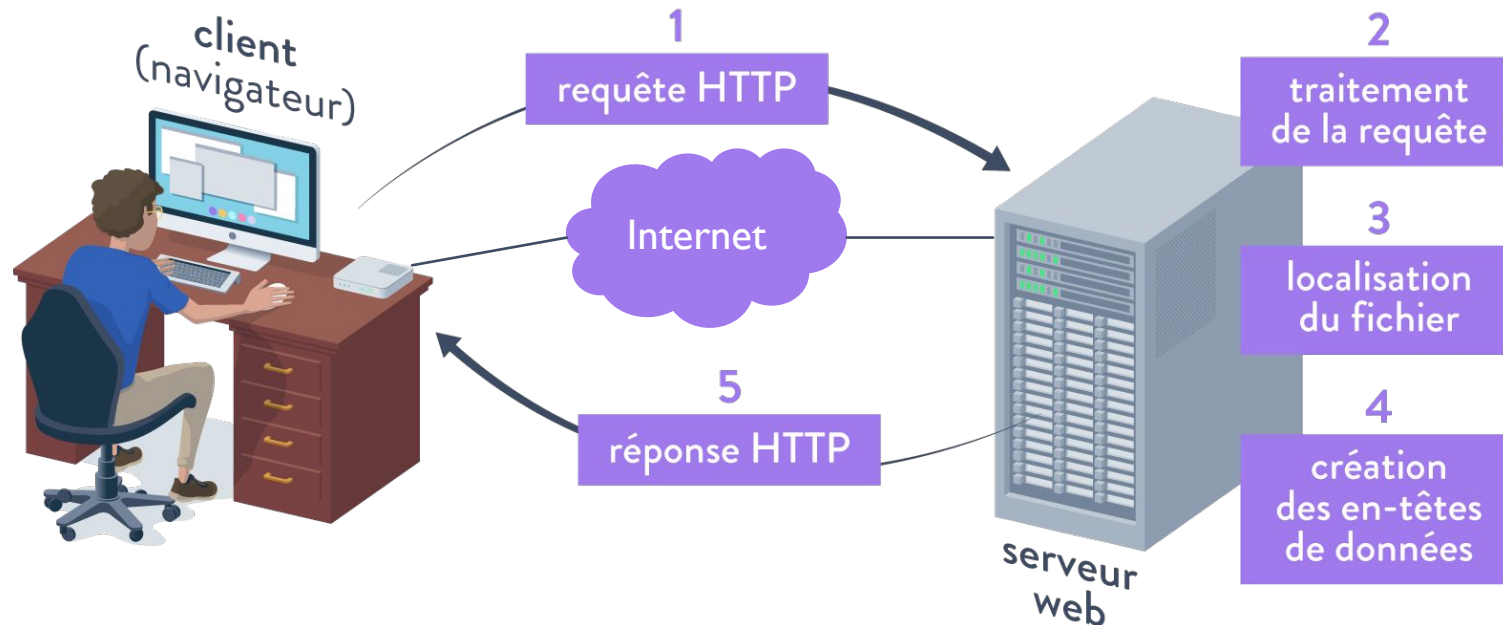
■ La classification par architecture:

○ Client-serveur (Client-server):

Sur le réseau:

- Il faut localiser le serveur ;
- Transporter la requête depuis la station client vers le serveur;
- Transporter les fichiers depuis le serveur vers la station client.

Une requête sur le web



I. RÉSEAUX INFORMATIQUES

I) INTRODUCTION

C. Quels sont les types des réseaux?

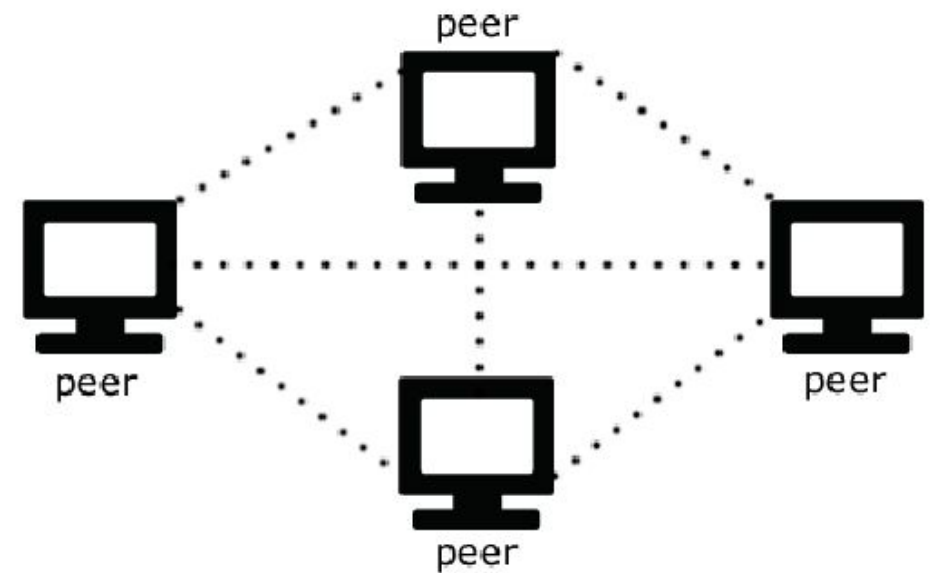
■ La classification par architecture:

○ Poste-à-poste (Peer-to-peer):

Connexion directe entre deux ou plusieurs ordinateurs, où chacun joue à la fois le rôle du client et du serveur.

Avantages: gestion décentralisée de la communication, solution moins coûteuse, aucune hiérarchie entre les utilisateurs, anonymat.

Inconvénients : Faible sécurité, partage illégal des fichiers.



I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

A. Modes de transmission

B. Le transfert de l'information

C. Commutation (switching)

D. Services de transport

E. Modes d'envoi:

F. Notion de protocole

G. Modèles en couches

I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

A. Modes de transmission:

Simplex

- Communication unidirectionnelle;
- Un expéditeur peut envoyer des données mais ne peut pas recevoir;
- Performance réduite;
- Ex: Clavier, moniteur.

Half duplex

- Communication bidirectionnelle mais une à la fois;
- Un expéditeur peut envoyer et recevoir des données mais un à la fois;
- Meilleure performance que le simplex;
- Ex: Talkie-walkie.

Full duplex

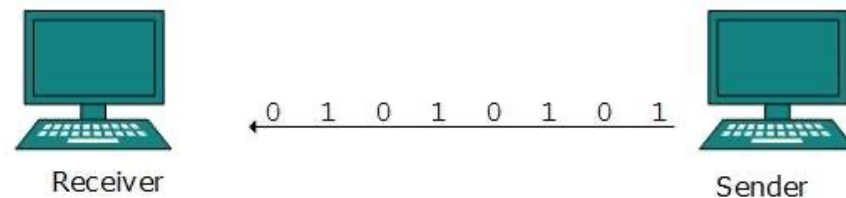
- Communication bidirectionnelle simultanée;
- Un expéditeur peut envoyer et recevoir des données simultanément;
- Meilleure performance car il double l'utilisation de la bande passante;
- Ex: Téléphone.

I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

B. Le transfert de l'information

- Types d'informations transmises par les réseaux informatiques:
 - Données, sons, images, vidéos, ...
- Sur les ordinateurs, les données sont binaires (0 et 1).
- Sur les câbles électriques, fibres optiques ou ondes hertziennes, les données sont des signaux .



I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

B. Le transfert de l'information

- La plus petite quantité d'information : quantum d'information **Bit** (Binary DigiT)
- Le **débit** est la quantité d'information que le réseau peut véhiculer par unité du temps.
- Le temps nécessaire pour transmettre une quantité **Q** de données sur un réseau offrant un débit **D**:

$$T = Q/D$$

- Le débit **D** du réseau est donc:

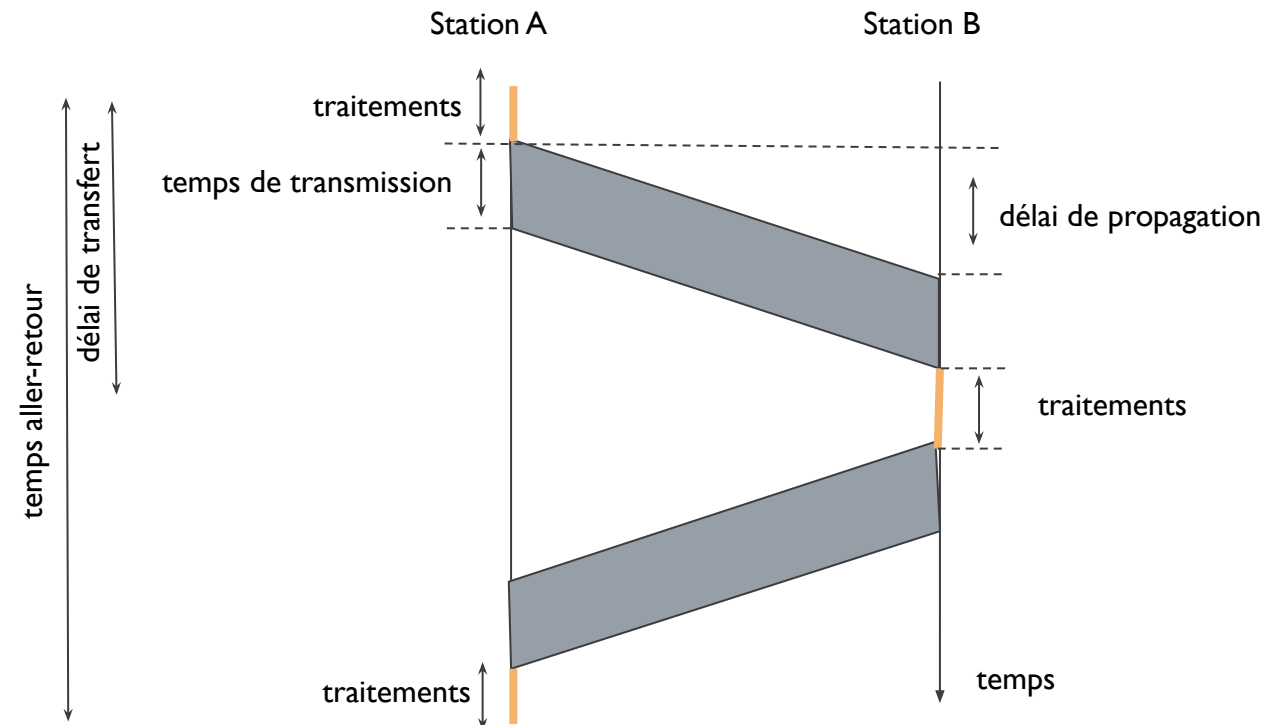
$$D = Q/T$$

I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

B. Le transfert de l'information

- $\text{Délai de transfert} = \text{Temps de transmission} + \text{temps de propagation} + \text{retards (traitements)}$
- Temps de propagation: éloignement, équipements intermédiaires, vitesse de propagation, ...
- Temps d'aller-retour:
 $(\text{transmission} + \text{propagation} + \text{traitements}) * 2$

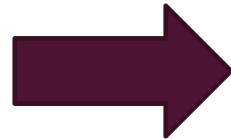


I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

C. Commutation (switching):

La commutation est la manière de faire passer l'information de l'émetteur au récepteur.



- Commutation de circuits
- Commutation de messages
- Commutation de paquets

- Un commutateur (Switch):

C'est un équipement qui relie plusieurs segments (câbles ou fibres) dans un réseau informatique.

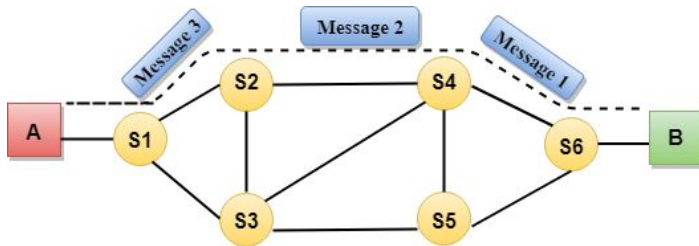


I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

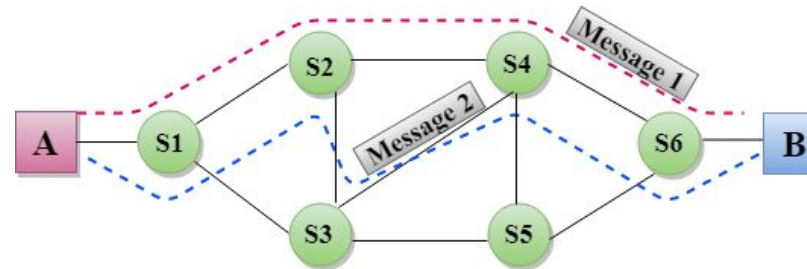
C. Commutation (switching):

Commutation de circuits



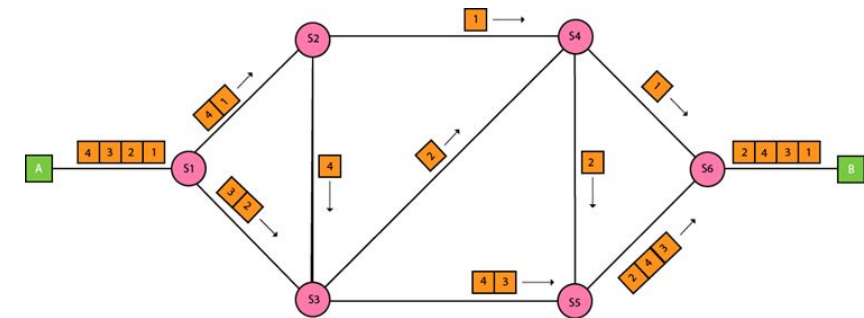
- Un chemin est établi entre les abonnés au début de la communication et est réservé jusqu'à la fin.

Commutation de messages



- Les messages se propagent de nœud en nœud. Chaque nœud conserve le message jusqu'à ce qu'un chemin approprié soit disponible.

Commutation de paquets



- L'information est découpée en paquets qui sont transportés de point en point à l'autre extrémité du réseau.

I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

D. Service de transport:

Service orienté connexion

- Connexion préalable requise;
- Transfert fiable de données;
- Congestion peu probable;
- Retransmission possible de données perdues;
- Les paquets voyagent séquentiellement vers leur nœud de destination et suivent le même itinéraire;
- Retard dans le transfert des informations;
- Allocation de ressources requise.

Service sans connexion

- Connexion non requise;
- Fiabilité non garantie;
- Congestion probable;
- Impossible de retransmettre les données perdues ;
- Les paquets atteignent la destination au hasard sans suivre le même itinéraire;
- Transmission plus rapide;
- Aucune allocation préalable n'est requise.

I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

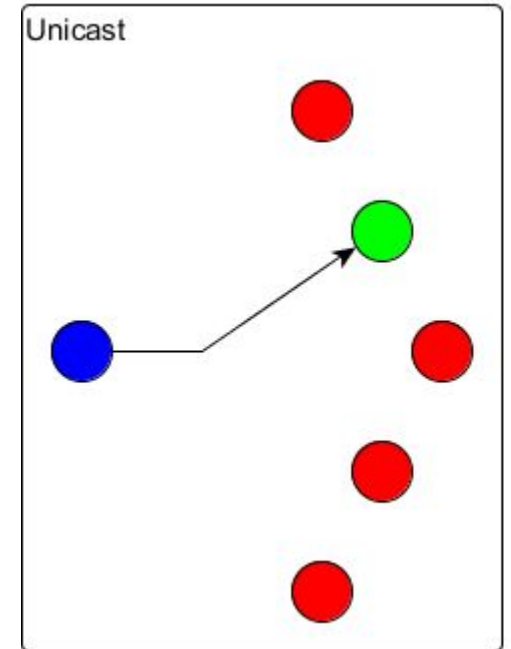
E. Modes d'envoi:

Unicast

Multicast

Broadcast

- Une source + une destination



I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

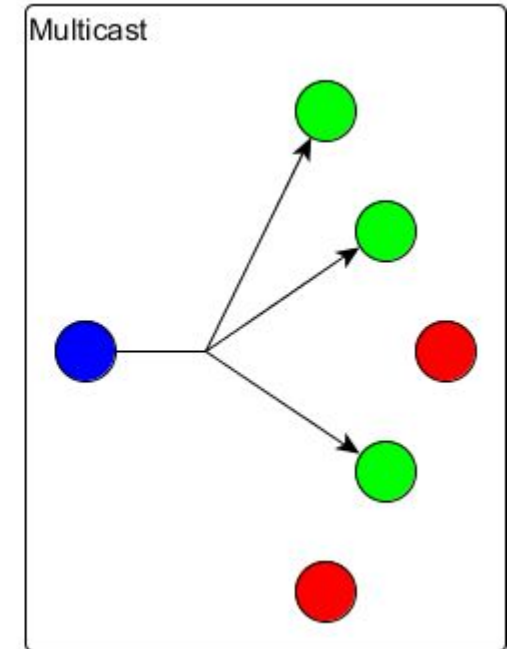
E. Modes d'envoi:

Unicast

Multicast

Broadcast

- Une source + quelques noeuds destination



I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

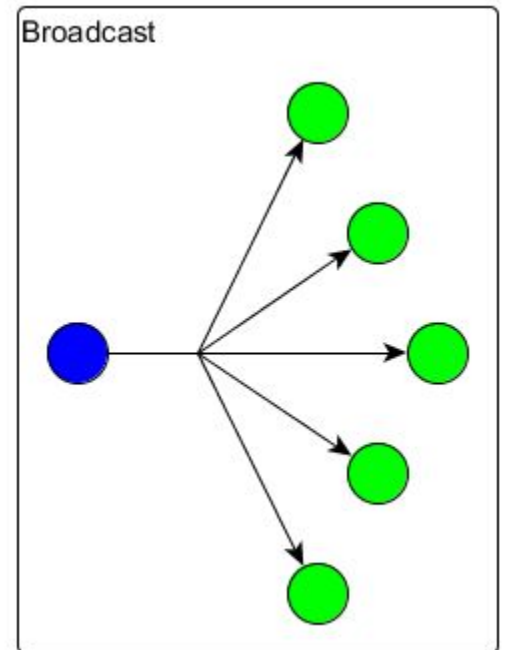
E. Modes d'envoi:

Unicast

Multicast

Broadcast

- Une source + toutes les cibles possibles



I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

F. Notion de protocole:

« For two parties to communicate, they must have a shared conceptual schema; they must have a **common language** or **protocol** and some common understanding about what strings in the language stand for ».

John Day

I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

F. Notion de protocole:

- Qu'est ce qu'un protocole?

Ensemble de conventions définissant les échanges entre des entités qui coopèrent pour réaliser un service.

- Relation entre les protocoles et les interfaces:

Une interface définit l'accès à un service, un protocole définit la réalisation d'un service.

I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches:

- Qu'est ce qu'un modèle en couche?

Décomposition hiérarchique de l'ensemble des mécanismes à mettre en œuvre en une série de couches.

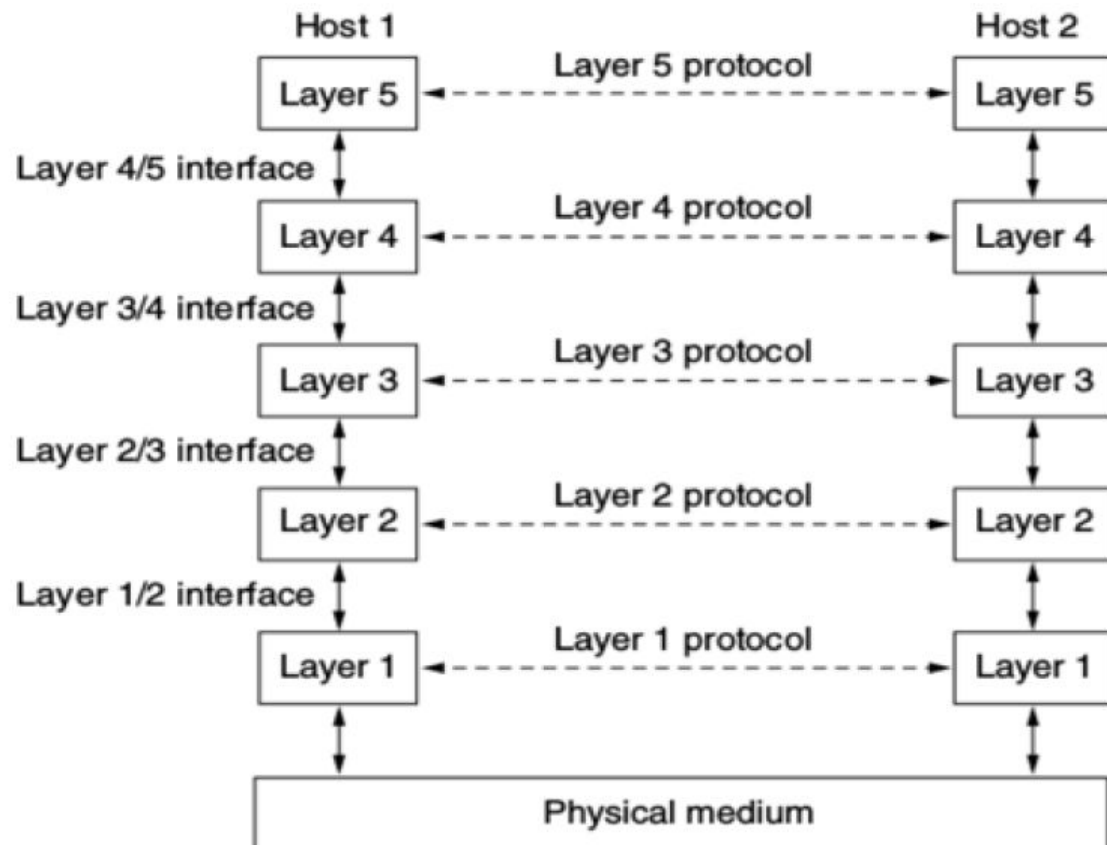
- Quel est l'objectif de ce modèle?

Réduire la complexité de conception des réseaux informatiques.

I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches:



- Une couche est spécialisée dans un ensemble de fonctions particulières. Elle utilise les fonctionnalités de la couche inférieure et propose ses fonctionnalités à la couche supérieure.
- Le protocole d'une couche N: définit l'ensemble des règles ainsi que les formats et la signification des objets échangés, qui régissent la communication entre les entités de la couche N.
- Le service d'une couche N définit l'ensemble des fonctionnalités possédées par la couche N et fournies aux entités de la couche N+1 à l'interface N/N+1.

I. RÉSEAUX INFORMATIQUES

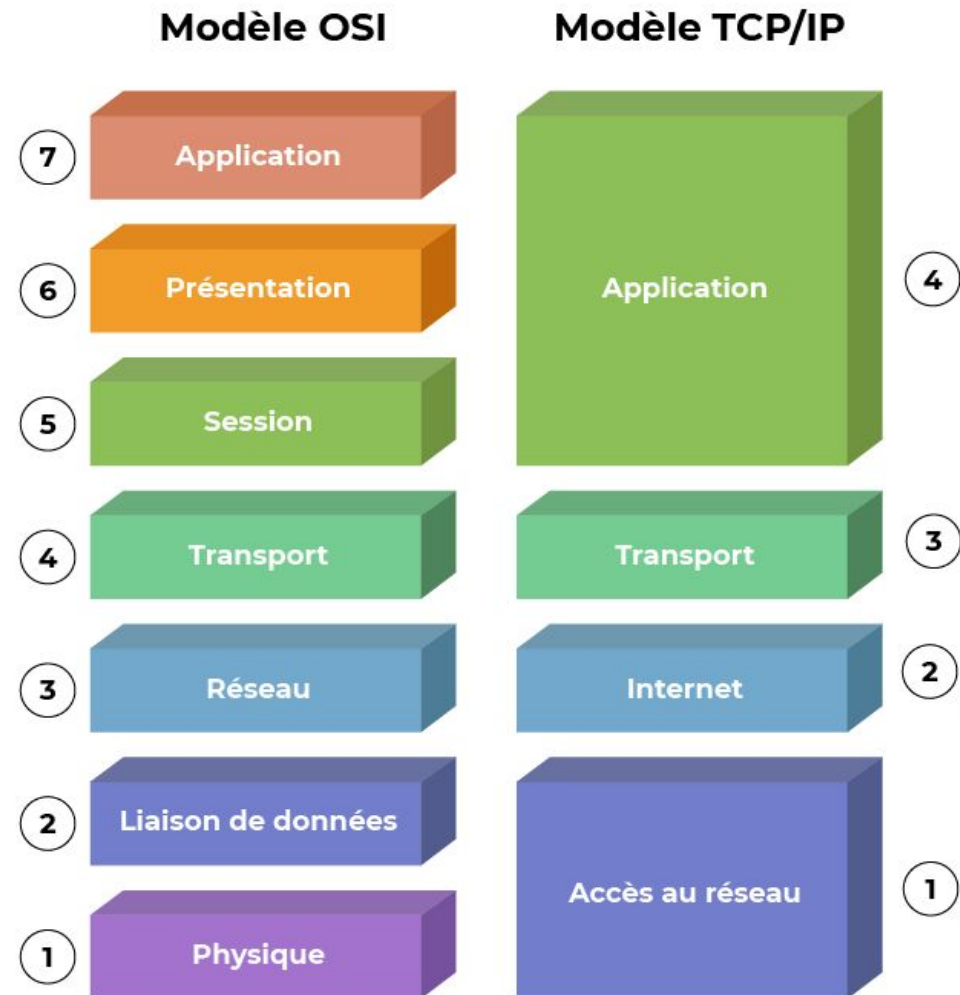
2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches:

- **OSI (Open Systems Interconnection):** Un modèle théorique qui décrit la manière dont les différents composants logiciels et matériels impliqués dans une communication en réseau se répartissent les tâches et interagissent les uns avec les autres.
- **TCP/IP:** Un approche pratique utilisée comme modèle de réseau de référence pour Internet.

TCP: définit la manière dont des applications créent des canaux de communication fiables à l'échelle du réseau.

IP: définit la manière dont les ordinateurs peuvent se transmettre des données via un ensemble routé de réseaux interconnectés.



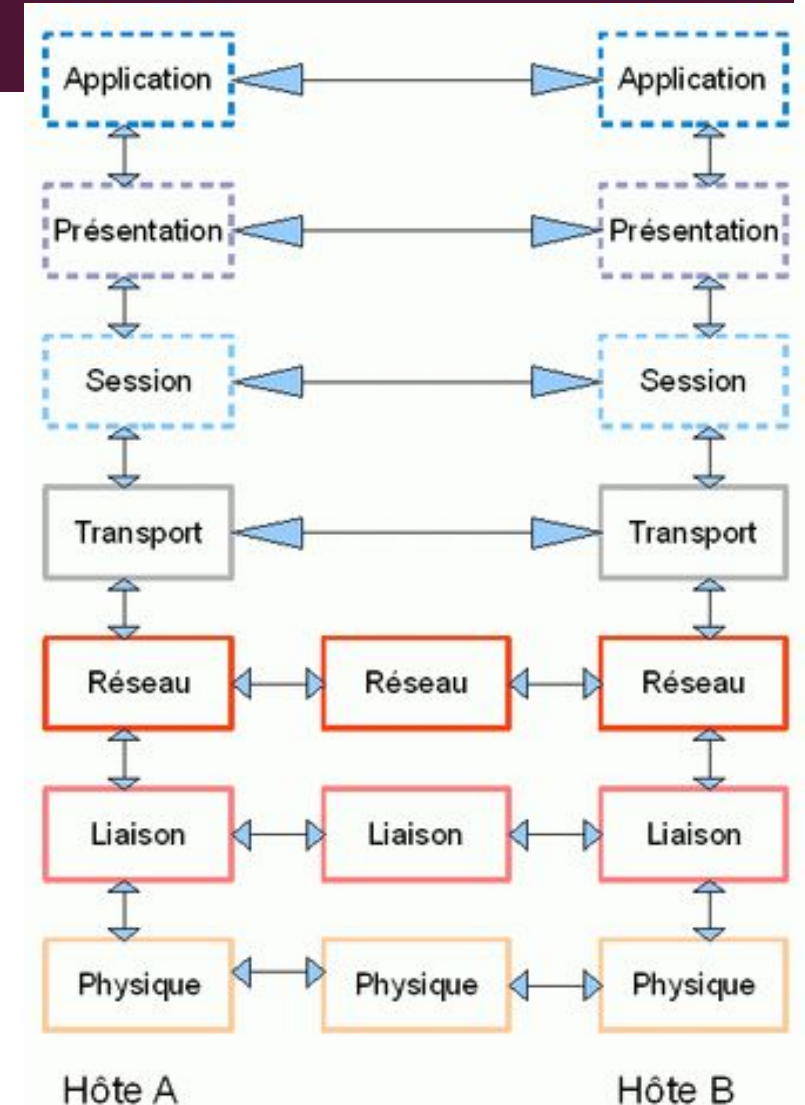
I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches:

- Le processus de communication:

1. Création de données au niveau de la couche application de l'hôte source;
2. Segmentation et encapsulation des données lorsqu'elles vont descendre la pile de protocoles dans l'hôte source;
3. Génération des données sur les différents supports qui se trouvent au niveau de la couche d'accès au réseau dans la pile;
4. Transport des données à travers l'inter-réseau, qui est composé de supports et de périphériques intermédiaires;
5. Réception des données par la couche d'accès au réseau de l'hôte de destination;
6. Décapsulation et assemblage des données lorsqu'elles remontent la pile au niveau de l'hôte de destination;
7. Transmission des données à l'application de destination.



I. RÉSEAUX INFORMATIQUES

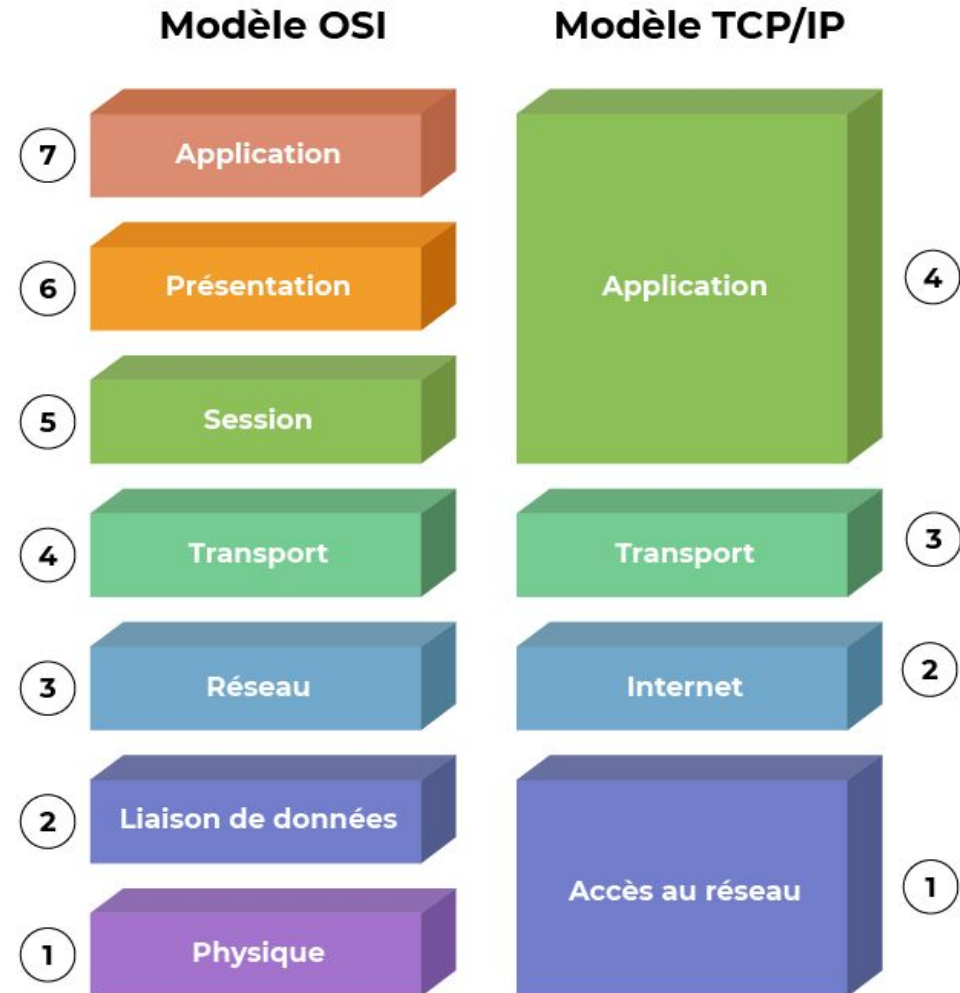
2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: OSI

- Objectif: assurer la transmission de bits entre les entités physiques: ETTD & ETCD
 - ❑ ETTD: équipement terminal de traitement de données
 - ❑ ETCD: équipement terminal de circuit de données
- Unité d'échanges: le bit
- Services:
 - ❑ La nature et caractéristiques du médium de communication;
 - ❑ Le mode de transmission sur le réseau;
 - ❑ Le choix du codage de bits;
 - ❑ Les tensions et fréquences utilisées.



1



I. RÉSEAUX INFORMATIQUES

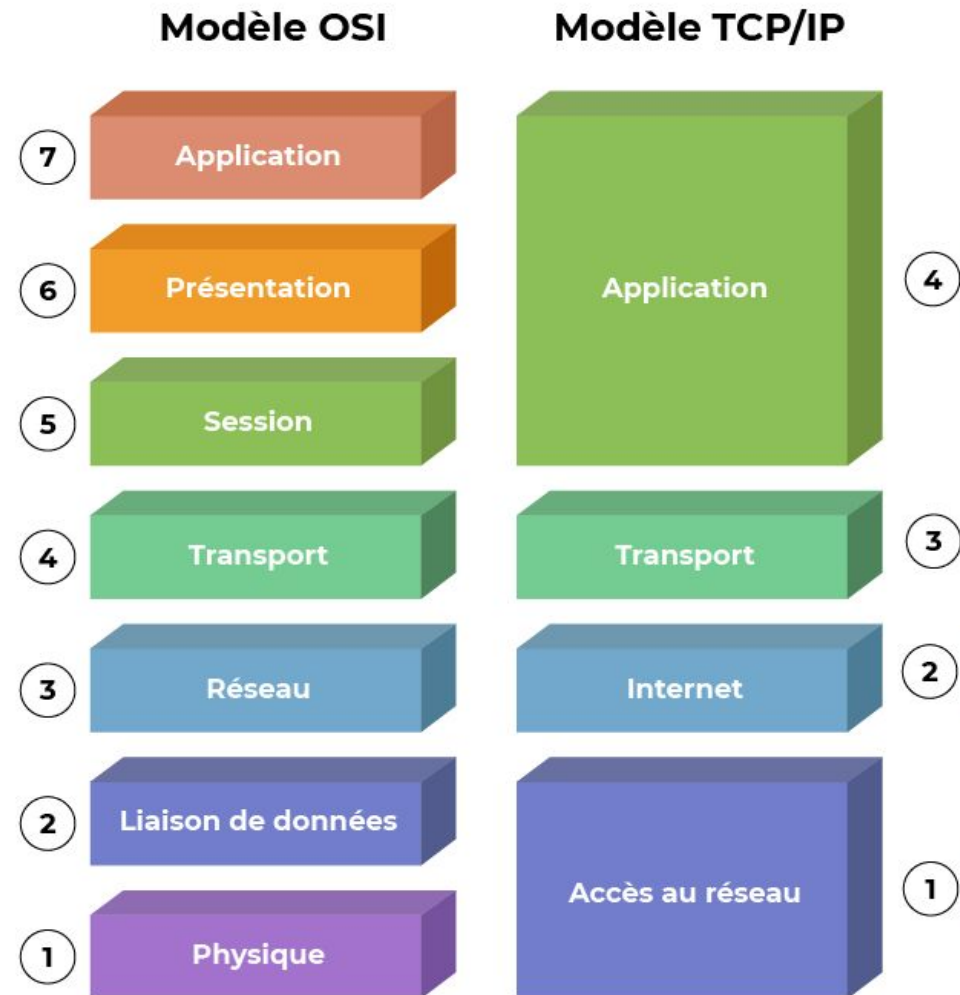
2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: OSI

- Objectif:
 - ☐ Assurer une transmission fiable entre deux systèmes;
 - ☐ Détecter et corriger les erreurs issues de la couche inférieure
- Unité d'échanges: la trame (frame)
- Services:
 - ☐ Structuration de données en trames;
 - ☐ Contrôle d'erreur en émission et réception;
 - ☐ Définition des règles de synchronisation.

Liaison de données

2



I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

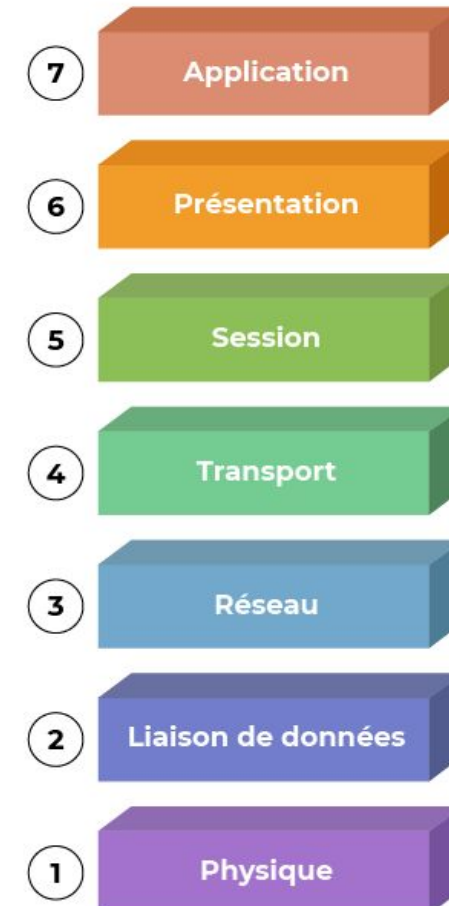
G. Modèles en couches: OSI

- Objectif:
 - ☐ Assurer l'acheminement des messages à travers le réseau;
 - ☐ Acheminement de bout-en-bout.
- Unité d'échanges: le paquet (packet)
- Services:
 - ☐ Le routage: déterminer un chemin permettant de relier deux périphériques finaux;
 - ☐ La commutation des paquets à travers le réseau.

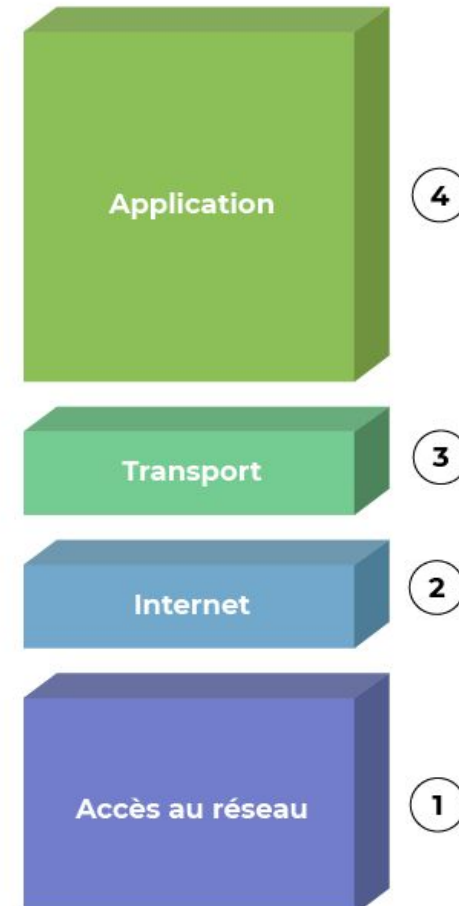


3

Modèle OSI



Modèle TCP/IP



I. RÉSEAUX INFORMATIQUES

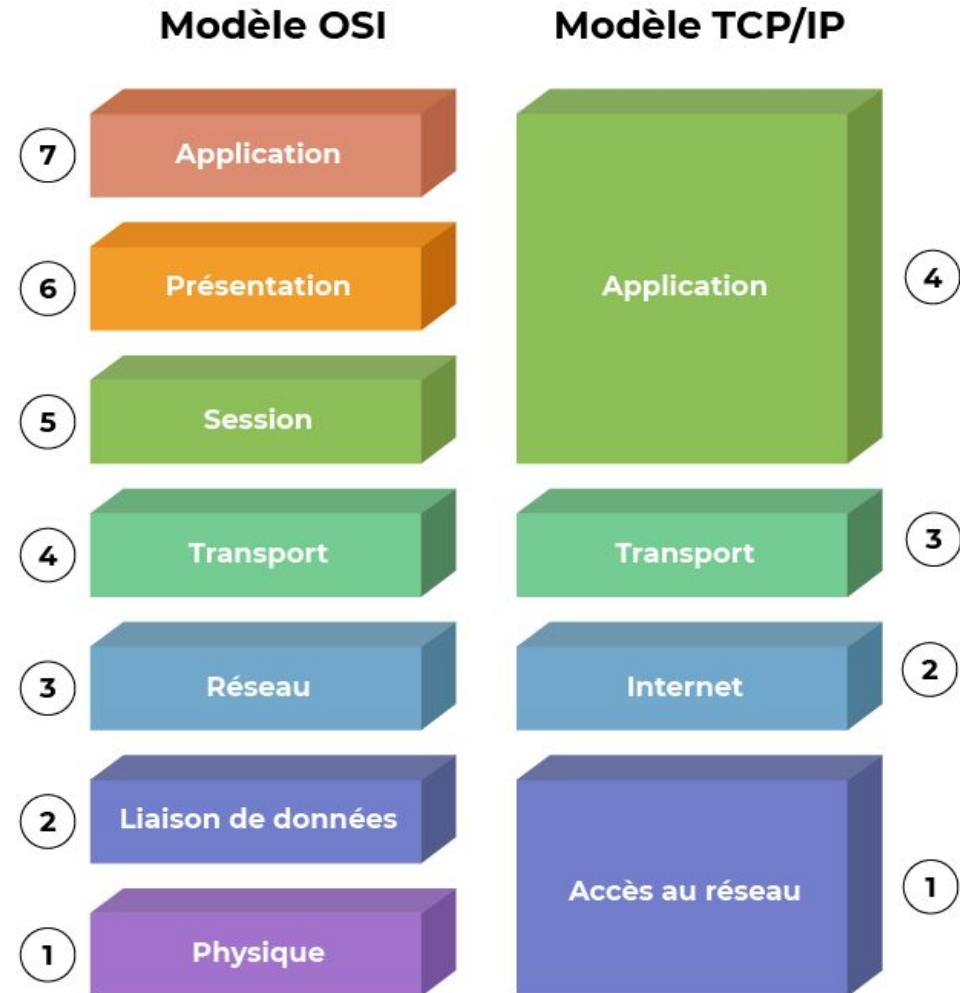
2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: OSI

- Objectif:
 - ☐ Acheminement de bout-en-bout.
- Unité d'échanges: le segment
- Services:
 - ☐ Prendre en charge la fragmentation en paquets et leur regroupement.



4



I. RÉSEAUX INFORMATIQUES

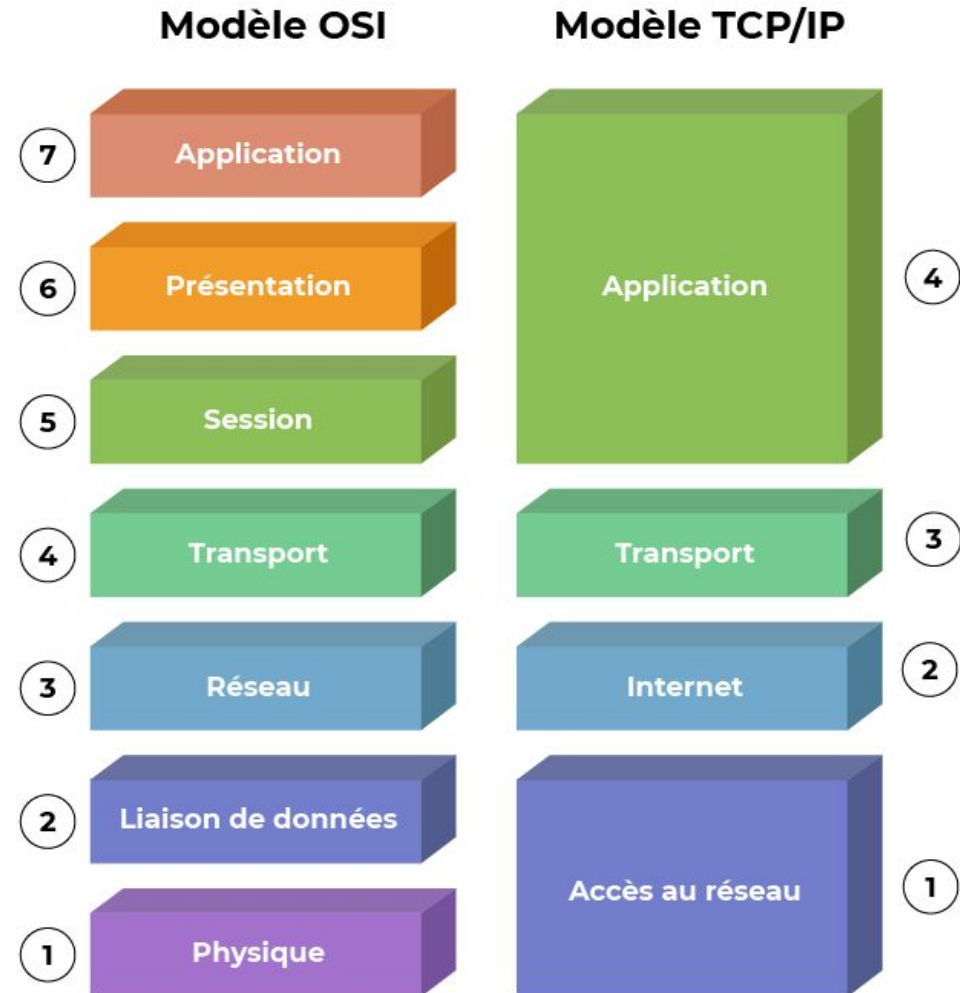
2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: OSI

- Objectif:
 - ❑ Fournir un ensemble de services pour la coordination des applications.
- Unité d'échanges: le datagramme
- Services:
 - ❑ Etablissement de connexion entre les applications;
 - ❑ Définition de points de synchronisation en cas d'erreur.



5

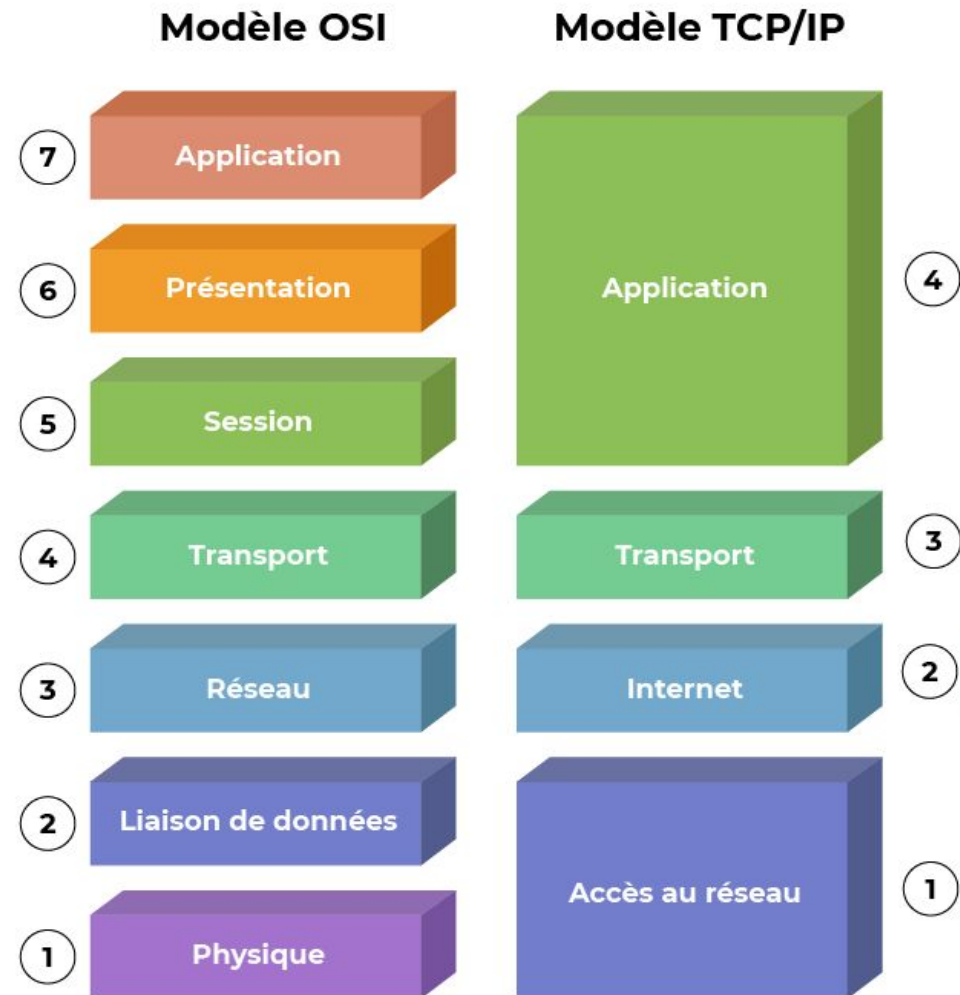


I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: OSI

- Objectif:
 - ❑ Permettre la manipulation des objets typés plutôt que des bits;
 - ❑ Fournir une représentation standard pour ces objets.
- Unité d'échanges: le datagramme
- Services:
 - ❑ Définition d'une notation abstraite pour les objets typés;
 - ❑ Compression / Cryptage.



I. RÉSEAUX INFORMATIQUES

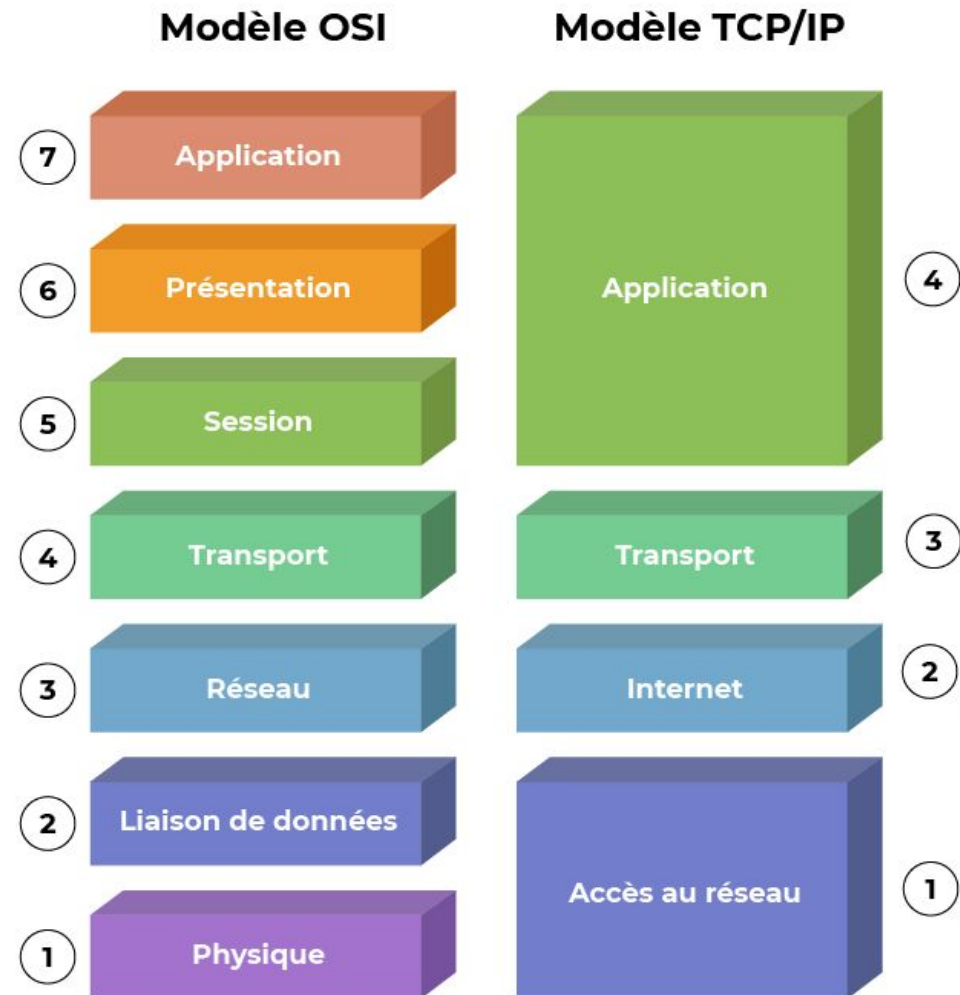
2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: OSI

- Objectif:
 - ❑ Apporter à l'utilisateur les services de base offerts par le réseau.
- Unité d'échanges: le datagramme
- Services:
 - ❑ Exemples d'applications standards: FTP, SMTP, WEB,
 - ❑ Terminaux virtuels: Telnet...



7



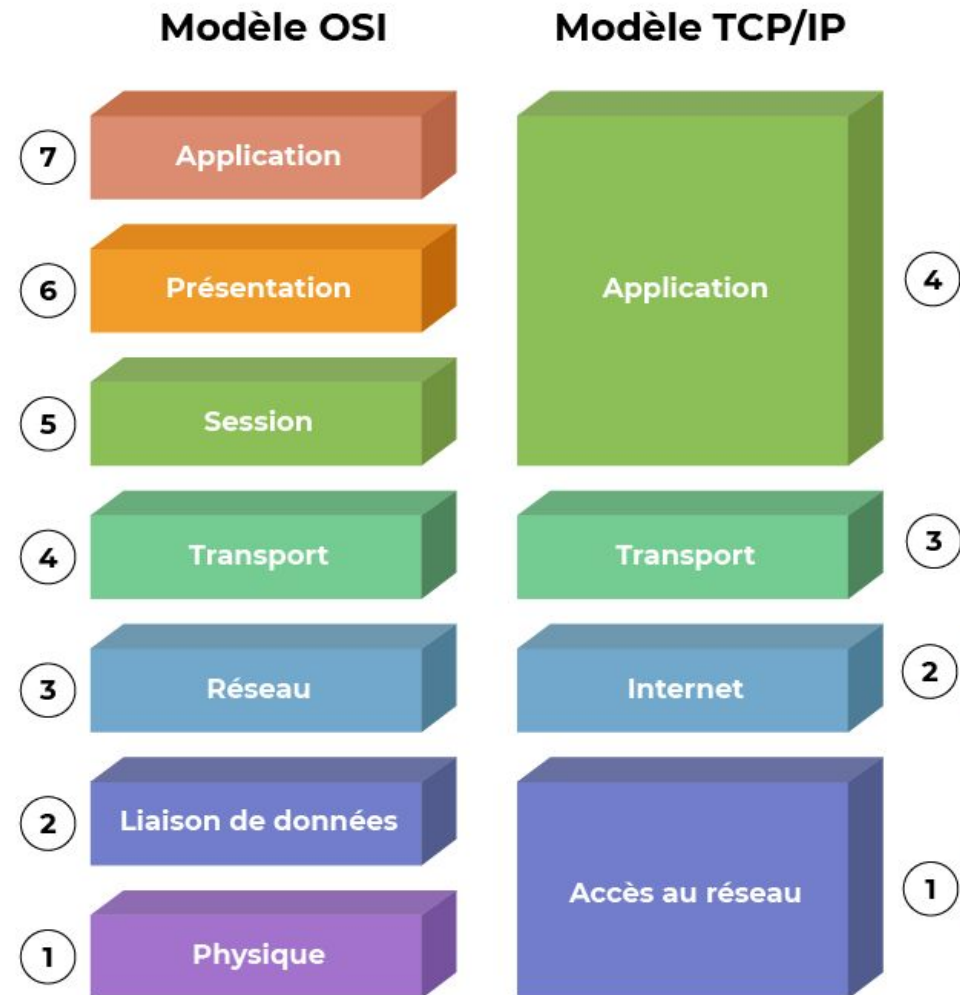
I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: TCP/IP



- Cette couche regroupe les couches physique et de liaison des données du modèle OSI.
- Objectif:
 - ❑ Fournir le moyen de délivrer des données aux systèmes rattachés au réseau.



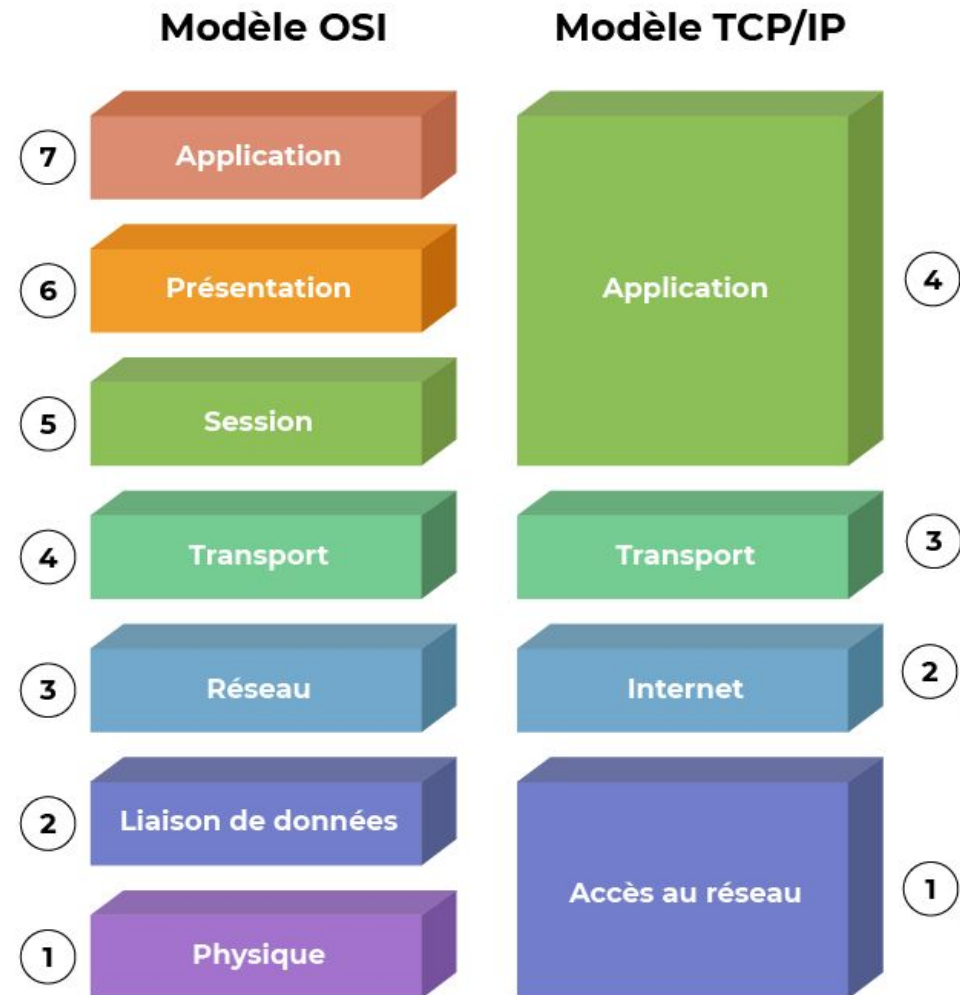
I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: TCP/IP



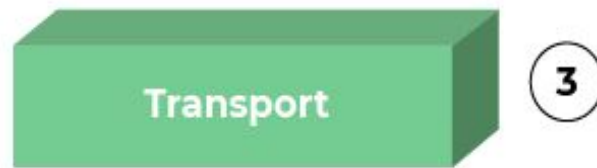
- L'équivalent de cette couche dans le modèle OSI est la couche Réseau.
- Objectif:
 - ❑ Assurer la transmission des paquets d'un hôte à un autre et de faire en sorte qu'ils arrivent à destination.



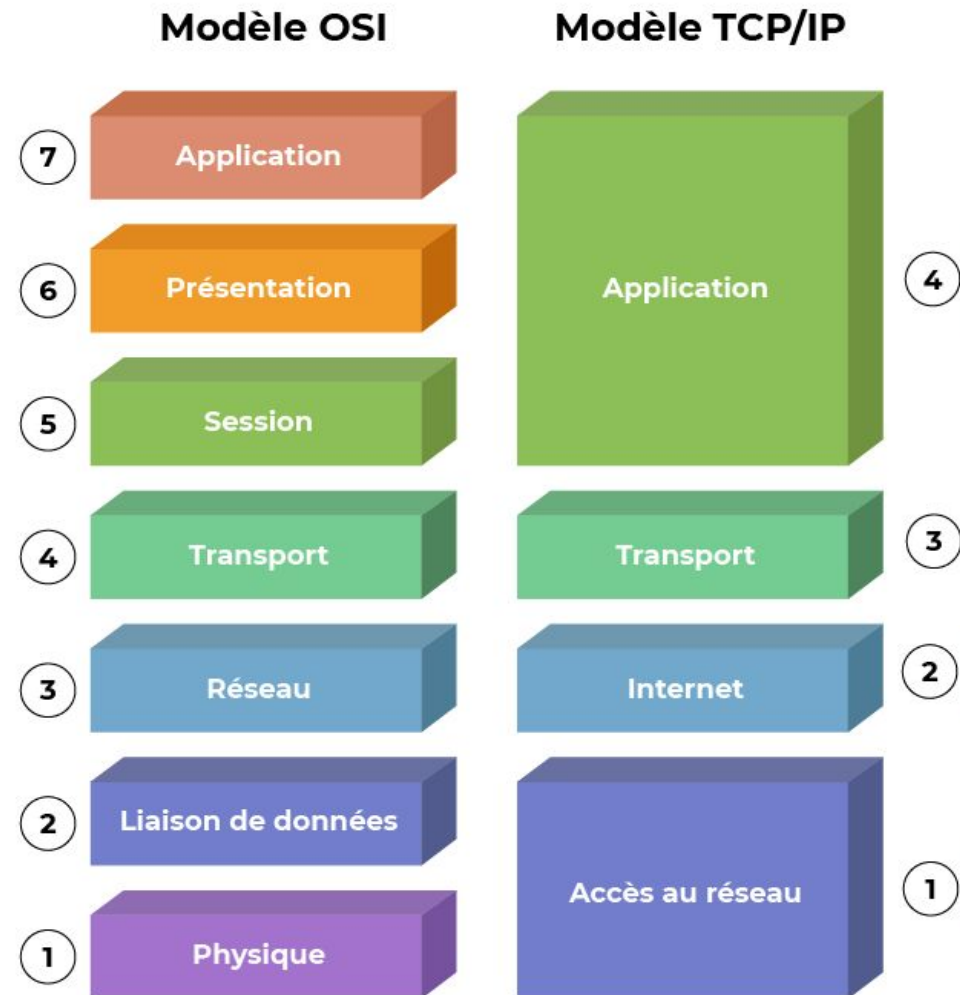
I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: TCP/IP



- Cette couche a le même rôle que la couche 4 du modèle OSI.
- Objectif:
 - ❑ Détermine comment les données doivent être envoyées : de manière fiable ou pas.



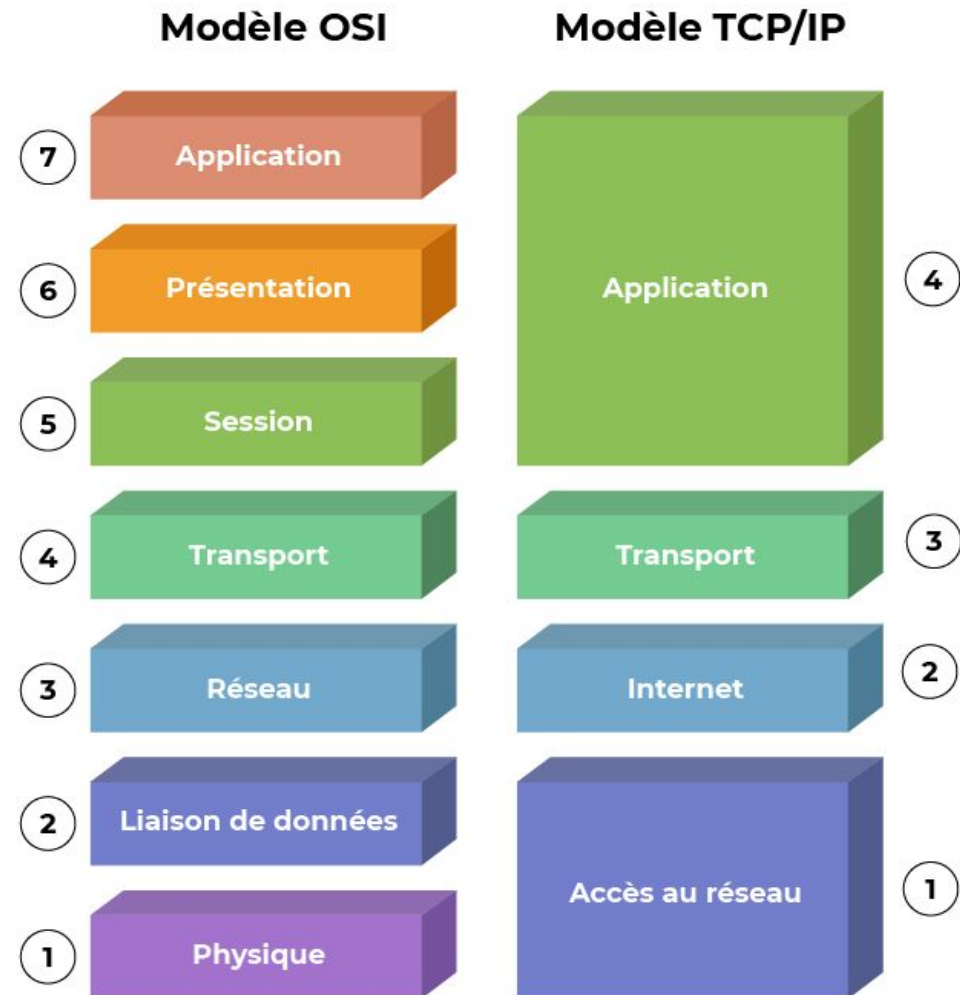
I. RÉSEAUX INFORMATIQUES

2) FONCTIONNEMENT DES RÉSEAUX

G. Modèles en couches: TCP/IP



- Cette couche regroupe les trois couches de session, présentation et application du modèle OSI.
- Objectif:
 - ❑ Fournir les protocoles de haut niveau.



PLAN DU COURS

I. Réseaux informatiques:

- 1) Introduction sur les réseaux et Internet
- 2) Fonctionnement des réseaux
- 3) Couche physique**
- 4) Couche Liaison de données
- 5) Couche réseau: Relayage et adressage IP
- 6) Routage dynamique
- 7) Couche transport: TCP & UDP
- 8) Applications: DNS & HTTP

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

A. Fonctions principales de la couche:

- Pour transmettre les bits entre un émetteur et un récepteur, il est nécessaire d'établir une liaison sur une voie de transmission munies d'équipement de transmission à ses extrémités.
- Types de support: câbles, fibre optique, ondes hertziennes,...
- Types de communication : Simplex, half-duplex, full-duplex



I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

A. Fonctions principales de la couche:



- Types de liaison:

ETTD – ETCD ou ETCD – ETCD

- ETTD: ordinateur, imprimante, scanner, téléphone mobile,...
- ETCD: modem, satellite, station de base,...

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

B. Les supports de transmission:

Deux catégories de supports:

- Les supports guidés (filaires) : Paires torsadées, câble coaxial, fibre optique.
- Les supports non guidés (non-filaires): Les ondes hertziennes, radio-électriques, lumineuses.

Performance d'un support: dépend de la bande passante, l'atténuation et le bruit.

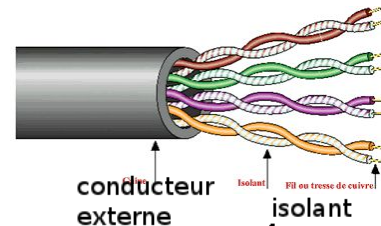
- Pour les supports non guidés, l'atténuation et le bruit sont très variables (conditions atmosphériques).

I. RÉSEAUX INFORMATIQUES

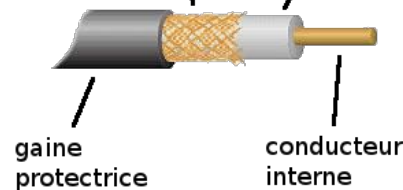
3) LA COUCHE PHYSIQUE

C. Les supports de transmission: supports guidés

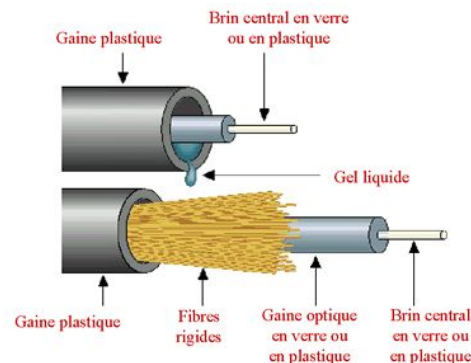
- La paire torsadée:



- Le câble coaxial:



- La fibre optique:



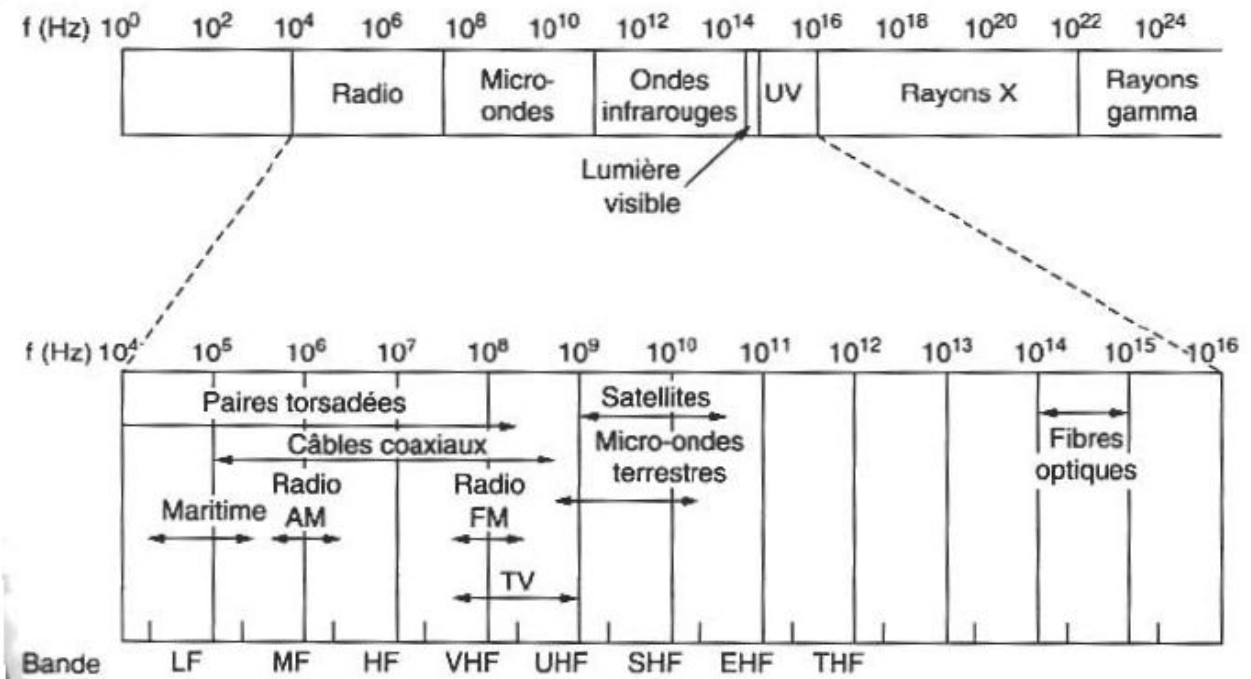
- Bande passante: 0.4 Mhz à 1000 MHz
- Débit: 10-100 Mbps
- Bande passante de quelques GHz
- Débit: jusqu'à 800 Mbps
- Très grande bande passante : plusieurs GHz
- Débit: 10-100 Gbps
- Distance entre les répéteurs: 10-100 km

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

C. Les supports de transmission: supports non-guidés

Support	Utilisations
Ondes radio	Radiodiffusion, militaire, wifi
Ondes micro-ondes	Téléphonie, télégraph, TV couleur, wifi, 4G, satellites
Ondes infrarouges	Communications point à point
Ondes millimétriques	Fixed-wireless access, 5G, réseau de backhaul
Ondes lumineuses	Li-fi, antennes, fibre



I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

C. Les équipements de la couche physique:

- Répéteur (Repeater): Un équipement qui permet de régénérer un signal entre deux nœuds du réseau afin d'étendre la distance du câblage d'un réseau.
- Concentrateur (Hub): Un équipement utilisé pour recevoir les câbles des divers ordinateurs connectés au réseau.
- Amplificateur (Amplifier): Un équipement augmentant la puissance d'un signal.
- Modem: Un équipement qui relie un ordinateur à un réseau analogique.



I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

D. La transmission de données:

- Types de transmission:
 - Numérique (ou en bande de base):
 - La transmission numérique consiste à faire transiter les informations sur le support physique de communication sous forme de signaux numériques.
 - Les signaux numérique sont facilement et fidèlement reproductibles
 - Analogique (ou par transposition en fréquence):
 - La transmission analogique de données consiste à faire circuler des informations sur un support physique de transmission sous la forme d'une onde.
 - Les signaux analogiques sont sensibles aux parasites et peuvent s'altérer dans le temps.

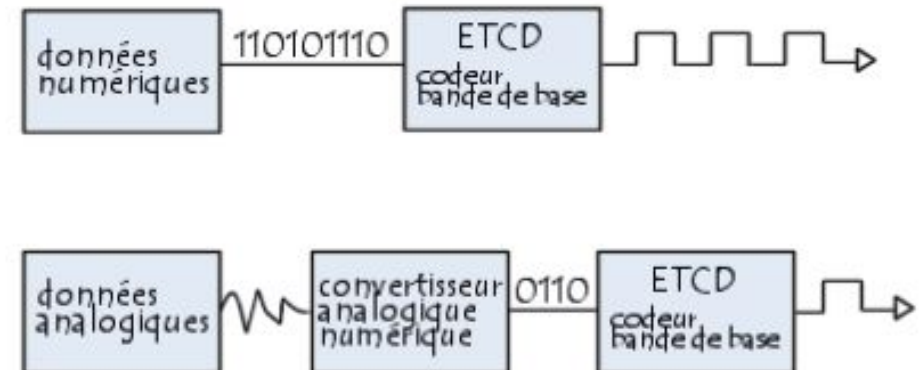
I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

D. La transmission de données:

□ Information:
suite de 0 et 1 représenté par un état logique.

- Transmission analogique d'informations analogiques
- Transmission analogique d'informations numériques
- Transmission numérique d'informations numériques
- Transmission numérique d'informations analogiques



I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

D. La transmission de données:

□ Signal:

Amplitude: Pour un signal électrique, l'amplitude correspond à la valeur maximale de la tension.

Fréquence: le nombre de périodes par unité de temps ce qui correspond à l'inverse de la période : $f = 1/T$

Phase: Décalage entre le départ de la sinusoïde et l'origine des temps.

Valence: nombre d'états physiques utilisés pour coder des bits.

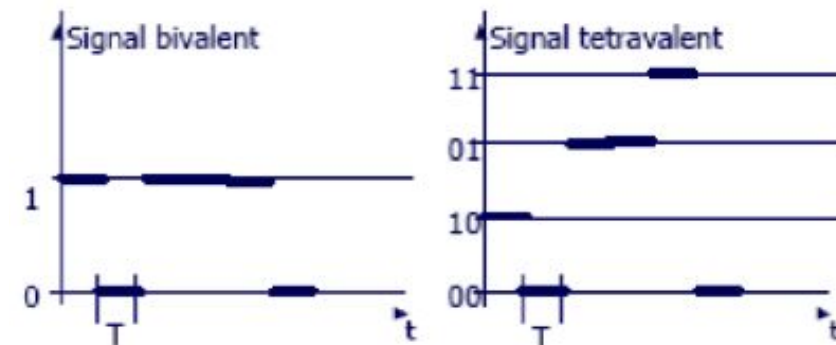
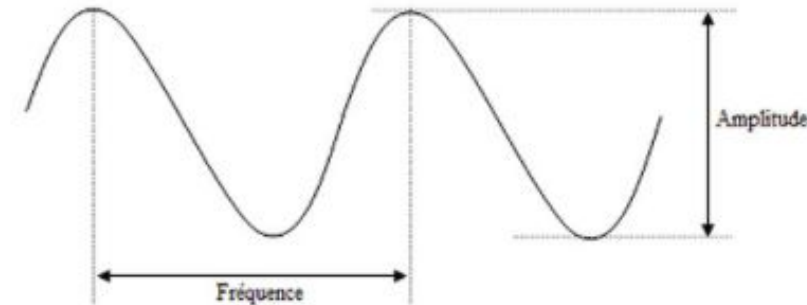
Moment élémentaire: durée T pour transmettre un état logique.

□ Problèmes de transmission:

Affaiblissement: (atténuation): diminution de la puissance d'une onde lorsqu'elle parcourt une certaine distance.

Déphasage: décalage entre le signal et l'origine des temps.

Bruit: signal extérieur qui s'intègre au signal transmis.



I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

D. La transmission:

- Débit binaire (Formule de Nyquist):

Formule de Nyquist : Permet de calculer le débit maximal d'une ligne de transmission idéale sans bruit.

$$D = 2B \times \log_2 V$$

D: Le débit binaire (bits/s)

B: La bande passante (Hz)

V: La valence du signal

$R = 2B$: La rapidité de modulation (Bauds)

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

D. La transmission:

- Débit binaire (Formule de Nyquist):

$$D = 2B \times \log_2 V$$

➤ Exemple:

Calculer le débit binaire maximal D_{\max} pour un signal binaire (valence 2) que l'on peut faire passer par une ligne de transmission idéalement non bruitée dont on sait que la bande passante est 2 MHz.

$$B = 2.10^6 \text{ Hz}$$

$$\log_2 V = \log_2 2 = 1$$

$$D_{\max} = 2.2.10^6.1 = 4 \text{ Mbps}$$

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

D. La transmission:

- Débit binaire (Formule de Shannon):

Formule de Shannon : Permet de calculer le débit maximum d'une ligne subissant un bruit.

$$D = B \times \log_2 (1 + S/N)$$

D: Le débit binaire (bits/s)

B: La bande passante (Hz)

S: La puissance du signal

N: La puissance du bruit

S/N: Le rapport signal sur bruit (Signal-to-noise ratio) (décibels dB)

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

D. La transmission:

- Débit théorique (Formule de Shannon):

$$D = B \times \log_2 (1 + S/N)$$

➤ Exemple:

Calculer le débit binaire maximal que l'on peut faire passer sur une ligne de transmission dont on sait que la bande passante est 2 MHz si le rapport S/N est de 12 dB ?

$$10 \log_{10} (S/N) = 12 \text{ dB}$$

$$S/N = 10^{1,2} = 16$$

$$D_{\max} = 2 \cdot 10^6 \cdot \log_2(17) = 2,41 \cdot 10^6 = 8,2 \text{ Mbps}$$

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

E. Codage des signaux :

□ Définition: Le codage du signal est utilisé pour représenter les 1 et les 0 d'un signal numérique sur le lien, ce processus est appelé codage en ligne. Après le codage en ligne, le signal peut être directement émis sur le canal de transmission, sous la forme de variations de la tension ou du courant.

□ Les codes usuels utilisés:

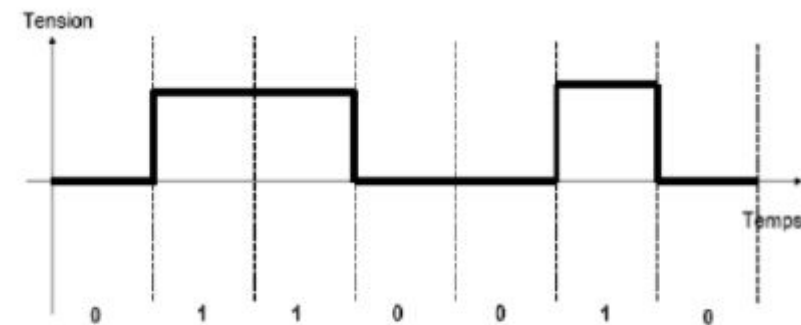
- Code en bande de base
- NRZ
- RZ
- Bipolaire
- Biphase (Manchester)
- Miller

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

E. Codage des signaux :

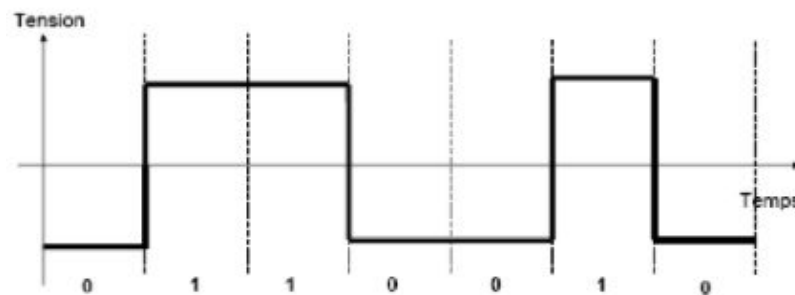
- Code en bande de base:
Le signal est transmis tel quel.



Code tout ou rien.

Circuit ouvert : 1
Circuit fermé : 0

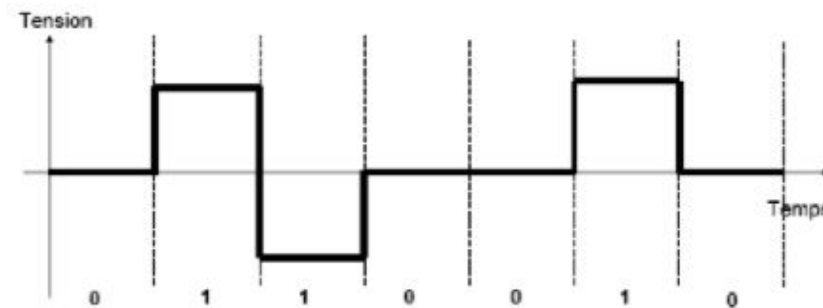
- Code NRZ:
Le signal est transmis avec une simple transposition en tension.



Code NRZ (Non Return to Zero).

Circuit ouvert : pas de transmission
Circuit fermé : tension positive 1
tension négative 0

- Code bipolaire:
Si 0 alors $V = 0$ sur la période.
Si 1 alors $V =$ alternativement $-A$ et $+A$ sur la période.



Code bipolaire

Circuit ouvert : 0
Circuit fermé alternativement positive et négative : 1
(Eviter de maintenir un signal continu)

I. RÉSEAUX INFORMATIQUES

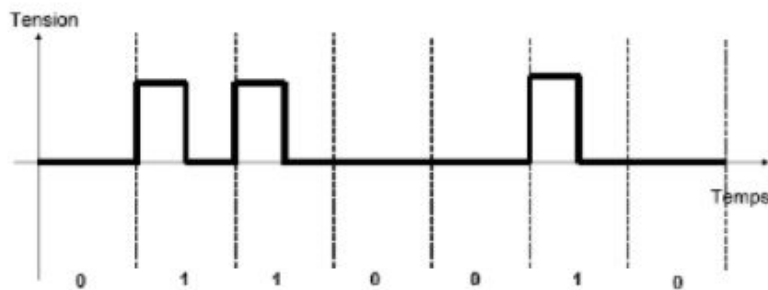
3) LA COUCHE PHYSIQUE

E. Codage des signaux :

□ Code RZ:

Si 0 alors $V = 0$

Si 1 alors $V = [+A, 0]$

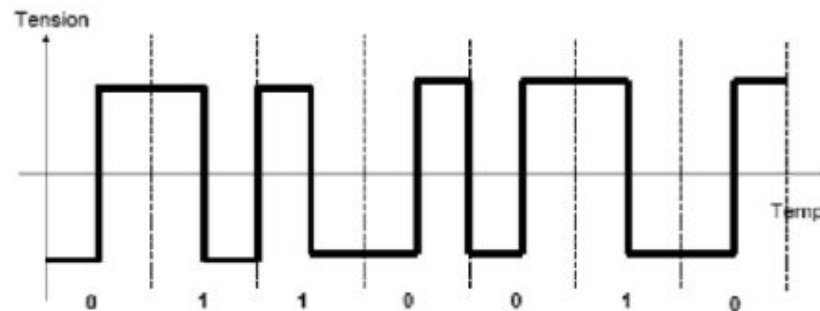


Code RZ (Return to Zero)

Front descendant : 1
Sinon : 0
(détecter un front descendant)

□ Code biphase (Manchester):

XOR entre les données et l'horloge.

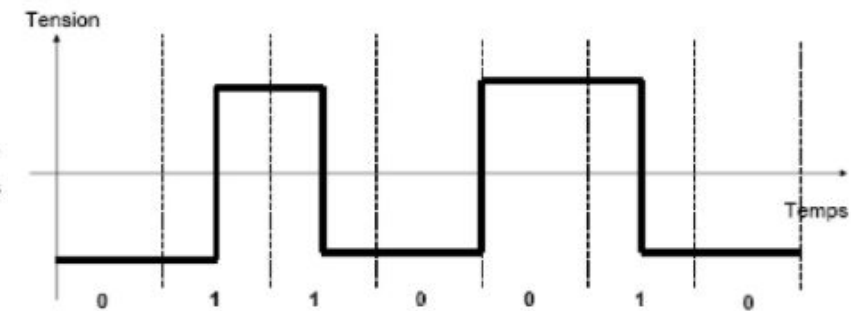


Code biphase (ou Manchester)

Front montant sur l'intervalle : 0
Front descendant sur l'intervalle : 1
(détecter les fronts)

□ Code Miller:

Une transition à la demi-période si 1,
Une transition à la fin de la période si 0
et si le bit suivant est également 0.



Code Miller

Front sur l'intervalle : 1
Pas de front sur l'intervalle : 0
(détecter des fronts et minimiser le nombre d'oscillations)

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

E. Codage des signaux :

□ Exercice:

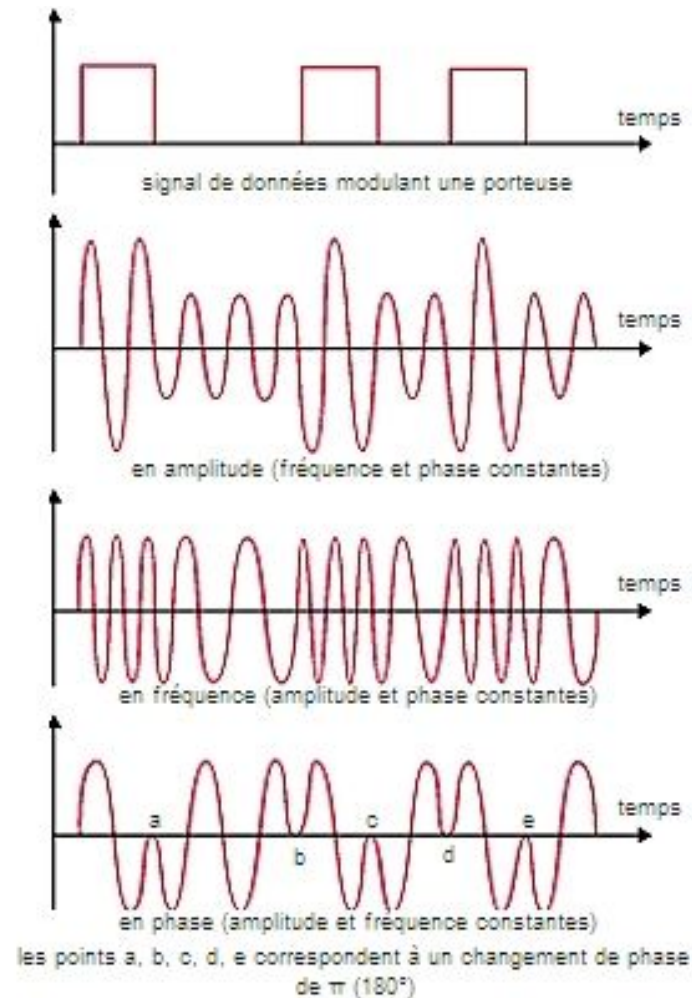
Représenter le signal 11000101 en utilisant les différents codages

I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

E. Codage des signaux :

□ Modulation:

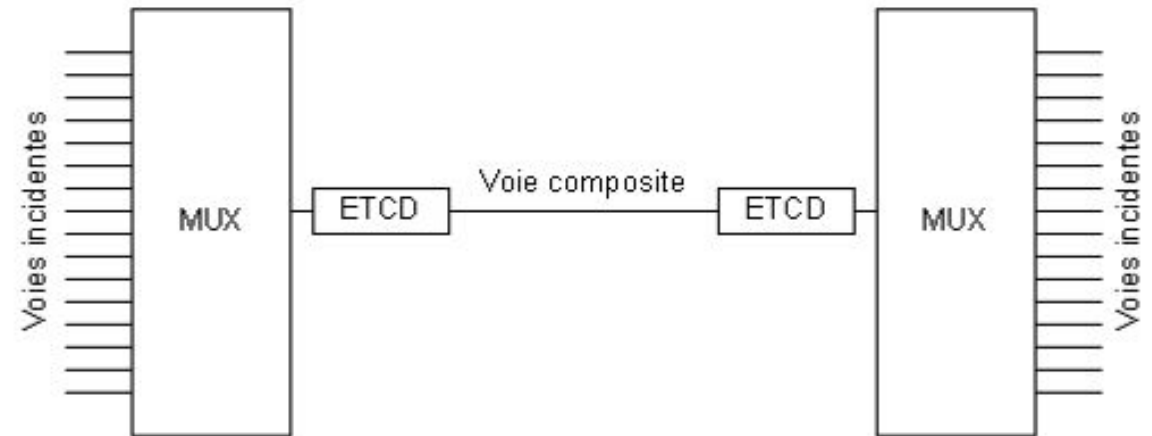


I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

H. Multiplexage:

- Définition: Un mécanisme qui permet le partage d'une même ligne de transmission entre plusieurs communications simultanées.
- Types de multiplexage:
 - FDMA (Frequency Division Multiple Access)
 - TDMA (Time Division Multiple Access)
 - CDMA (Code Division Multiple Access)



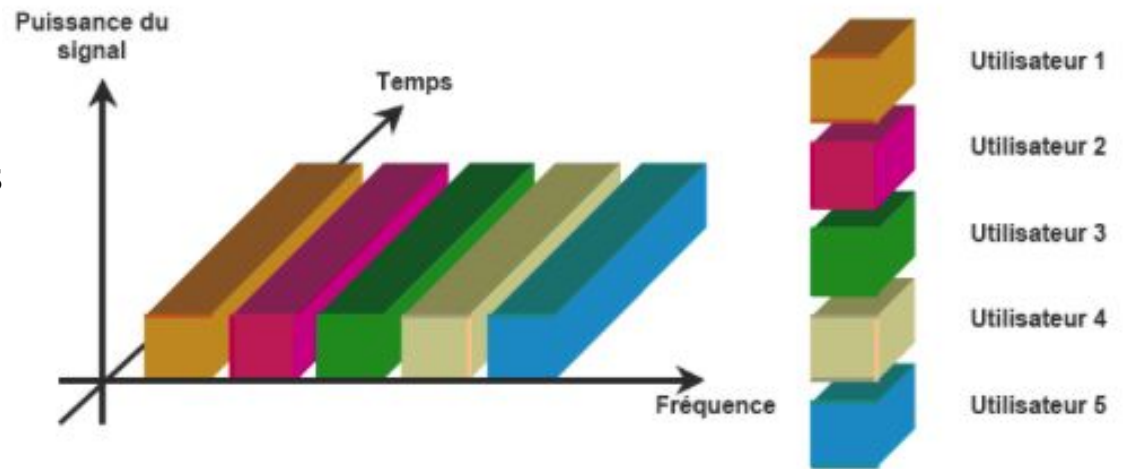
I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

H. Multiplexage:

→ FDMA: Répartition en fréquence

- ◆ Principe: répartir la bande passante en N sous-bandes, chacune est attribuée à une communication (utilisateur).
- ◆ Avantage: plus adapté aux transmissions analogiques simple à réaliser.
- ◆ Inconvénient: gaspillage de la bande passante (des utilisateurs peuvent ne pas utiliser leurs sous-bandes).



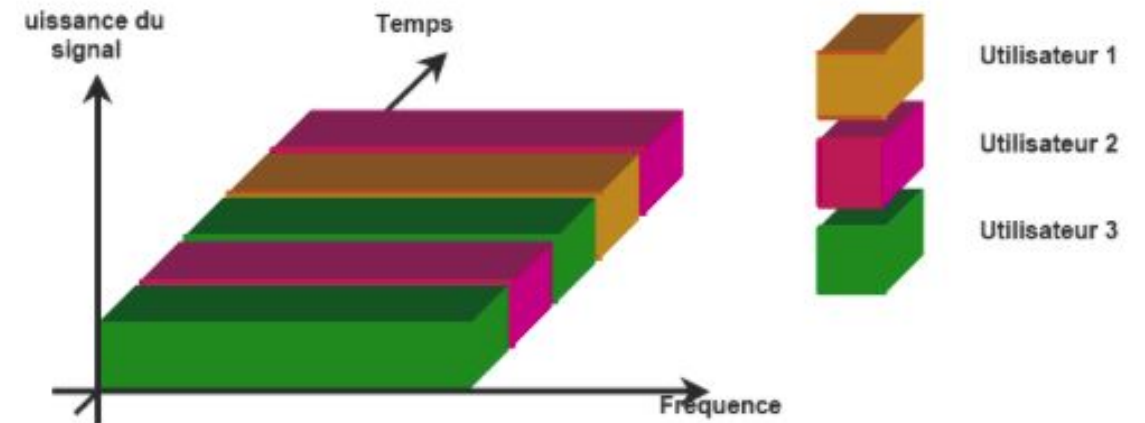
I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

H. Multiplexage:

→ TDMA: Répartition en temps

- ◆ Principe: Le temps de transmission est découpé en un ensemble d'intervalles.
- ◆ Avantage: Moins d'interférences.
- ◆ Inconvénient: gaspillage de la bande passante (une seule station utilise toute la bande passante pendant un intervalle).



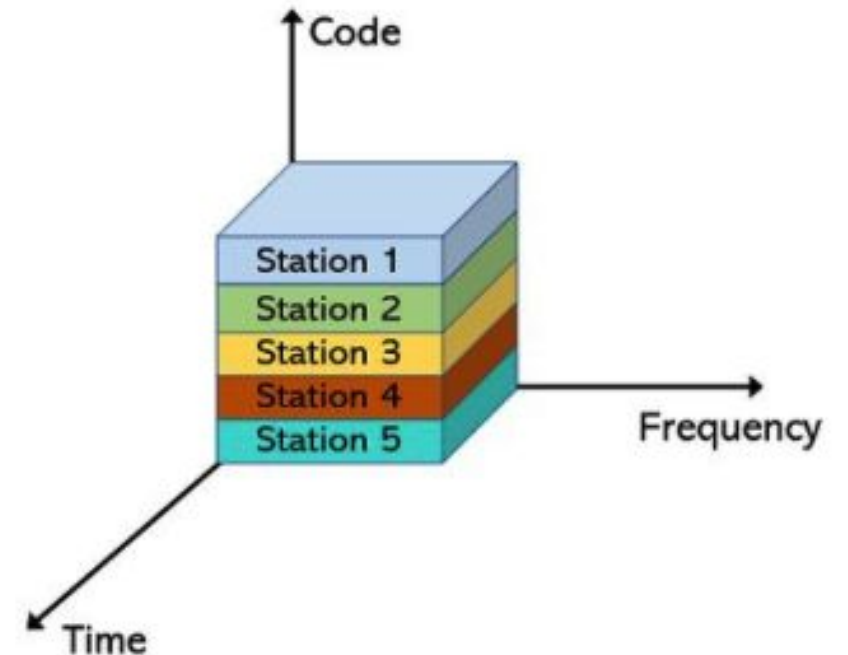
I. RÉSEAUX INFORMATIQUES

3) LA COUCHE PHYSIQUE

H. Multiplexage:

→ CDMA: Répartition en code

- ◆ Principe: utilisation simultanée de plusieurs codes.
- ◆ Avantage: Résistance aux interférences, plusieurs stations peuvent utiliser une même bande de fréquence et peuvent transmettre en même temps.
- ◆ Inconvénient: L'exécution du CDMA se dégrade avec une augmentation du nombre d'utilisateurs.



PLAN DU COURS

I. Réseaux informatiques:

- 1) Introduction sur les réseaux et Internet
- 2) Fonctionnement des réseaux
- 3) Couche physique
- 4) Couche Liaison de données**
- 5) Couche réseau: Relayage et adressage IP
- 6) Routage dynamique
- 7) Couche transport: TCP & UDP
- 8) Applications: DNS & HTTP

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

A. Les fonctions de la couche LD:

- Sous couches:
 - MAC (Multiple Access Control) : Contrôle d'accès au support
 - Filtrage des trames reçues;
 - Encapsulation en trames: adressage, délimitation de trames, contrôle d'erreurs;
 - Gérer l'accès au support partagé.
 - LLC (Logical Link control) : Contrôle de la liaison logique
 - Multiplexage/démultiplexage de protocoles;
 - Retransmission des trames erronées.

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

B. Adressage physique (@ MAC):

- Adresse physique (adresse MAC): est un identifiant unique physique stocké dans une carte réseau.

B4-6D-83-DD-CE-49

Identification du
constructeur

Identification de
la carte réseau

- Adresse logique (adresse IP): un numéro d'identification attribué à une machine connectée au réseau Internet.

17.172.224.47

8 bits
(1 byte)

8 bits
(1 byte)

8 bits
(1 byte)

8 bits
(1 byte)

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

- Le contrôle d'erreur consiste à vérifier la validité des données transmises :
 - On ajoute à l'information transmise des informations de contrôle calculées par un algorithme spécifié;
 - À la réception : on exécute le même algorithme pour vérifier la validité de l'information reçue;
 - Si c'est le cas : pas d'erreurs de transmission et l'information reçue est traitée;
 - Sinon : l'information est invalide et elle est ignorée.
- Les codes de protection d'erreur:
 - Contrôle de parité (Parity check)
 - Contrôle polynomial (CRC)
 - Somme de contrôle (Checksum)

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

- Contrôle de parité:

Principe du codage: On compte le nombre de bits à 1 et on colle le résultat directement à la suite des bits envoyés.

- Deux types: Parité paire et parité impaire.

Exemple: (parité paire)

1000001 -> 10000010

0110100 -> 01101001

- A la réception, si le nombre de bits 1 est pair, on suppose qu'il n'y a pas eu d'erreur. Sinon, on sait qu'il y a eu une erreur mais on ne peut pas la localiser.

- Inconvénient: ne détecte que les erreurs en nombre impair.

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

- Contrôle polynomial:

A toute séquence de bits on associe un polynôme: $P(x) = p_0 + p_1.X + p_2.X^2 \dots + p_n.X^n$

Les séquences envoyées doivent être un multiple d'un polynôme $G(X)$ dit polynôme générateur.

Le polynôme générateur est connu à l'avance par l'émetteur et le récepteur.

Exemple: |1000| correspond au polynôme $X^5 + X^4 + X^0$

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

▪ Contrôle polynomial:

Principe du codage:

A l'émission:

- On multiplie $P(x)$ par X^r (r : le degré de $G(X)$)
- On divise le polynôme obtenu par $G(X)$
- On obtient un reste noté $R(X)$
- On transmet le polynôme $T(X)$ constitué à partir de $P(X)$ et $R(X)$:

$$(\text{mot de code}) \quad T(X) = X^r \cdot P(X) + R(X)$$

A la réception:

- On reçoit l'information $S(X)$
- On divise $S(X)$ par $G(X)$
- Si le reste $R'(X)$ obtenu est nul, l'information reçue correspond à celle émise
- Si $R'(X)$ est non nul, le circuit de données a introduit des erreurs et l'information n'est pas prise en compte.

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

c. Contrôle d'erreur:

- Contrôle polynomial:

Exercice:

- Trame: 1101011011
- Générateur: 10011
- La trame émise: ?

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

- Contrôle polynomial:

Solution:

➤ 1^{ère} étape: représenter la trame 1101011011 par un polynôme $P(X)$

$$P(X) = X^9 + X^8 + X^6 + X^4 + X^3 + X + 1$$

➤ 2^{ème} étape: représenter le générateur 10011 par un polynôme $G(X)$

$$G(X) = X^4 + X + 1$$

➤ 3^{ème} étape: multiplier $P(X)$ par X^4

$$P'(X) = P(X) \cdot X^4 = X^{13} + X^{12} + X^{10} + X^8 + X^7 + X^5 + X^4 = (\text{en binaire}) 11010110110000$$

I. RÉSEAUX INFORMATIQUES

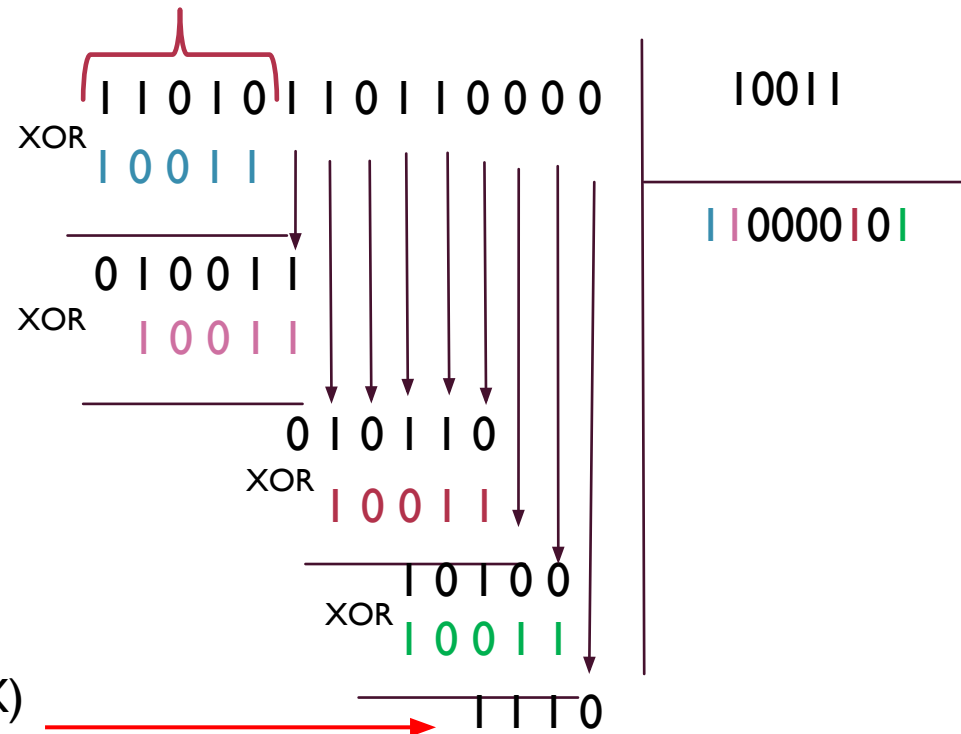
4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

▪ Contrôle polynomial:

Solution:

- 4^{ème} étape: Diviser $P'(X)$ par $G(X)$



Le reste $R(X)$

Rappel

$$\text{XOR}(1,0) = 1$$

$$\text{XOR}(1,1) = 0$$

$$\text{XOR}(0,0) = 0$$

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

- Contrôle polynomial:

Solution:

□ 5^{ème} étape: Calculer la trame émise

$$T(X) = P'(X) + R(X)$$

$$\begin{array}{r} 11010110110000 \\ + 1110 \\ \hline 11010110111110 \end{array}$$

↖ Résultat

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

- Checksum:

Principe du codage: Regrouper des données en blocs de n bits et puis ajouter le checksum à la fin de la trame.

Le checksum est le complément de la somme des blocs.

Exemple 1: On suppose que la trame **1010 1001 0011 1001** sera envoyée

□ Emission:

10101001

00111001

Somme 11100010

Checksum 00011101

Trame envoyée: **10101001 00111001 00011101**

□ Réception:

10101001

00111001

00011101

Somme 11111111

Checksum **00000000** -> **Pas d'erreur**

≠ Erreur

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

- Checksum:

Principe du codage: Regrouper des données en blocs de n bits et puis ajouter le checksum à la fin de la trame.

Le checksum est le complément de la somme des blocs.

Exemple 2: On suppose que la trame **1110 0010 1110 1100** sera envoyée. En utilisant des blocs de 4 bits, quel sera le checksum ajouté à la trame?

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

C. Contrôle d'erreur:

- Checksum:

Principe du codage: Regrouper des données en blocs de n bits et puis ajouter le checksum à la fin de la trame.

Le checksum est le complément de la somme des blocs.

Exemple 2: On suppose que la trame **1110 0010 1110 1100** sera envoyée. En utilisant des blocs de 4 bits, quel sera le checksum ajouté à la trame?

1110

0010

1110

1100

1010

10

Somme

1100 checksum: 0011

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

D. Synchronisation des trames:

- Les flux de bits transmis doivent être encadrés en blocs d'informations discernables.
- Ceci peut être réalisé en utilisant le cadrage, c'est-à-dire attacher des types spéciaux de modèles de bits au début et à la fin de la trame.
- Méthodes de cadrage de trames:
 - Character count (Nombre de caractères);
 - Flag byte with character stuffing (Octets de drapeau avec bourrage d'octets);
 - Starting and ending flags, with byte stuffing (Octets de drapeau avec bourrage de bits).

I. RÉSEAUX INFORMATIQUES

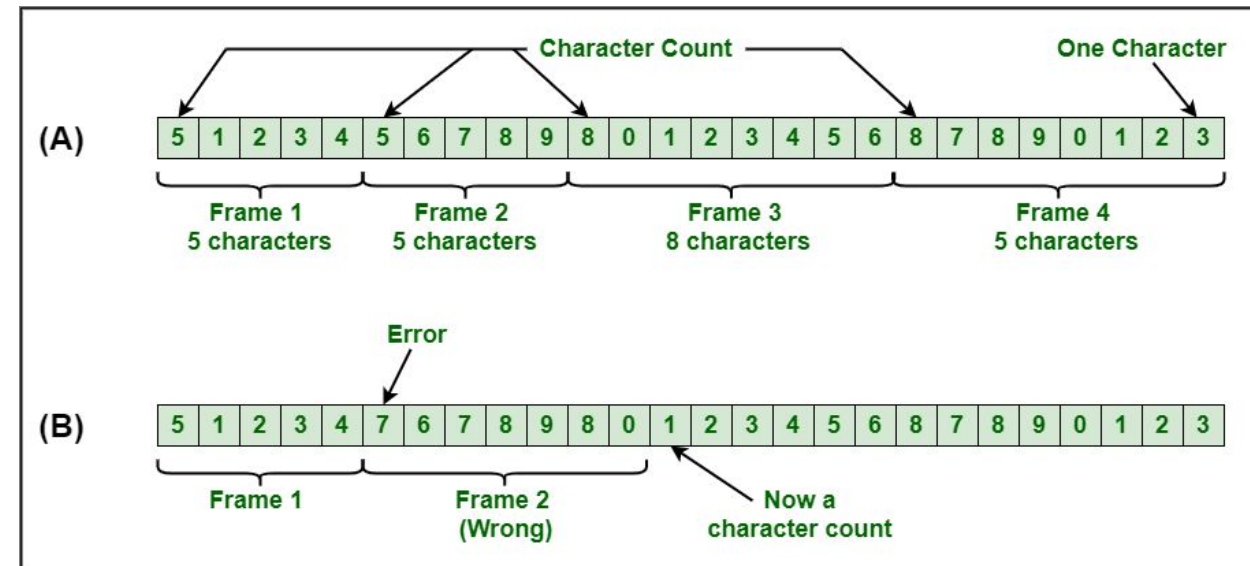
4) LA COUCHE LIAISON DE DONNÉES

D. Synchronisation des trames:

- **Nombre de caractères:**

Principe: Ajouter un champ dans l'en-tête qui contient le nombre total de caractères présents dans la trame.

- Si le nombre de caractères est perturbé par une erreur alors la destination peut perdre la synchronisation.



A Character Stream

(A) Without Errors

(B) With one Error

I. RÉSEAUX INFORMATIQUES

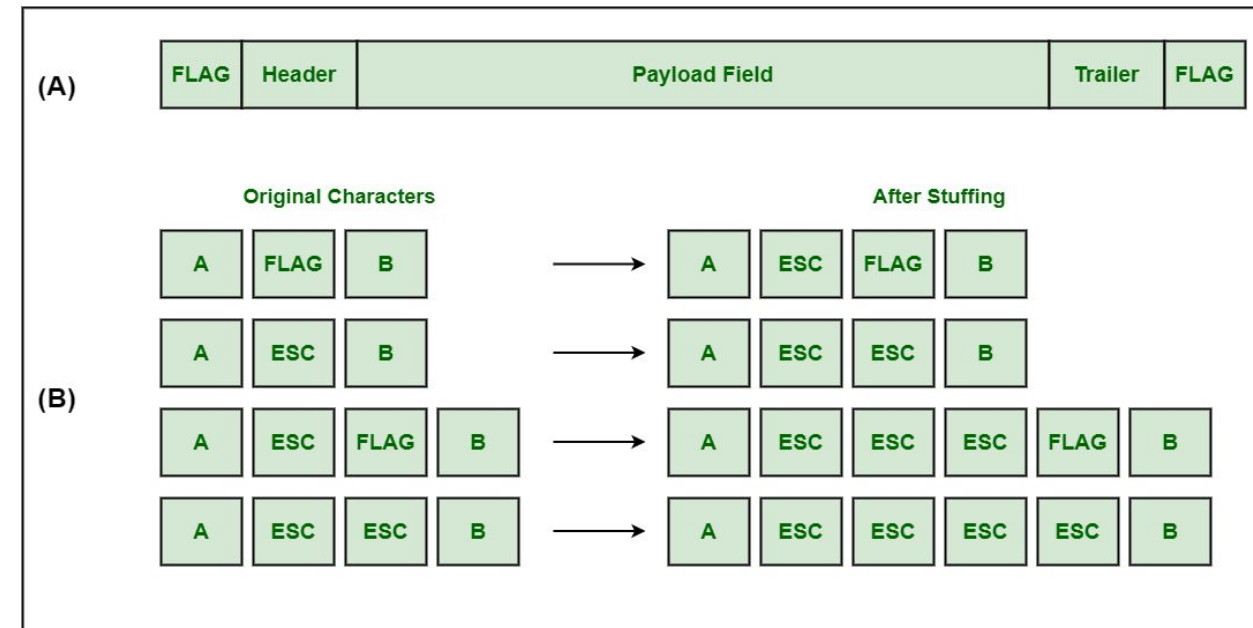
4) LA COUCHE LIAISON DE DONNÉES

D. Synchronisation des trames:

- Octets de drapeau avec bourrage d'octets:

Principe:

- Ajouter un drapeau au début et fin de la trame.
- Ajouter un caractère d'échappement ESC à la section de données de la trame lorsqu'il y a un caractère qui a le même motif que celui de l'octet de drapeau ou l'ESC.



A Character Stuffing

(A) A frame delimited by flag bytes

(B) Four examples of byte sequences before and after byte stuffing

I. RÉSEAUX INFORMATIQUES

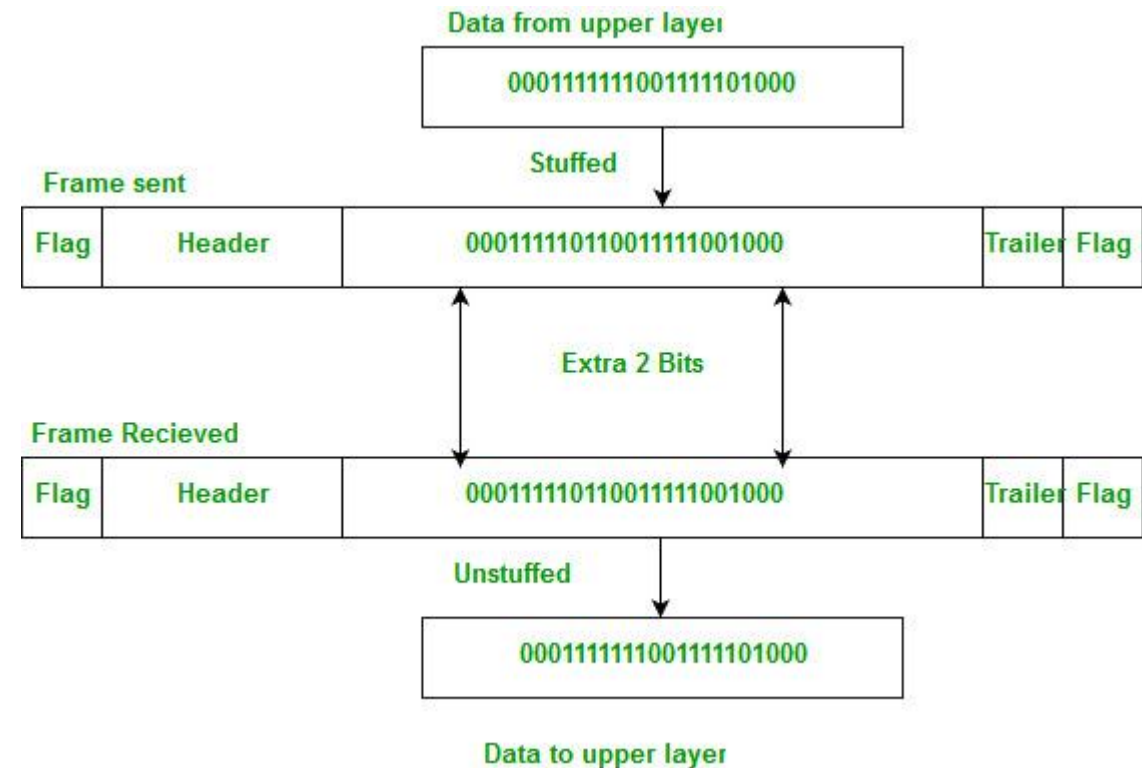
4) LA COUCHE LIAISON DE DONNÉES

D. Synchronisation des trames:

- Octets de drapeau avec bourrage de bits:

Principe:

- Généralement le bloc « 01111110 » est utilisé comme drapeau de début et de fin.
- Pour assurer qu'à l'intérieur de la trame la configuration 01111110 ne soit pas interprétée comme un drapeau de fin, on ajoute un bit 0 après 5 bits « 1 » consécutifs.



I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

D. Synchronisation des trames:

- Exercice:

Synchronisez la trame suivante en utilisant les 3 méthodes:

10010101 01011111 11010111 11011001

Drapeau = 01111110

ESC = 11100000

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

D. Synchronisation des trames:

Gestion d'acquittement:

- Erreurs de transmission: réception d'une trame erronée ou perte d'une trame.
- Comment assurer la bonne transmission des trames? Envoi d'un message d'acquittement à l'émetteur.
- Protocoles de gestion d'acquittement:
 - Protocole d'attente/réponse (Send and Wait protocol-ARQ);
 - Rejet simple (Sliding window protocol);
 - Rejet sélectif (Sliding window protocol with selective repeat).

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

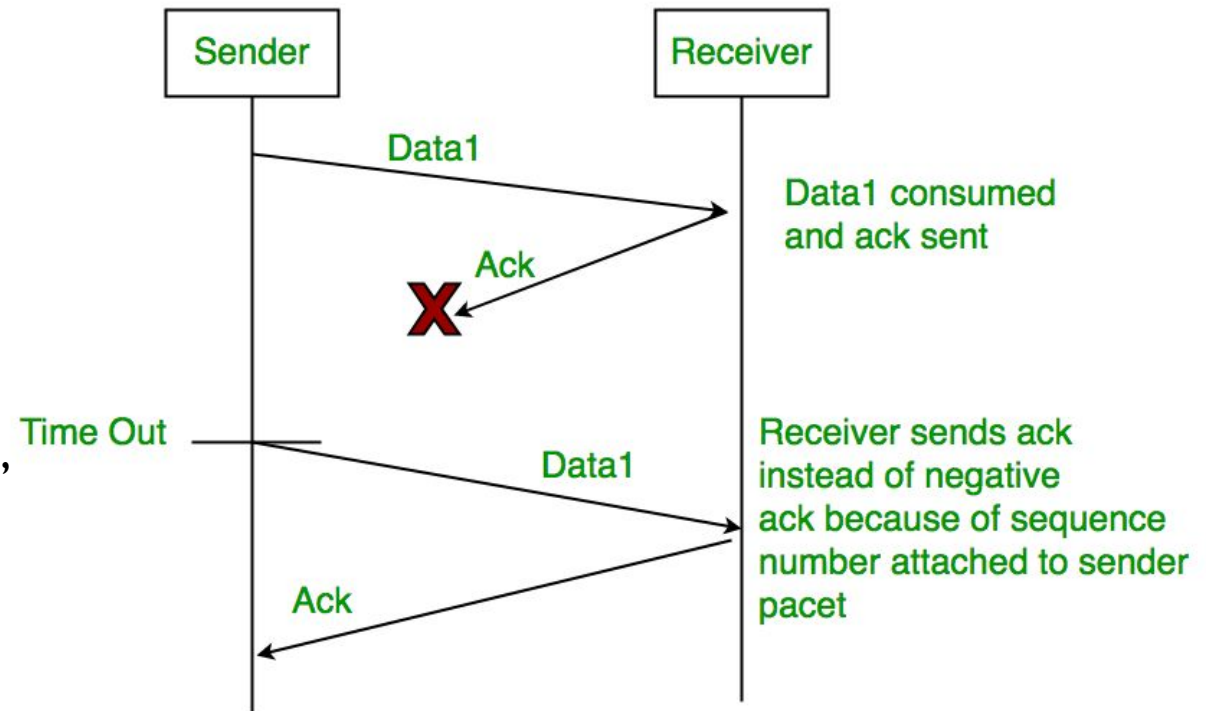
D. Synchronisation des trames:

Gestion d'acquittement:

□ Protocole Send and Wait (ARQ):

Envoi de la trame et attendre la réception d'un acquittement.

- Si celui-ci n'est pas reçu après l'expiration du temps d'attente, la trame est renvoyée.
- Si l'acquittement est reçu, la trame suivante est envoyée.



I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

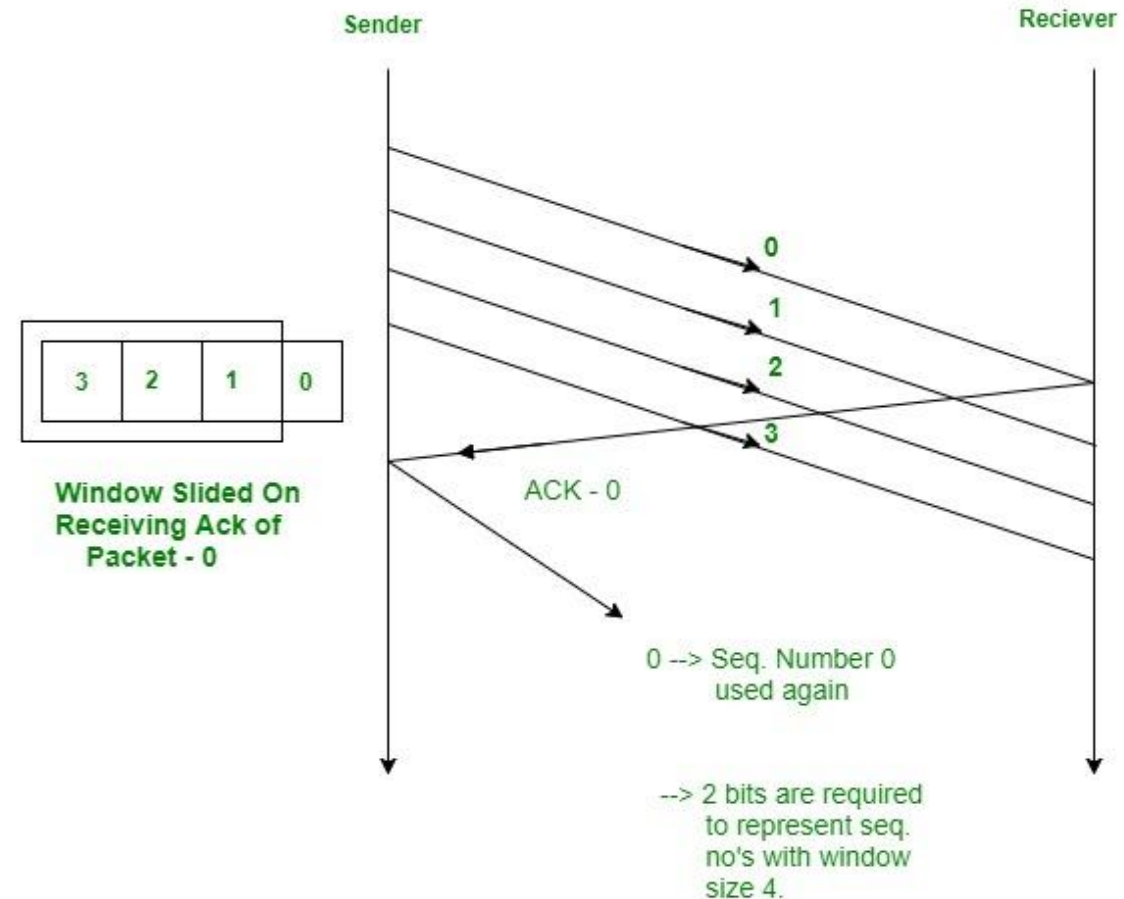
D. Synchronisation des trames:

Gestion d'acquittement:

□ Rejet simple:

Les trames sont envoyées les unes après les autres.

- Une trame arrivée sans erreur -> envoi d'acquittement
- Une trame arrivée erronée -> pas d'envoi d'acquittement et les trames suivantes sont ignorées.



I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

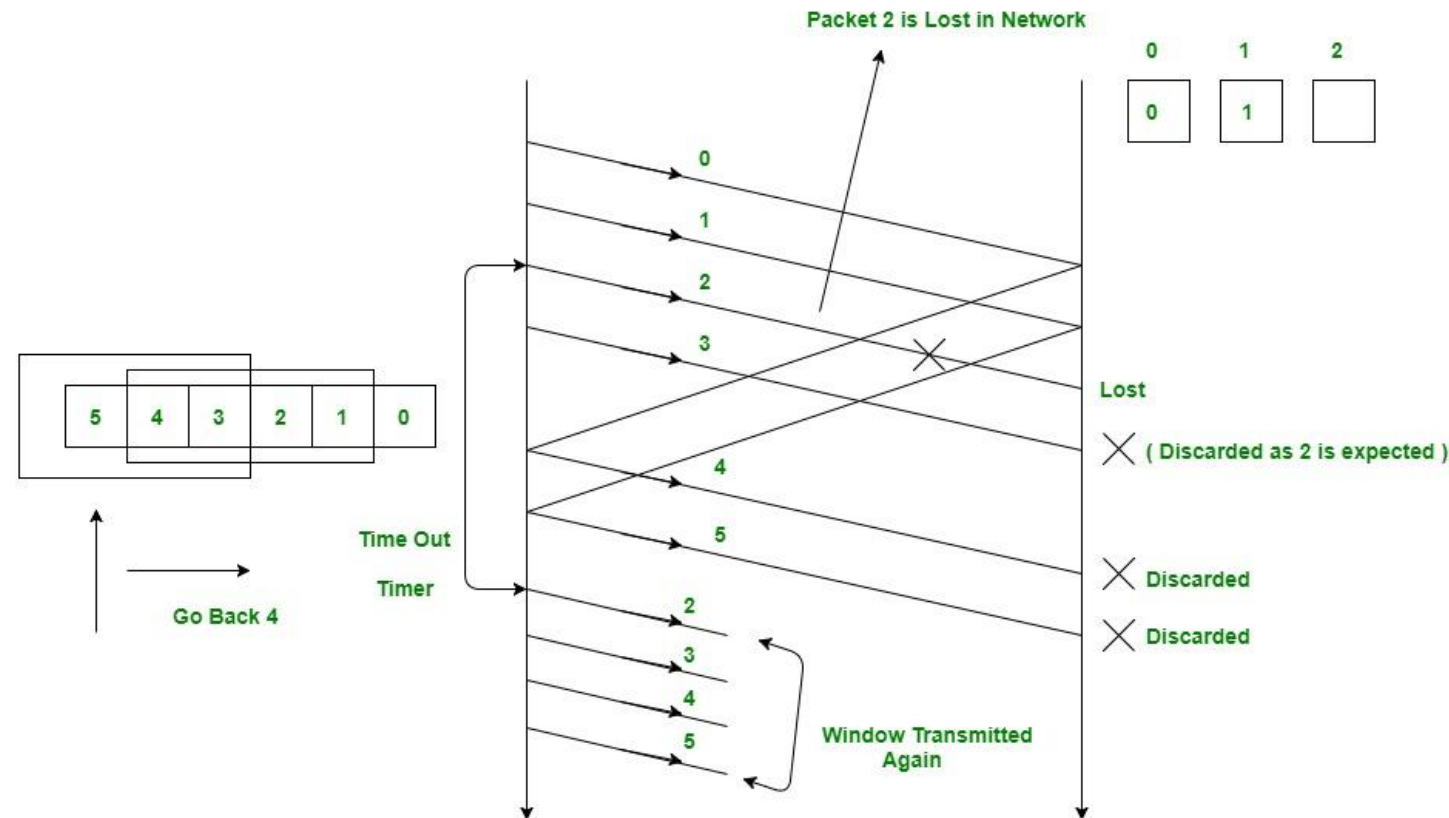
D. Synchronisation des trames:

Gestion d'acquittement:

□ Rejet simple:

Si réception d'un NACK ou échéance du temporisateur:

- Arrêt de la transmission en cours;
- La reprise depuis le bloc erroné ou perdu.



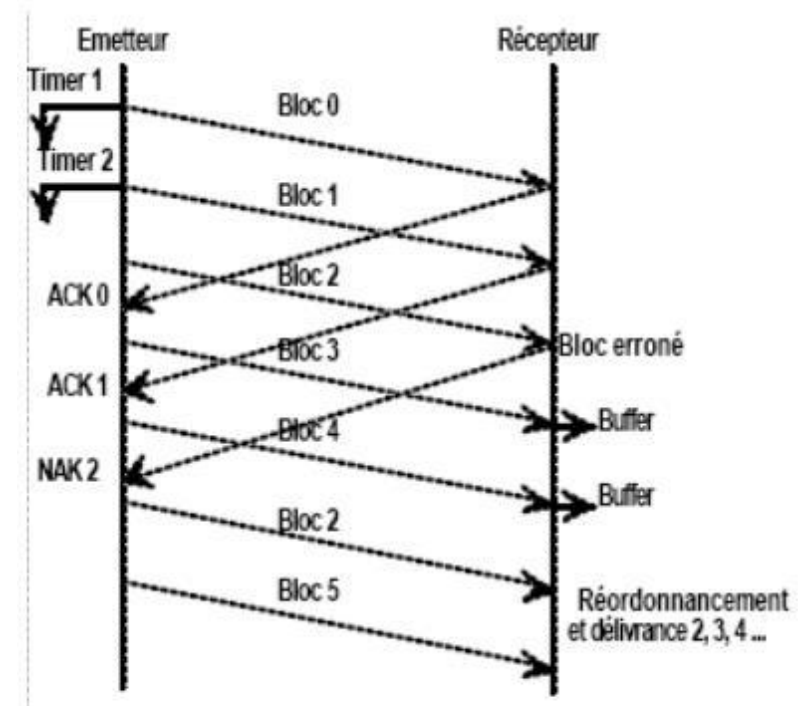
I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

D. Synchronisation des trames:

Gestion d'acquittement:

- Rejet sélectif:
 - Si trame perdue ou erronée: envoi du NACK et stockage des trames et remise en séquence après la réception de la trame manquante.
 - Seul la trame erronée est retransmis.



I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

E. Encapsulation des paquets en trames:

- Les bits à envoyer sont regroupés en trames suivant un schéma précis :

- Indicateur de début et de fin,
- Adresses MAC destination et source,
- Type PDU contenu dans la trame,
- Champs de contrôle,
- Données utiles de la trame,
- Champs de contrôle pour la détection d'erreurs.



I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

F. Protocoles d'accès multiple:

Pourquoi un protocole d'accès?

Le rôle de la couche MAC est de définir une méthode régissant l'accès à un support partagé entre plusieurs machines.

- **Méthodes statiques:** chaque station utilise pour toute une session une partie allouée des ressources du support.
- **Méthodes aléatoires:** les stations qui ont des données à émettre tentent d'accéder au support au risque de provoquer des collisions.

I. RÉSEAUX INFORMATIQUES

4) LA COUCHE LIAISON DE DONNÉES

F. Protocoles d'accès multiple:

- ALOHA: Chacun des émetteurs envoie ses paquets dès qu'il le souhaite. Si le paquet est perdu, celui-ci sera retransmis après un temps aléatoire.
 - Grand risque de collision.
- CSMA (carrier sense multiple access): une station désirant transmettre sur le canal commence par « écouter » le support pour détecter une transmission en cours. Dès qu'elle ne détecte plus de signaux, la station transmet sa trame.
 - Deux variants: CSMA/CD et CSMA/CA
 - Limite le risque de collision.

PLAN DU COURS

I. Réseaux informatiques:

- 1) Introduction sur les réseaux et Internet
- 2) Fonctionnement des réseaux
- 3) Couche physique
- 4) Couche Liaison de données
- 5) Couche réseau: adressage IP et routage**
- 6) Routage dynamique
- 7) Couche transport: TCP & UDP
- 8) Applications: DNS & HTTP

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

A. Services de la couche réseaux:

- Permettre une transmission entre deux machines pas nécessairement connectées;
- Fragmentation en paquets;
- Acheminer les paquets à travers le maillage du réseau;
- Adressage des stations et routage de l'information.

Adressage:

- Chaque interface sur un réseau internet a une adresse unique, appelée l'adresse IP;
- Il existe trois types d'adresses IP: unicast, multicast, broadcast.

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

A. Adressage:

Une adresse IP est scindée en deux parties :



- **l'adresse réseau:** elle identifie un groupe de machine, généralement regroupées sur un même sous réseau physique;
- **l'adresse machine:** elle identifie la machine dans le sous-réseau considéré.

Il existe 5 classes d'adresses IP:

- Les classes A, B et C servent à adresser des réseaux de différentes tailles
- La classe D est utilisée pour la diffusion;
- La classe E est prévue pour les évolutions futures d'Internet.

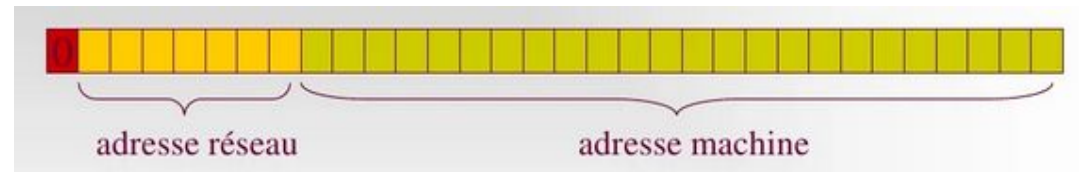
I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

A. Adressage:

Classe A:

- Une adresse de la classe A est du format R.M.M.M;
- Le premier bit est à 0
- Le premier octet est toujours inférieur à 128 (exclus)
- Exemple: 26.102.0.3



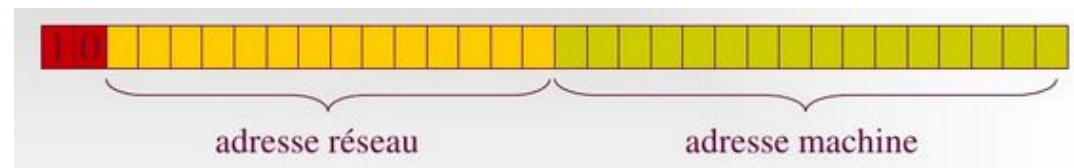
I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

A. Adressage:

Classe B:

- Une adresse de la classe B est du format R.R.M.M;
- Les deux premiers bits sont à 10
- Le premier octet est toujours compris entre 128 et 192 (exclus)
- Exemple: 128.55.7.1



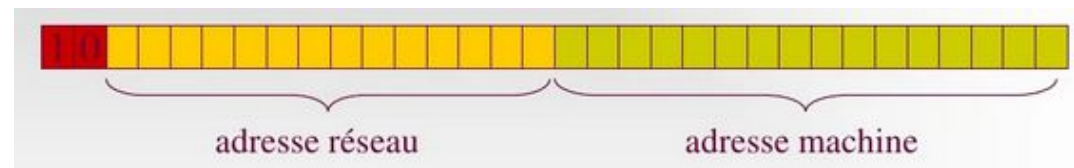
I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

A. Adressage:

Classe C:

- Une adresse de la classe C est du format R.R.R.M;
- Les trois premiers bits sont à 110
- Le premier octet est toujours compris entre 192 et 224 (exclus)
- Exemple: 196.121.56.1



I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

A. Adressage:

- A quelle classe appartiennent les adresses suivantes ?

195 . 74 . 212 . 12

100 . 150 . 25 . 3

136 . 10 . 100 . 25

Classe	Adresse inférieure	Adresse supérieure
A	0.1.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.0.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

A. Adressage:

Quelle est la capacité d'adressage de la classe C?

- En théorie on pourrait dire : 255 puisqu'en Classe C, un octet est réservé à la partie machine.
- Mais, l'adresse **R.R.R.0** est réservée pour désigner le réseau et l'adresse **R.R.R.255** est réservée pour le « broadcast »
- si dans une adresse IP, N bits sont réservés à la partie machine, on pourra installer au maximum:

$$2^N - 2$$

Classe	Adresse inférieure	Adresse supérieure
A	0.1.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.0.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

B. Sous-adressage:

- **Sous-réseau:** technique de partitionnement d'un grand réseau physique en plusieurs petits sous-réseaux logiques.
- **Avantages:** rendre le routage plus pratique, réduire la taille de l'information de la table de routage, réduire l'épuisement des adresses.
- **Masque sous-réseau:** La limite entre la partie réseau et la partie hôte est déterminée par le masque sous-réseau. Ce masque va permettre de découper un réseau en plusieurs sous-réseaux.

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

B. Sous-adressage:

- **Notation d'un masque sous-réseau:**
 - Notation décimale: 255.255.255.0, 255.0.0.0, 255.255.248.0, etc
 - Notation CIDR (classless inter domain routing): /8, /16, /22, /24, etc (0 -> 32)
- Exemple: Quelle est l'adresse réseau de l'hôte 192.168.1.1/24
 - La valeur /24 correspond au masque sous-réseau en notation CIDR, ce qui correspond à 255.255.255.0 en notation décimale.

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

B. Sous-adressage:

- Exemple: Quelle est l'adresse réseau de l'hôte 192.168.1.1/24

□ Première étape: convertir l'adresse de l'hôte et le masque en binaire:

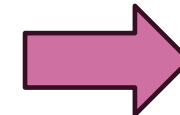
192.168.1.1 -> 11000000.10101000.00000001.00000001

255.255.255.0 -> 11111111.11111111.11111111.00000000

11000000 10101000 00000001 00000001

11111111 11111111 11111111 00000000

11000000 10101000 00000001 00000000



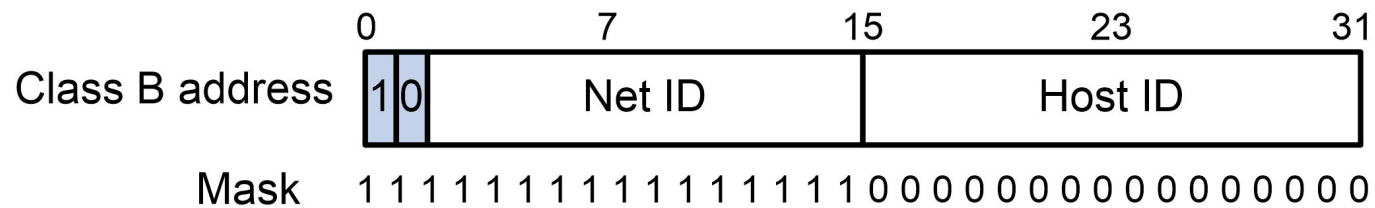
192.168.1.0/24

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

B. Sous-adressage:

- Le nombre d'adresses disponibles dans la classe B:



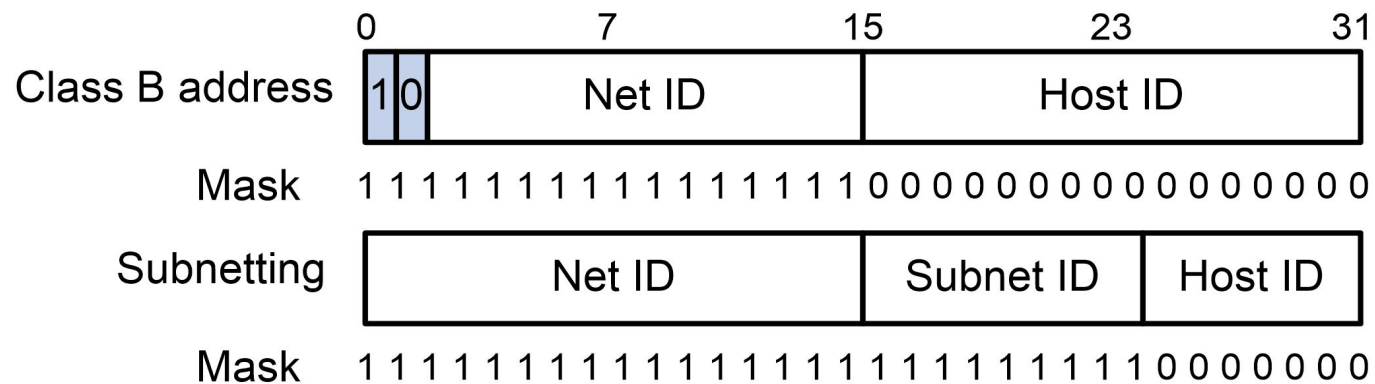
- Les deux premiers bits (10) sont fixes, donc il reste 14 bits pour identifier les réseaux.
- Avec 14 bits on peut avoir 2^{14} adresses.
- Cas général: si **N** est le nombre de bits du préfixe réseau
 - Classe A: 2^{N-1} réseaux
 - Classe B: 2^{N-2} réseaux
 - Classe C: 2^{N-3} réseaux

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

B. Sous-adressage:

- Pour identifier les sous-réseaux, des bits de la partie hôte sont utilisés :



- Comment définir le nombre de bits à utiliser pour adresser les sous-réseaux?

Pour diviser un réseau en **M** sous-réseaux, il nous faudrait **N** bits pour les adresser tel que:

$$2^N \geq M$$

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

B. Sous-adressage:

- Exemple:

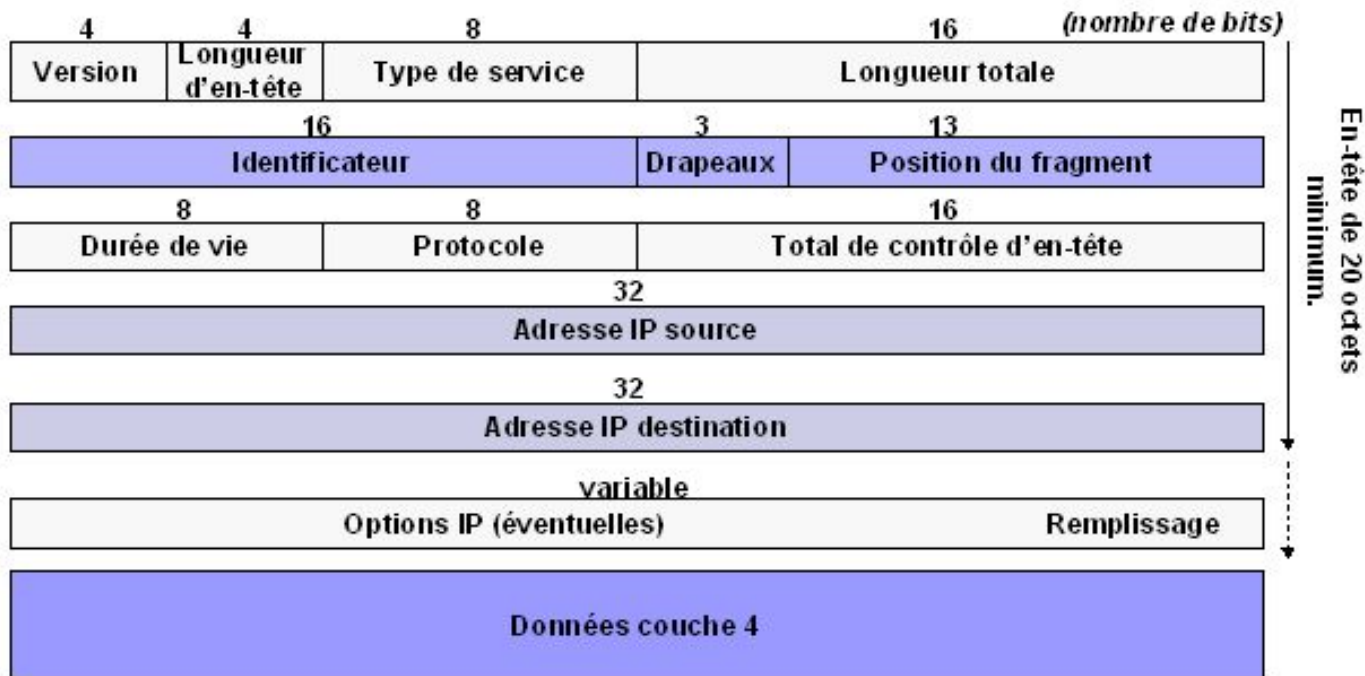
On désire découper en sous-réseau le réseau ayant l'adresse IP: **192.168.123.0**.

- Quelle est la classe de cette adresse ? Le masque réseau ? **Classe C , 255.255.255.0/24**
- Quel masque sous-réseau faudrait-il utiliser pour avoir 5 sous-réseaux? **255.255.255.224/27**
- Combien de machines peut-on avoir par sous-réseau? **30 machines**

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

C. Format d'un paquet:

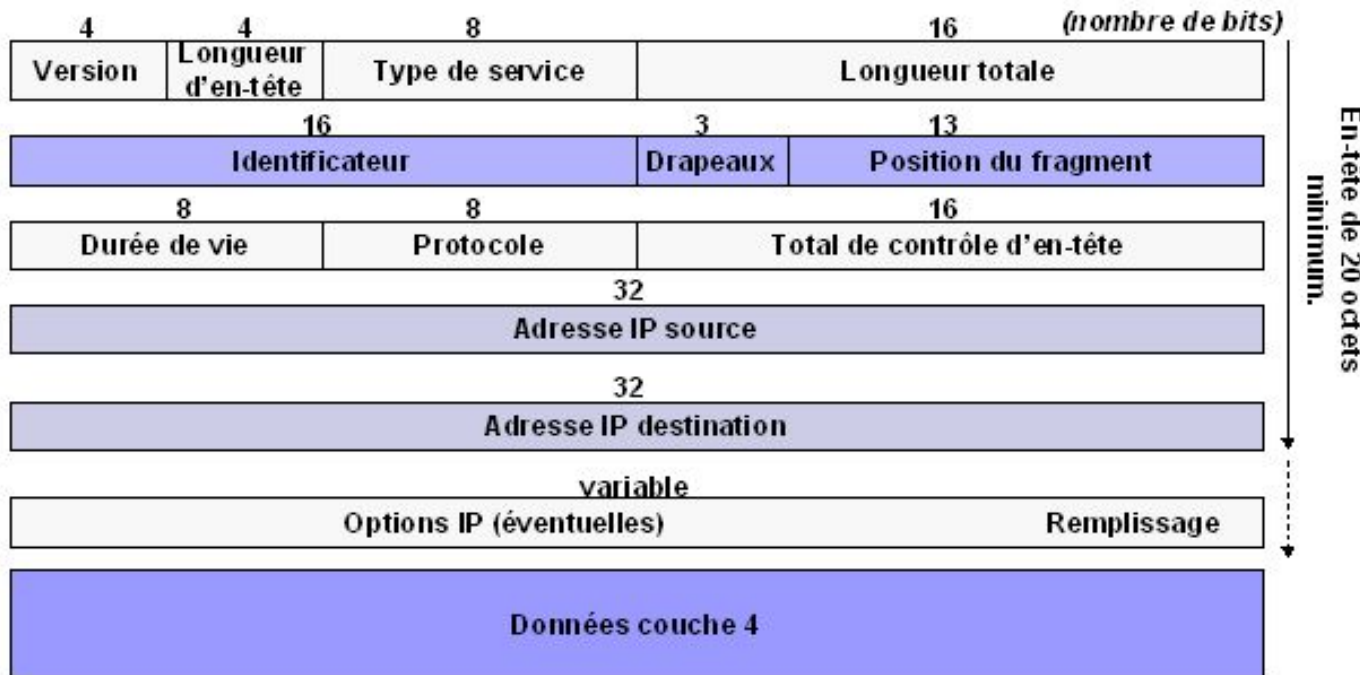


- Version: numéro de version du protocole IP (4 ou 6)
- Longueur d'en-tête: Nombre de mots (1 mot = 4 octets)
- Type de service: définit le niveau de priorité;
- Longueur totale: exprimée en octets;
- Id: identifie le paquet pour la fragmentation;
- Drapeaux: gère la fragmentation sur 3 bits « 0 DF MF »
- Position: indique la position du fragment dans le paquet courant par multiple de 8 octets ;

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

C. Format d'un paquet:



- Durée de vie (TTL): décrément de 1 à chaque passage dans un routeur (32 à 256);
- Protocole: indique le protocole utilisé par la couche supérieure (1: ICMP, 6: TCP, 17: UDP);
- Contrôle d'entête (header checksum): faire la somme des valeurs des octets de l'entête;
- @IP source et destination;
- Options: permet de mettre en œuvre des mécanismes évolués.

I. RÉSEAUX INFORMATIQUES

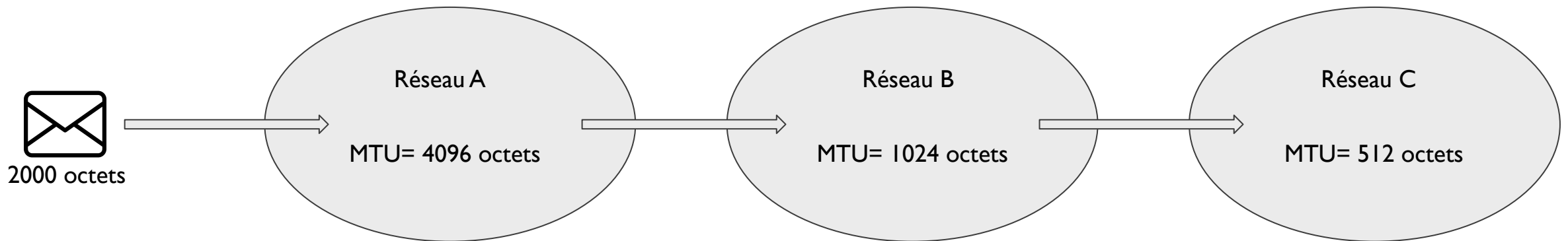
5) LA COUCHE RÉSEAUX

D. Fragmentation:

Exemple:

Un datagramme IP contenant *2000 octets* de données est émis sur un réseau A de *MTU égal à 4096*. En passant par un routeur R1, il atteint le réseau B de *MTU égal à 1024 octets*. Il passe ensuite par un routeur R2 pour atteindre un réseau C de *MTU égal à 512 octets*.

La structure de l'en-tête du datagramme dans le réseau A est présentée ci-dessous :



- MTU (Maximal Transfer Unit) : longueur totale du paquet = longueur de l'entête + longueur des données

I. RÉSEAUX INFORMATIQUES

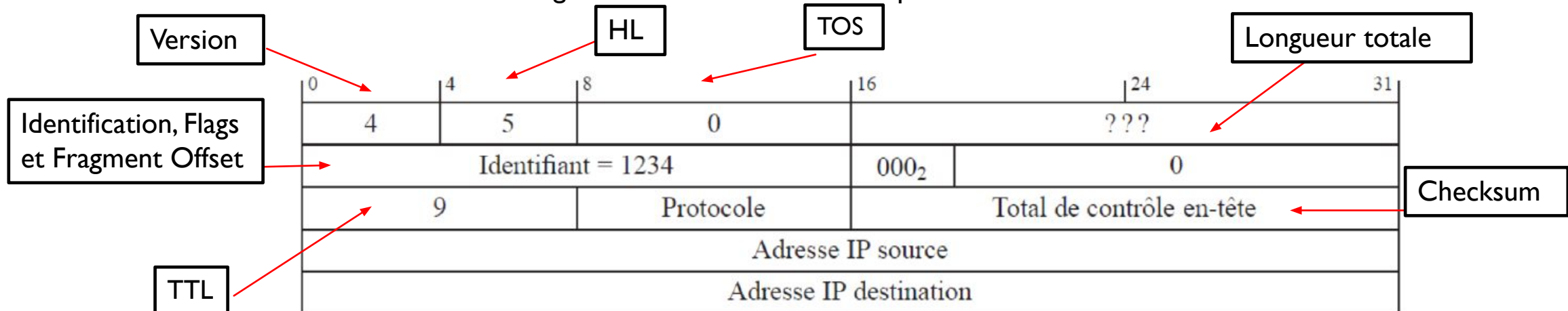
5) LA COUCHE RÉSEAUX

D. Fragmentation:

Exemple:

Un datagramme IP contenant 2000 octets de données est émis sur un réseau A de *MTU égal à 4096*. En passant par un routeur R1, il atteint le réseau B de *MTU égal à 1024 octets*. Il passe ensuite par un routeur R2 pour atteindre un réseau C de *MTU égal à 512 octets*.

La structure de l'en-tête du datagramme dans le réseau A est présentée ci-dessous :



I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

D. Fragmentation:

Exemple:

Un datagramme IP contenant *2000 octets* de données est émis sur un réseau A de *MTU égal à 4096*. En passant par un routeur R1, il atteint le réseau B de *MTU égal à 1024 octets*. Il passe ensuite par un routeur R2 pour atteindre un réseau C de *MTU égal à 512 octets*.

La structure de l'en-tête du datagramme dans le réseau A est présentée ci-dessous :

0	4	8	16	24	31
4	5	0	2020		
Identifiant = 1234			000 ₂	0	
9		Protocole		Total de contrôle en-tête	
Adresse IP source					
Adresse IP destination					

Le champ longueur totale vaut $5 \times 4 + 2000 = 2020$ octets sur le réseau A.

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

D. Fragmentation:

Exemple:

La structure de l'entête des datagrammes dans les réseaux B et C:

Réseau B

4	5	0	1020	
Identif = 1234			001b	0
8		Protocole	Total de contrôle en-tête	
Adresse IP source				
Adresse IP destination				

4	5	0	1020	
Identif = 1234			000b	125
8		Protocole	Total de contrôle en-tête	
Adresse IP source				
Adresse IP destination				

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

D. Fragmentation:

Exemple:

La structure de l'entête des datagrammes dans les réseaux B et C:

Réseau C

4	5	0	508	
Identif = 1234			001b	0
7		Protocole	Total de contrôle en-tête	
Adresse IP source				
Adresse IP destination				

4	5	0	508	
Identif = 1234			001b	61
9		Protocole	Total de contrôle en-tête	
Adresse IP source				
Adresse IP destination				

4	5	0	44	
Identif = 1234			001b	122
7		Protocole	Total de contrôle en-tête	
Adresse IP source				
Adresse IP destination				

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

D. Fragmentation:

Exemple:

La structure de l'entête des datagrammes dans les réseaux B et C:

Réseau C

4	5	0	508	
Identif = 1234			001b	125
7		Protocole	Total de contrôle en-tête	
Adresse IP source				
Adresse IP destination				

4	5	0	508	
Identif = 1234			001b	186
7		Protocole	Total de contrôle en-tête	
Adresse IP source				
Adresse IP destination				

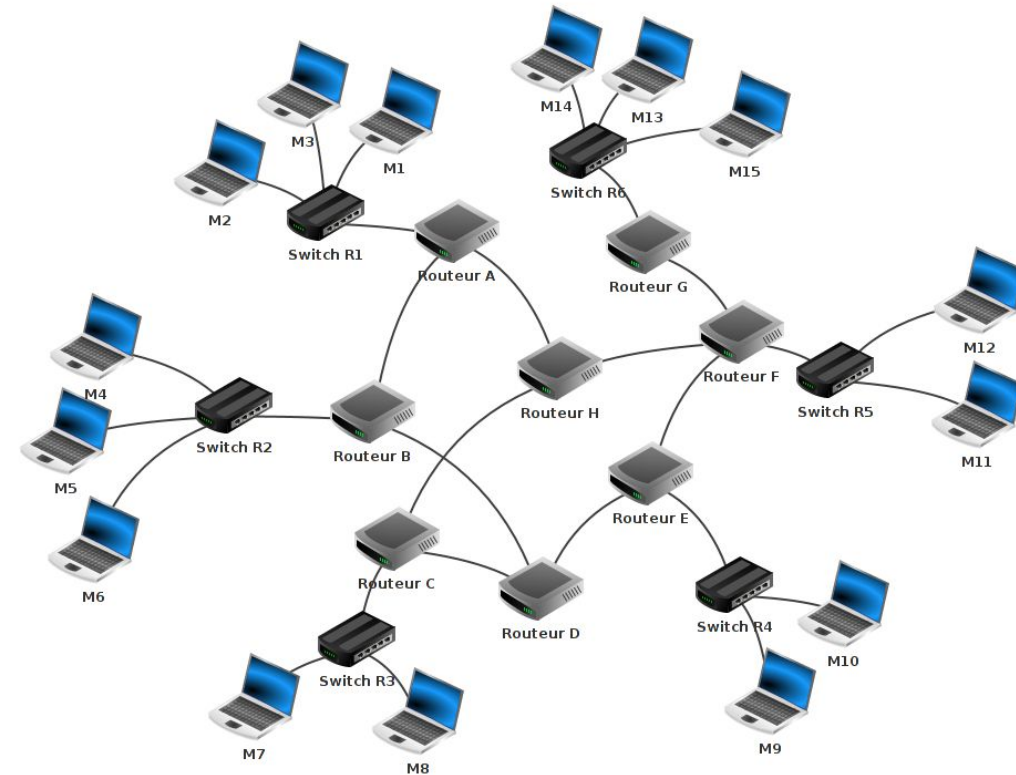
4	5	0	44	
Identif = 1234			000b	247
7		Protocole	Total de contrôle en-tête	
Adresse IP source				
Adresse IP destination				

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

- Le routage IP: l'ensemble des protocoles qui déterminent le chemin que suivent les paquets de leur source à leur destination.
- Les paquets sont acheminés à travers plusieurs réseaux par une série de routeurs.
- Les protocoles de routage permettent aux routeurs de constituer une table de routage qui met en corrélation les destinations finales avec les adresses du saut suivant.

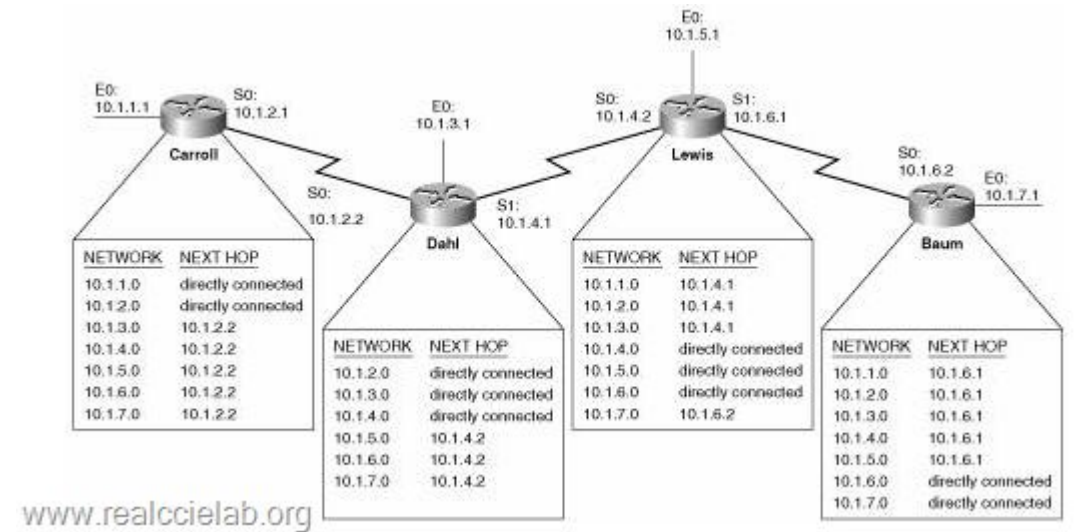


I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

- Types de routage:
 - Réseaux directement connectés: le routeur peut acheminer les paquets directement à la destination finale.
 - Les routes statiques: configurés sur le routeur par l'administrateur du réseau.
 - Les routes dynamiques: apprises d'un protocole de routage dynamique.



I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

- Exemple: On considère la figure ci-dessous. Donnez la table de routage de la passerelle C.

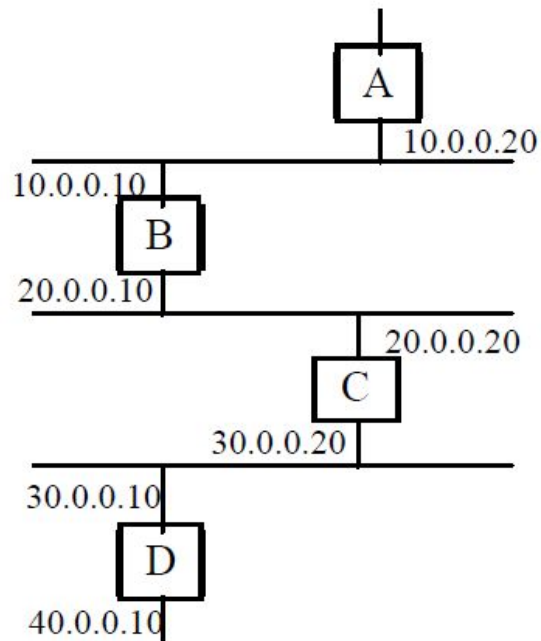


Table de routage du nœud/routeur C

Préfixe @destinataire	@routeur par où passer
10.0.0	20.0.0.10
20.0.0	Direct
30.0.0	Direct
40.0.0	30.0.0.10

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

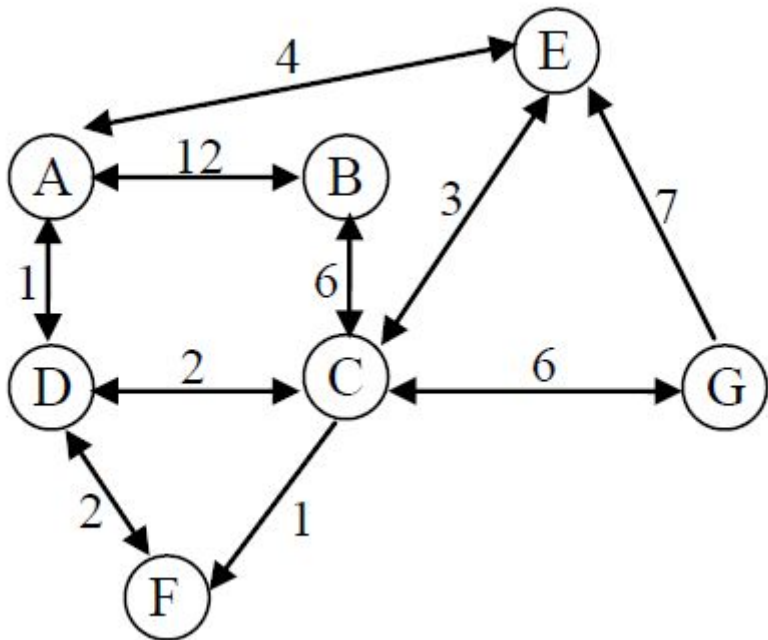
- RIP (Routing Information Protocol):
 - Chaque nœud connaît la distance vers tous ses voisins immédiats;
 - Chaque nœud échange régulièrement avec tous ses voisins ses meilleurs chemins;
 - A chaque échange, le nœud apprend de nouveaux chemins ou retient de meilleurs chemins.
- OSPF (Open Shortest Path First):
 - Diffusion de l'état de toutes les liaisons connues à tous les routeurs;
 - Tous les routeurs ont une même vision de la topologie du réseau;
 - Calcul local des meilleurs chemins entre le routeur et toutes les destinations: exemple d'algorithme: « Shortest path first » de Dijkstra.

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

- Sur le réseau suivant, montrer comment la table de routage de A est obtenue à l'aide de l'algorithme SPF (*Shortest Path First*) de Dijkstra.



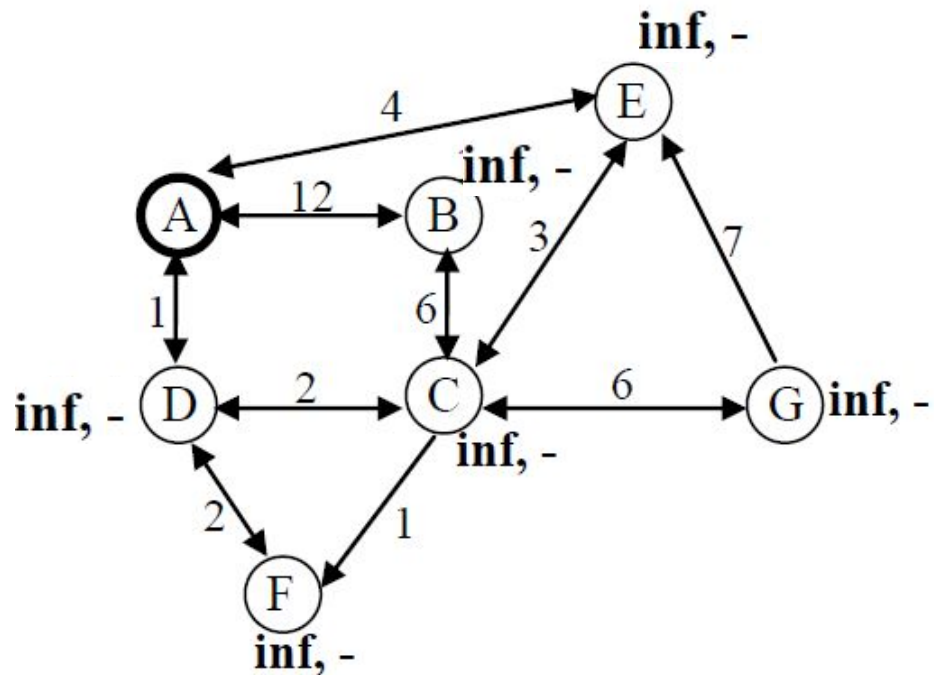
Nœud destination	Prochain nœud	Coût	Chemin
B			
C			
D			
E			
F			
G			

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

- Sur le réseau suivant, montrer comment la table de routage de A est obtenue à l'aide de l'algorithme SPF (*Shortest Path First*) de Dijkstra.



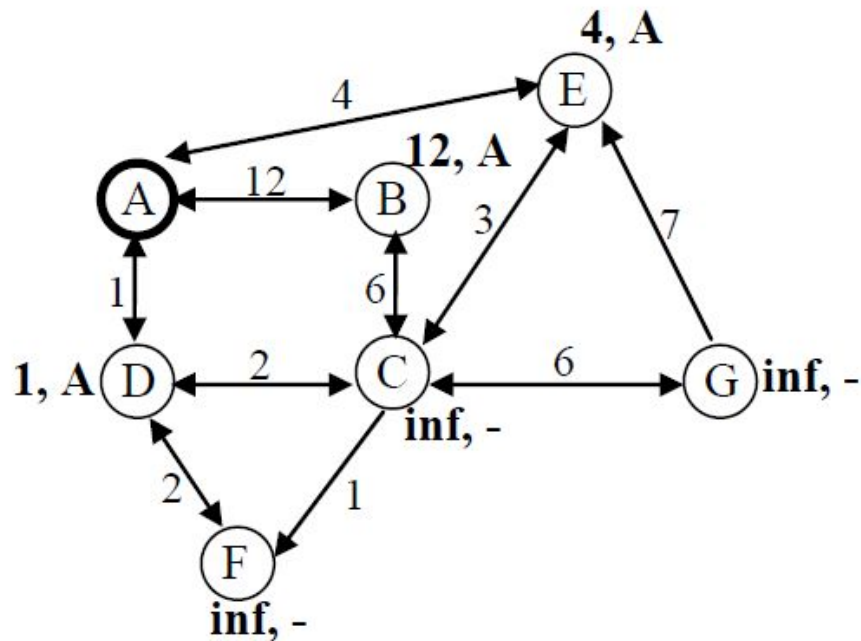
Nœud destination	Prochain nœud	Coût	Chemin
B		inf	
C		inf	
D		inf	
E		inf	
F		inf	
G		inf	

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

- Sur le réseau suivant, montrer comment la table de routage de A est obtenue à l'aide de l'algorithme SPF (*Shortest Path First*) de Dijkstra.



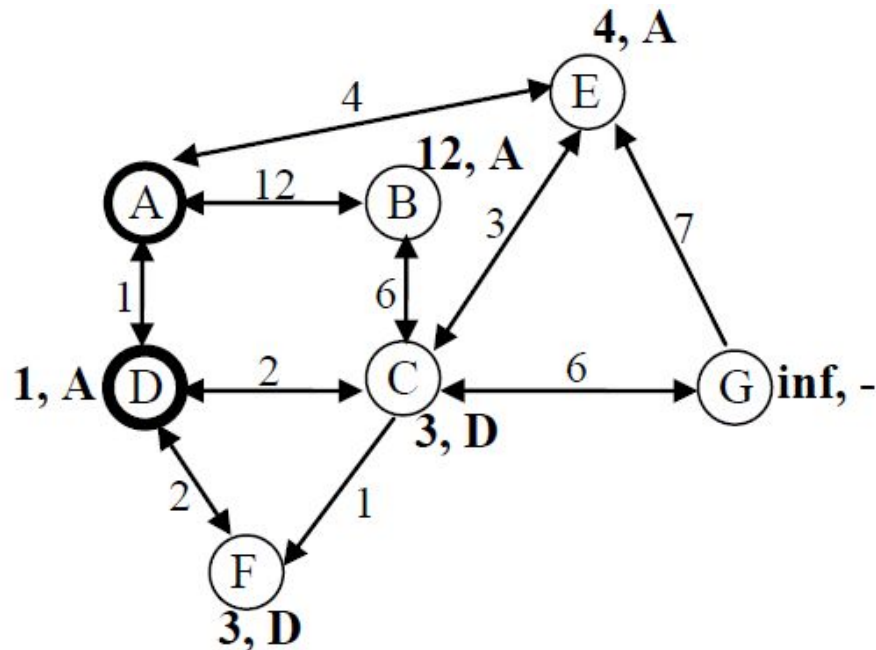
Nœud destination	Prochain nœud	Coût	Chemin
B	B	12	AB
C		inf	
D	D	1	AD
E	E	4	AE
F		inf	
G		inf	

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

- Sur le réseau suivant, montrer comment la table de routage de A est obtenue à l'aide de l'algorithme SPF (*Shortest Path First*) de Dijkstra.



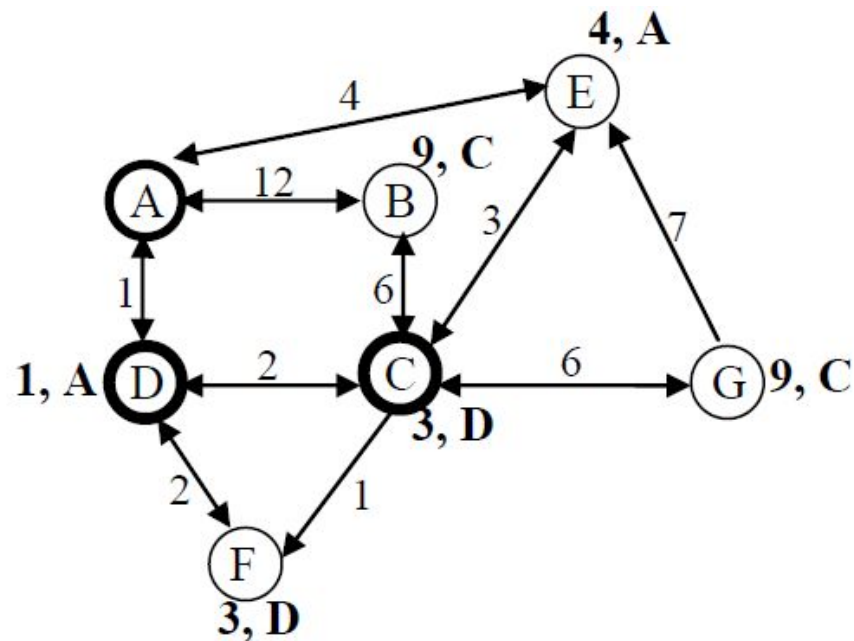
Nœud destination	Prochain nœud	Coût	Chemin
B	B	12	AB
C	D	3	ADC
D	D	1	AD
E	E	4	AE
F	D	3	ADF
G		inf	

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

E. Routage:

- Sur le réseau suivant, montrer comment la table de routage de A est obtenue à l'aide de l'algorithme SPF (*Shortest Path First*) de Dijkstra.



Nœud destination	Prochain nœud	Coût	Chemin
B	D	9	ADCB
C	D	3	ADC
D	D	1	AD
E	E	4	AE
F	D	3	ADF
G	D	9	ADCG

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

F. Protocoles:

Chaque interface possède une adresse physique (MAC) unique et une adresse logique (IP).

Il faut donc faire les correspondances entre une adresse IP et une adresse MAC.

- ARP (Address Resolution Protocol): permet de faire correspondre une adresse MAC à une adresse IP.
- RARP (Reverse Address Resolution Protocol): permet de faire correspondre une adresse IP à une adresse MAC.

I. RÉSEAUX INFORMATIQUES

5) LA COUCHE RÉSEAUX

F. Protocoles:

- ARP (Address Resolution Protocol):
 1. Une machine A veut obtenir l'adresse MAC d'une machine B;
 2. A envoie un paquet *ARP.request(MACA,IPA, 0, IPB)* ;
 3. B répond par un paquet *ARP.reply(MACA,IPA, MACB, IPB)*.

- RARP (Reverse Address: Resolution Protocol):
 1. Une machine A veut obtenir l'adresse IP d'une machine B;
 2. La machine A diffuse l'adresse MAC de la machine B;
 3. Un serveur RARP renvoie l'adresse IP correspondante.

PLAN DU COURS

I. Réseaux informatiques:

- 1) Introduction sur les réseaux et Internet
- 2) Fonctionnement des réseaux
- 3) Couche physique
- 4) Couche Liaison de données
- 5) Couche réseau: adressage IP et routage
- 6) **Couche transport: TCP & UDP**
- 7) Applications: DNS & HTTP

I. RÉSEAUX INFORMATIQUES

6) LA COUCHE TRANSPORT

A. Services:

- Fournir une communication logique entre deux applications exécutées sur deux hôtes différents.
- Fractionner le message en plusieurs segments;
- Rassembler et remettre en ordre l'ensemble des paquets provenant de la couche réseau;
- Eliminer les doublons et provoquer la réémission des trames perdues;
- Modes de communication:
 - Connecté et non connecté.
- Protocoles:
 - Internet: UDP, TCP.

I. RÉSEAUX INFORMATIQUES

6) LA COUCHE TRANSPORT

B. Le protocole UDP:

- UDP (User Datagram Protocol) : permet seulement de transmettre les paquets sans assurer la fiabilité.
 - Pas de connexion préalable ;
 - Pas de contrôle d'intégrité des données;
 - Les données ne sont envoyées qu'un fois ;
 - Les paquets arrivent à destination dans le désordre.

I. RÉSEAUX INFORMATIQUES

6) LA COUCHE TRANSPORT

C. Le protocole TCP:

- TCP (Transmission Control Protocol) : sert à établir une communication fiable entre deux hôtes. Pour cela, il assure les fonctionnalités suivantes :
 - Connexion: L'émetteur et le récepteur se mettent d'accord pour établir une connexion. La connexion reste ouverte jusqu'à ce qu'on la referme.
 - Fiabilité: des tests sont faits pour vérifier qu'il n'y a pas eu d'erreur dans la transmission.
 - Les paquets arrivent à destination dans l'ordre où ils ont été envoyés.

I. RÉSEAUX INFORMATIQUES

6) LA COUCHE TRANSPORT

C. Le protocole TCP:

- La gestion de la connexion:
 - Vérifier que l'ensemble des paquets de données constituant le message sont bien parvenus à la destination;
 - Rappeler les paquets perdus;
 - Remettre en ordre les paquets dans le cas où ils arriveraient désordonnés;
 - Éliminer les doublons;
 - au pire, générer des messages d'erreur.

Solution: connexion et vérification du séquençement des paquets

II. PROGRAMMATION RÉSEAU

I) LES SOCKETS

A. Introduction:

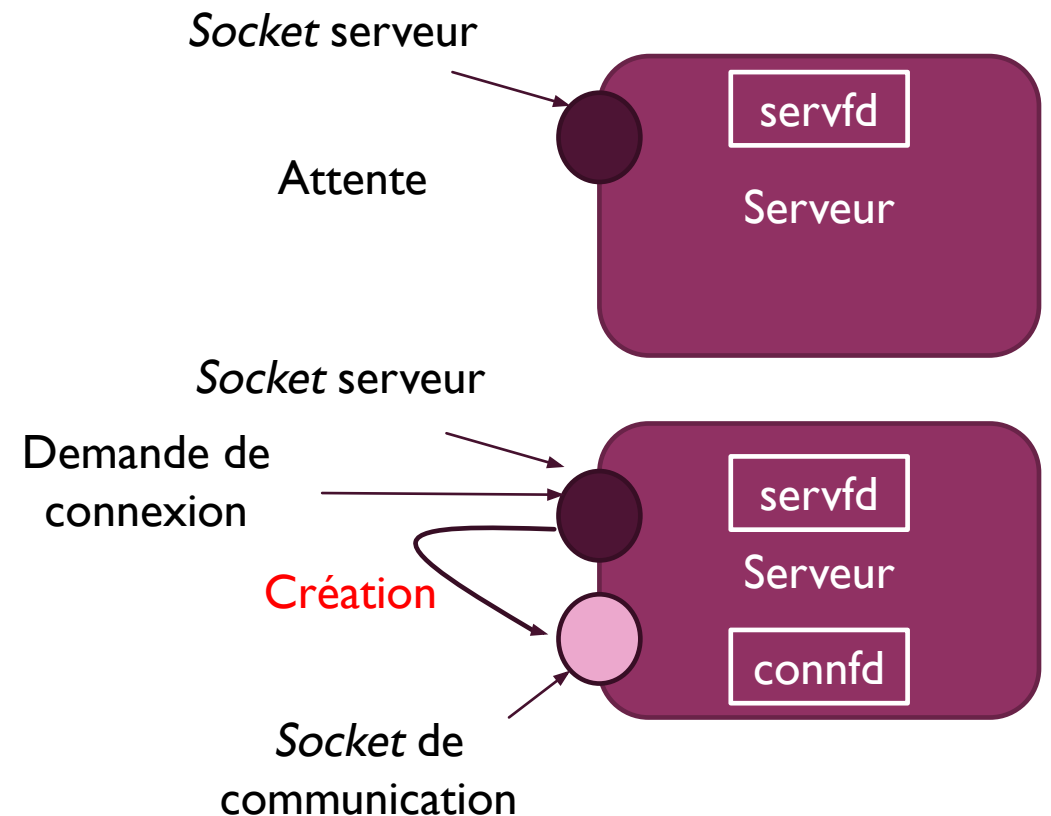
- Les sockets sont des interfaces qui servent à mettre en œuvre des connexions entre hôtes d'un réseau dans une relation client-serveur à l'aide d'une adresse IP et un port.
- Les sockets permettent la gestion de flux entrant/sortant afin d'assurer une communication entre deux hôtes, soit d'une manière fiable à l'aide du protocole TCP/IP, soit non fiable mais plus rapide avec le protocole UDP.
- Applications qu'on peut réaliser à l'aide des sockets:
 - Jeux en ligne;
 - Systèmes distribués;
 - Applications de messagerie instantanée : whatsapp, FB messenger.

II. PROGRAMMATION RÉSEAU

I) LES SOCKETS

A. Introduction:

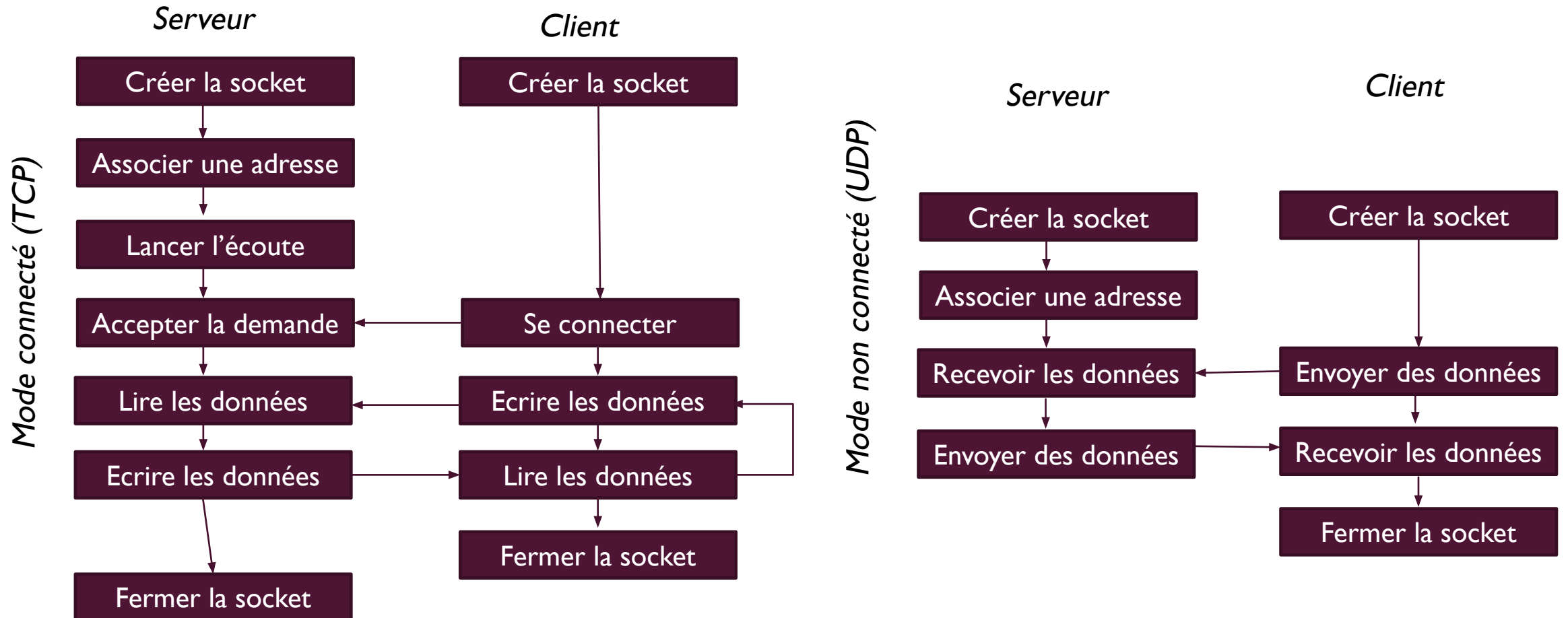
- Étape 1: Attente de nouvelle demande de connexion de la part d'un client;
- Étape 2: traitement des requêtes envoyées sur cette connexion par le client.
- Étape 3: séparation des fonctions d'attente et de traitement.



II. PROGRAMMATION RÉSEAU

I) LES SOCKETS

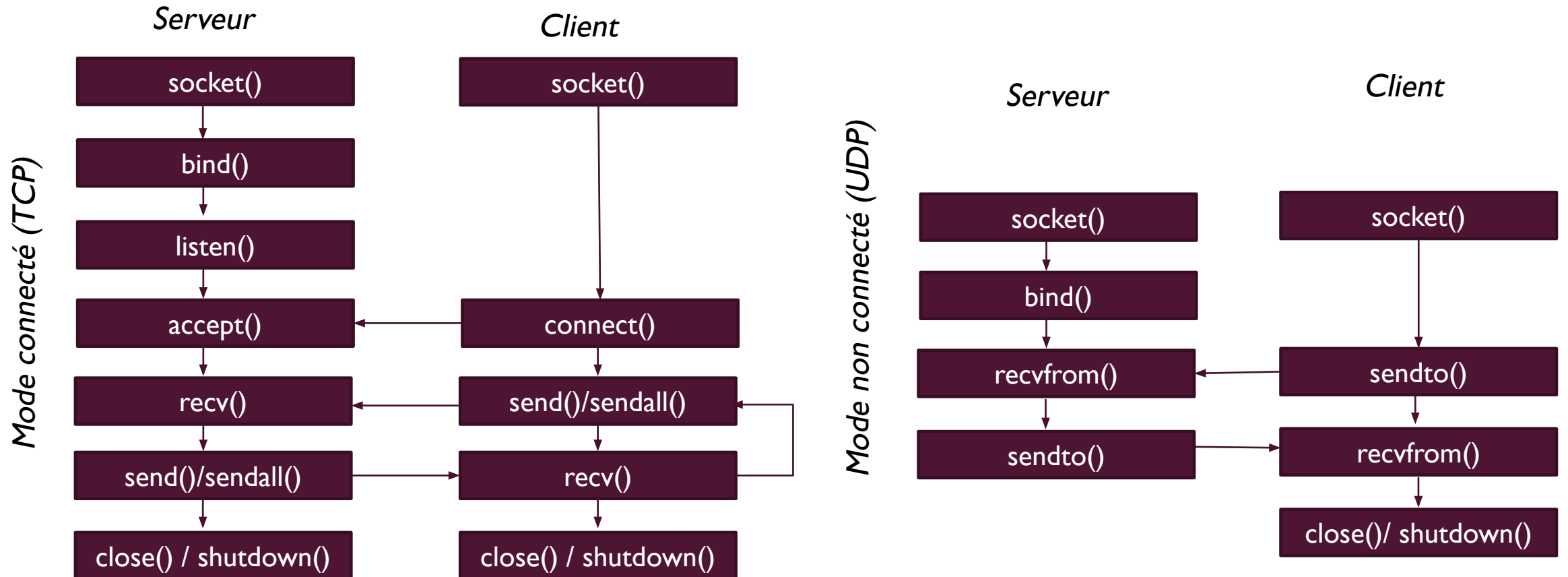
C. Une application client-serveur avec les sockets:



II. PROGRAMMATION RÉSEAU

I) LES SOCKETS

C. Une application client-serveur avec les sockets:



II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

A. Les sockets TCP côté serveur:

- Étape 1: créer une socket serveur TCP

```
socket.socket(famille, type)
```

- Le constructeur `socket()` permet de créer un objet de la classe `socket`. Les paramètres sont des constantes.
- Le paramètre *famille* indique la version du protocole IP utilisé (`AF_INET` pour IPv4 et `AF_INET6` pour IPv6).
- Le paramètre *type* indique le protocole de transport utilisé (`SOCK_STREAM` pour le protocole TCP)

```
import socket  
  
my_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

A. Les sockets TCP côté serveur:

- Étape 2: associer la socket à une adresse IP et un numéro de port TCP.

```
my_socket.bind((HOST , PORT))
```

- Le paramètre de cette méthode est un tuple (HOST, PORT).
- HOST peut être soit une adresse IP ou une chaîne de caractères vide:
 - Si seulement les processus sur la machine peuvent se connecter au serveur, l'adresse IP de l'interface loopback est utilisée (127.0.0.1).
 - Si une chaîne de caractère vide est utilisée, le serveur accepte les connexions sur toutes les interfaces IPv4.
- PORT est le numéro de port du protocole TCP (de 1 à 65535).

II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

A. Les sockets TCP côté serveur:

- Étape 3: permettre au serveur de prendre connaissance d'une nouvelle connexion.

```
my_socket.listen(backlog)
```

- La méthode `listen()` permet d'écouter les appels.
- Le paramètre `backlog` permet de fixer le nombre de connexions à accepter. S'il n'est pas spécifié, une valeur par défaut est utilisée.

II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

A. Les sockets TCP côté serveur:

- Étape 4: obtention d'un canal de communication par l'application.

```
connexion, address = my_socket.accept()
```

- Pour accepter les appels, la méthode `accept()` est utilisée.
- Cette méthode renvoie deux valeurs: une socket de communication et un tuple (HOST,PORT) contenant l'adresse IP du client et le numéro de port TCP.

II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

A. Les sockets TCP côté serveur:

- Étape 5: Réception des données.

```
data = connexion.revc(bufferize)
```

- Pour accepter les appels, la méthode `accept()` est utilisée.
- Le *bufferize* permet de définir la taille des données à recevoir en une fois.
- Cette méthode ne garantit pas la réception de toutes les données envoyées, il faut donc la rappeler plusieurs fois jusqu'à la fin de la réception.
- Si elle renvoie un objet nul ' ' ça veut dire que le client a mis fin à la connexion.

II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

A. Les sockets TCP côté serveur:

- Étape 6: Envoi des données.

```
connexion.send(data)
```

- Pour envoyer des données au client, il faut appeler la méthode `send()`.
- Comme pour `recv()`, cette méthode ne garantit pas l'envoi de toute la donnée. Il faut l'appeler plusieurs fois jusqu'à ce que toute la donnée soit transmise.
- Contrairement à `send()`, la méthode `sendall()` continue à envoyer les données jusqu'à ce que l'ensemble des données soit transmis ou en cas d'erreur. Cette méthode ne renvoie rien en cas de transmission réussie.

II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

A. Les sockets TCP côté serveur:

- Fichier serveur:

```
import socket
HOST = "127.0.0.1"
PORT = 65432
my_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
my_socket.bind((HOST, PORT))
my_socket.listen()
connexion, address = my_socket.accept()
while True:
    data = connexion.recv(1024)
    if not data:
        break;
    conn.sendall(data)
```

II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

B. Les sockets TCP côté client:

- Étape 1: créer une socket.
- Étape 2: établir une connexion avec le serveur.
- Étape 3: échange des données avec le serveur.

```
import socket
HOST = "127.0.0.1"
PORT = 65432
my_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
my_socket.connect((HOST, PORT))
my_socket.sendall(b"Hello, world")
data = my_socket.recv(1024)
```

II. PROGRAMMATION RÉSEAU

2) LES SOCKETS TCP

C. Fermeture d'une connexion:

- Fermer une connexion:

```
my_socket.shutdown(socket.SHUT_RDWR)  
my_socket.close()
```

- La méthode `shutdown()` permet de fermer la connexion dans un des deux sens (écriture ou/et lecture).
- La méthode `close()` ferme la connexion dans les deux sens et désalloue la socket.

II. PROGRAMMATION RÉSEAU

3) LES SOCKETS UDP

- Création d'une socket côté serveur et client:

```
my_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
```

- Pour le mode non connecté, on utilise la valeur *SOCK_DGRAM* pour le paramètre *type*.

- Réception de données:

```
(data, addr) = my_socket.recvfrom(buffer_size)
```

- Emission de données:

```
my_socket.sendto(data, addr)
```