REPUBLIQUE LAGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE FERHAT ABBAS - SETIF-1
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE

SUPPORT DU COURS

RESEAUX DE COMMUNICATION

2eme année Licence Informatique

Enseignante : KHARCHI Samia

ANNEE UNIVERSITAIRE: 2019/2020

INTRODUCTION AUX RESEAUX INFORMATIQUES

1. DEFINITION

Un réseau informatique (computer network) est un système de communication (ensemble matériel + logiciel) qui permet à un ensemble d'ordinateurs (au sens large) d'échanger des informations.
- sens large : points d'accès, terminaux de paiement, téléphones, capteurs divers, etc.
- L'échange d'information : Les réseaux servent avant tout à réaliser des services accessibles à partir de tout organe connecté au réseau mis en œuvre par un ensemble d'ordinateurs sur le réseau

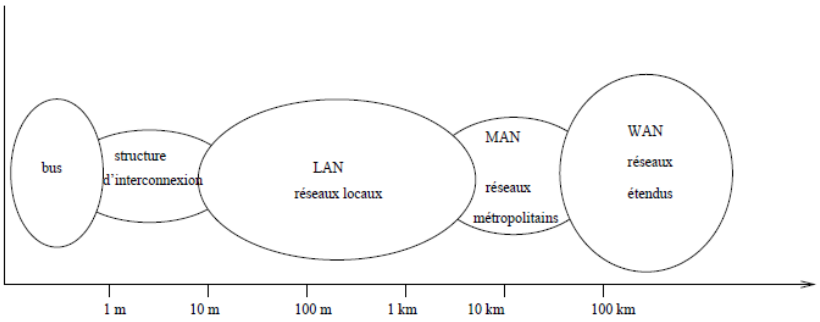
exemples de services

- le courrier électronique (mail)
- le transfert de fichiers
- l'accès à distance
- l'accès au World Wide Web
- les services utilisant le Web :documentation, commerce électronique, etc.

2. CLASSIFICATION DES RESEAUX

Les réseaux peuvent être classés selon différents critères

- On peut faire une première classification des réseaux à l'aide de leur taille comme on peut le voir dans la figure suivante:



- **Les bus** que l'on trouve dans un ordinateur pour relier ses différents composants (mémoires, périphériques d'entrée-sortie, processeurs, ...) peuvent être considérés comme des réseaux dédiés à des tâches très spécifiques.

- **Les structures d'interconnexion** sont des réseaux de très haut débits, mais de faible étendue, et regroupent les pré et post-processeurs des ordinateurs vectoriels par exemple.

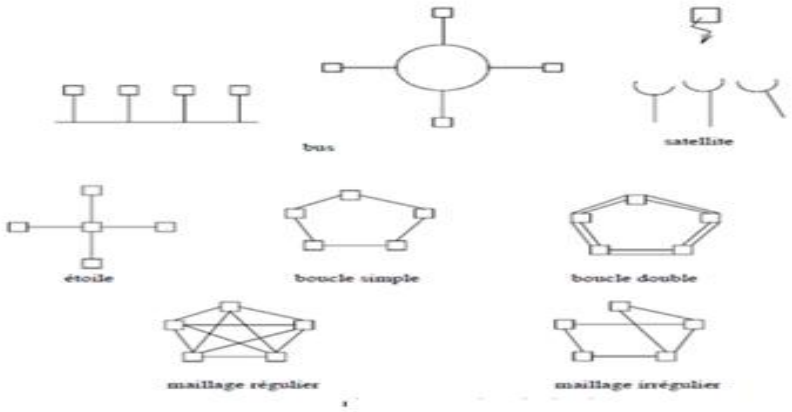
- **Réseaux locaux (Local Area Networks, LAN)**
  - Communication au sein d'une organisation (département d'entreprise, etc.)
  - Couverture géographique limitée (~1 km)
  - Débit élevé, taux d'erreur faible

- **Réseaux à grande distance (Wide Area Networks, WAN)**
  - Communication entre des organisations diverses
  - Couverture géographique étendue : un pays, toute la planète
  - Débit variable, taux d'erreur parfois non négligeable
  - Les infrastructures physiques pouvant être terrestres ou spatiales à l'aide de satellites de télécommunications.

- **Réseaux métropolitains (Metropolitan Area Networks, MAN)**
  - Intermédiaires entre LAN et WAN
  - quelques dizaines de km, ville ou région

- par exemple les différents sites d'une université ou d'une administration, chacun possédant son propre réseau local.

- On peut également différencier les réseaux selon leur structure ou plus précisément leur topologie comme illustré dans la figure suivante:



On y distingue ainsi deux classes de réseaux :

- ceux en mode de diffusion
- ceux en mode point à point

Le premier mode de fonctionnement consiste à partager un seul support de transmission. Chaque message envoyé par un équipement sur le réseau est reçu par tous les autres. C'est l'adresse spécifique placée dans le message qui permettra à chaque équipement de déterminer si le message lui est adressé ou non. À tout moment un seul équipement à le droit d'envoyer un message sur le support, il faut donc qu'il "écoute" au préalable si la voie est libre; si ce n'est pas le cas il attend selon un protocole spécifique à chaque architecture.

Les réseaux locaux adoptent pour la plupart le mode diffusion sur une architecture en **bus.** Les réseaux **satellitaires** ou **radio** suivent également ce mode de communication.

Dans une telle configuration la rupture du support provoque l'arrêt du réseau, par contre la panne d'un des éléments ne provoque pas (en général) la panne globale du réseau.

Dans le mode point à point, le support physique (le câble) relie une paire d'équipements seulement. Quand deux éléments non directement connectés entre eux veulent communiquer ils le font par l'intermédiaire des autres nœuds du réseau.

Dans le cas de **l'étoile** le site central reçoit et envoie tous les messages, le fonctionnement est simple, mais la panne du nœud central paralyse tout le réseau.

Dans une **boucle (anneau) simple**, chaque nœud recevant un message de son voisin en amont le réexpédie à son voisin en aval. Pour que les messages ne tournent pas indéfiniment le nœud émetteur retire le message lorsqu'il lui revient.

Si l'un des éléments du réseau tombe en panne, alors tout s'arrête. Ce problème est partiellement résolu par la **double boucle** dont chacune des boucles fait tourner les messages dans un sens opposé. En cas de panne d'un équipement, on reconstitue une boucle simple avec les éléments actifs des deux boucles, mais dans ce cas tout message passera deux fois par chaque nœud.

Dans le **maillage régulier** l'interconnexion est totale, ce qui assure une fiabilité optimale du réseau, par contre c'est une solution coûteuse en câblage physique. Si l'on allège le plan de câblage, **le maillage** devient **irrégulier** et la fiabilité peut rester élevée.

3. LES EQUIPEMENTS D'INTERCONNEXION

Un réseau sert à interconnecter les ordinateurs d'une organisation, toutefois une organisation comporte généralement plusieurs réseaux locaux, il est donc parfois indispensable de les relier entre eux. Dans ce cas, des équipements spécifiques sont nécessaires. Ainsi,les équipements à mettre en œuvre sont différents selon la configuration face à laquelle on se trouve.

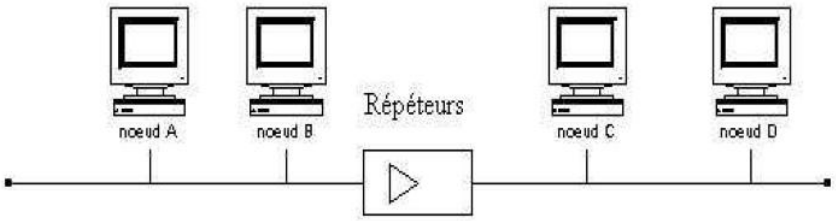
A/ Les équipements

1. Les répéteurs :

Ces systèmes permettent l'interconnexion de médias similaires ou différents pour une méthode d'accès donnée et assurent ainsi une continuité de la topologie physique pour constituer un réseau local unique.

C'est le matériel de plus bas niveau sur le réseau local. Il n'interprète pas les trames qu'il reçoit et se contente de les retransmettre bit à bit sur les autres segments.

La principale fonction du répéteur est régénération du signal: en effet le signal subit une atténuation tout au long de sa propagation dans le câble. Le répéteur émet les signaux reçus en les remettant en forme.

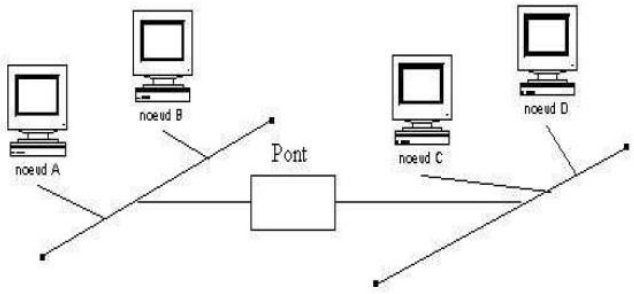


Interconnexion par répéteur

2. Les ponts (Bridge) :

Ils réalisent la connexion entre deux réseaux locaux de type différent. Ils permettent de gérer des liaisons locales ou distantes pour réaliser l'interconnexion des réseaux et optimiser les flux de communications.

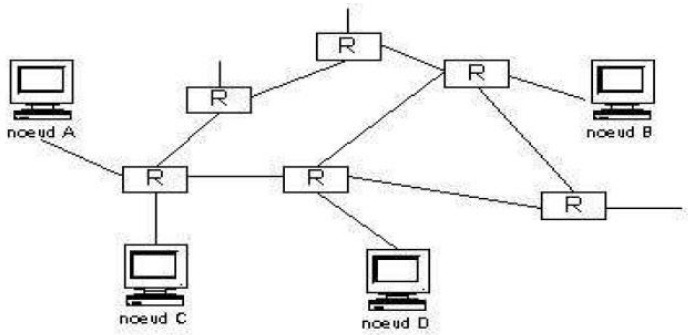
Au niveau de sécurité les ponts sont plus puissants que le répéteur car ils peuvent éviter la propagation de certains défauts, en plus ils filtrent les trames entre deux réseaux.



Interconnexion par pont

3. Les routeurs :

Les routeurs sont des éléments actifs qui permettent d'interconnecter localement ou à distance des réseaux entre eux. Ils proposent un certain nombre de mécanismes évolués permettant de déterminer le meilleur chemin pour assurer l'acheminement de l'information.



Interconnexion par routeur

4. Les Hubs (concentrateurs) :

Les Hubs (concentrateurs) permettent la connexion de plusieurs nœuds sur un même point d'accès sur le réseau, en se partageant la bande passante totale.

La structure physique qui s'en dégage est une étoile, mais la topologie logique reste un bus.

5. Le commutateur (ou switch) :

Le commutateur (switch) est un système assurant l'interconnexion de stations ou de segments d'un réseau local en leur attribuant l'intégralité de la bande passante, à l'inverse du concentrateur qui la partage.

Les commutateurs ont donc été introduits pour augmenter la bande passante globale d'un réseau d'entreprise et sont une évolution des concentrateurs (hubs).

6. La passerelle (Gateway) :

C'est un système complet du point de vue de la connexion puisqu'elle ressemble au routeur qui assure l'interconnexion des réseaux hétérogènes, elle fait de la conversion de protocole. C'est la seule qui travaille jusqu'à un haut niveau de protocole.

7. Le modem (modulateur-démodulateur):

C'est un équipement qui sert à lier le réseau téléphonique au réseau informatique. Souvent pour transmettre des données informatiques à distance, on utilise la ligne téléphonique comme support de transmission. Et comme nous savant que la ligne téléphonique ne transporte que des signaux analogiques et que les réseaux informatiques n'utilisent que des signaux numériques, le modem a pour rôle de convertir le signal numérique en signal analogique et vice versa.

B. Les supports de transmission

Nous décrivons brièvement quelques uns des supports de transmission les plus usités.

1. la paire torsadée est un câble téléphonique constitué à l'origine de deux fils de cuivre isolés et enroulés l'un sur l'autre (d'où le nom). Elle est très répandue de connexion facile et d'un faible coût mais elle possède une faible immunité aux bruits. Les performances (et les coûts) de ce support dépendent de la qualité des matériaux employés et des détails de réalisation.

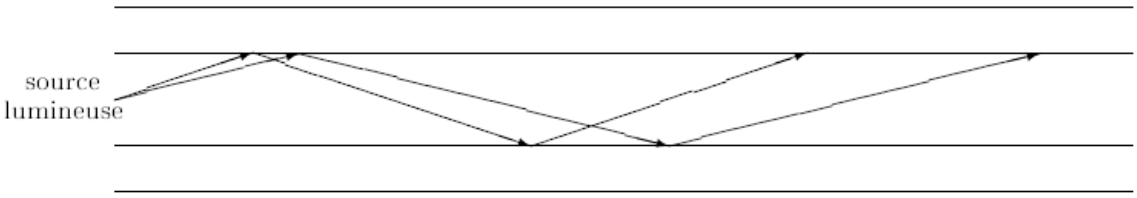
Dans le type de réseaux utilisant les paires torsadées, chaque poste est relié à un **hub**, par une liaison point à point, formant physiquement une étoile (dont le centre est un hub), mais dont le fonctionnement est en mode de diffusion de type bus. Cependant l'orientation actuelle est de remplacer les hubs par des **commutateurs** qui eux réalisent de la diffusion en mode point à point.

2. le câble coaxial est un câble utilisé également en téléphonie et en télévision, il est constitué d'un cœur qui est un fil de cuivre. Ce cœur est dans une gaine isolante elle-même entourée par une tresse de cuivre, le tout est recouvert d'une gaine isolante. Certains coaxiaux " large bande" peuvent atteindre un débit maximal de 150 Mhz mais son encombrement est nettement supérieur à celui de

la paire torsadée et ses performances n'atteignant pas celle de la fibre optique, il a tendance à disparaître des nouveaux plans de câblage.

3. **la fibre optique** est un support d'apparition plus récente mais son utilisation prend de l'ampleur de jour en jour car elle permet(tra) des débits de plusieurs Gbit/s sur de très longues distances. Elle est particulièrement adaptée à l'interconnexion de réseaux par exemple entre plusieurs bâtiment d'un même site.

D'un point de vue technique une fibre optique est constituée d'un cœur et d'une gaine en silice de quelques µm recouvert d'un isolant. À une extrémité une diode électroluminescente (LED) ou une diode laser émet un signal lumineux et à l'autre une photodiode ou un phototransistor est capable de reconnaître ce signal.



Les différents rayons lumineux issus de la source sont guidés par le fil de verre en suivant un principe de réflexion interne qui se produit au niveau de la frontière entre le cœur et la gaine comme illustré dans la figure. Si la réflexion ne laisse subsister qu'un seul rayon, car le diamètre du fil est très réduit, alors on parle de fibre monomode sinon, lorsqu'il existe plusieurs rayons simultanément on parle de fibre multimode. Enfin, la bande passante d'une fibre optique étant très large (plusieurs MHz) il est aisé de faire du multiplexage fréquentiel pour faire transiter simultanément plusieurs communications.

4. **les liaisons sans fil** sont possibles grâce à des liaisons infrarouges ou laser sur de courtes distances et grâce aux faisceaux hertziens pour les liaisons satellitaires. Les débits sont très élevés mais les transmissions sont sensibles aux perturbations et les possibilités d'écoute sont nombreuses.

FONCTIONNEMENT DES RESEAUX

1. Modes de fonctionnement

Quelle que soit l'architecture physique d'un réseau on trouve deux modes de fonctionnement différents :

- avec connexion
- sans connexion

- Dans le mode **avec connexion**, toute communication entre deux équipements suit le processus suivant:

1. l'émetteur demande l'établissement d'une connexion par l'envoi d'un bloc de données spécial
2. si le récepteur (ou le gestionnaire de service) refuse cette connexion la communication n'a pas lieu
3. si la connexion est acceptée, elle est établie par mise en place d'un circuit virtuel dans le réseau reliant l'émetteur au récepteur
4. les données sont ensuite transférées d'un point à l'autre
5. la connexion est libérée

C'est le fonctionnement bien connu du réseau téléphonique classique.

Les avantages du mode avec connexion sont la sécurisation du transport par identification claire de l'émetteur et du récepteur. Les défauts sont la lourdeur de la mise en place de la connexion qui peut se révéler beaucoup trop onéreuse si l'on ne veut échanger que quelques octets ainsi que la difficulté à établir des communications multipoint.

- Dans le mode **sans connexion** les blocs de données, appelés datagrammes, sont émis sans vérifier à l'avance si l'équipement à atteindre, ainsi que les nœuds intermédiaires éventuels, sont bien actifs.

C'est alors aux équipements gérant le réseau d'acheminer le message étape par étape et en assurant éventuellement sa temporisation jusqu'à ce que le destinataire soit actif. Ce service est celui du courrier postal classique et suit les principes généraux suivants:

- le client poste une lettre dans une boîte aux lettres
- chaque lettre porte le nom et l'adresse du destinataire
- chaque client a une adresse propre et une boîte aux lettres
- le contenu de l'information reste inconnu du prestataire de service
- les supports du transport sont inconnus de l'utilisateur du service

2. Différentes techniques de commutation :

Le réseau doit permettre l'échange de messages entre les abonnés quelle que soit leur localisation.

Il existe 4 techniques de commutation :

• **Commutation de circuits** (ex. : le téléphone). Un chemin physique est établi à l'initialisation de la communication entre l'émetteur et le récepteur et reste le même pendant toute la durée de la communication. Si les deux correspondants n'ont pas de données à transmettre pendant un certain temps, la liaison restera inutilisée. L'idée est de concentrer plusieurs correspondants sur une même liaison. Dans le cas où les communications seraient nombreuses, il faut prévoir des mémoires pour stocker des informations en attendant que la liaison soit disponible.

• **Commutation de messages** : Un message est un ensemble d'information logique formant un tout (fichier, mail) qui est envoyé de l'émetteur vers le récepteur en transitant nœud à nœud à travers le réseau. On a un chemin logique par message envoyé. Le message ne peut être envoyé au nœud

suivant tant qu'il n'est pas reçu complètement et sans erreur par le nœud actuel. Dans cette approche il devient très difficile de transmettre de longs messages.

• **Commutation de paquets** : optimisation de la commutation de message qui consiste à découper les messages en plusieurs paquets pouvant être acheminés plus vite et indépendamment les uns des autres. Cette technique nécessite la mise en place de la numérotation des paquets.

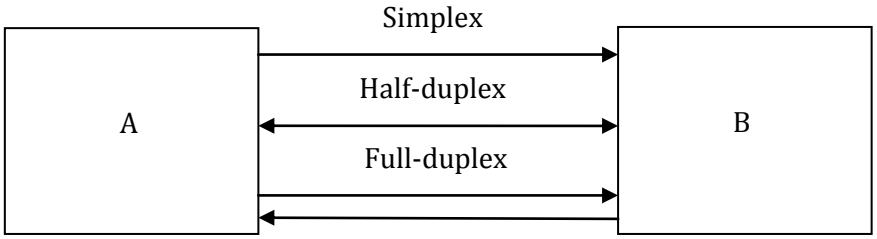
• **la commutation de cellules** : une cellule est un paquet particulier dont la taille est toujours fixée à 53 octets (5 octets d'en-tête et 48 octets de données). C'est la technique de base des réseaux hauts débits ATM (Asynchronous Transfert Mode) qui opèrent en mode connecté où avant toute émission de cellules, un chemin virtuel est établi par lequel passeront toutes les cellules. Cette technique mixe donc la commutation de circuits et la commutation de paquets de taille fixe permettant ainsi de simplifier le travail des commutateurs pour atteindre des débits plus élevés. Commutation de cellule= superposition de 2 types de commutation : commutation de circuit + commutation de paquets.

3. Types de liaison

Une liaison entre 2 équipements A et B peut être simplex (unidirectionnelle), dans ce cas A est toujours l'émetteur et B le récepteur. C'est ce que l'on trouve par exemple entre un banc de mesure et un ordinateur recueillant les données mesurées.

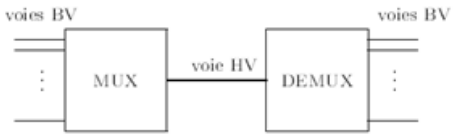
La communication est half-duplex (bidirectionnelle à l'alternat) quand le rôle de A et B peut changer, la communication change de sens à tour de rôle (comme avec des talkies-walkies).

Elle est full-duplex (bidirectionnelle simultanée) quand A et B peuvent émettre et recevoir en même temps (comme dans le cas du téléphone).



4. Le multiplexage:

Le multiplexage consiste à faire transiter sur une seule et même ligne de liaison, dite voie haute vitesse, des communications appartenant à plusieurs paires d'équipements émetteurs et récepteurs comme représenté dans la figure:



Chaque émetteur (resp. récepteur) est raccordé à un multiplexeur (resp. démultiplexeur) par une liaison dite voie basse vitesse.
Plusieurs techniques sont possibles :

4.1. **le multiplexage fréquentiel** consiste à affecter à chaque voie basse vitesse une bande passante particulière sur la voie haute vitesse en s'assurant qu'aucune bande passante de voie basse vitesse ne se chevauche. Le multiplexeur prend chaque signal de voie basse vitesse et le réémet sur la voie haute vitesse dans la plage de fréquences prévues. Ainsi plusieurs transmissions peuvent être faites

simultanément, chacune sur une bande de fréquences particulières, et à l'arrivée, le démultiplexeur est capable de discriminer chaque signal de la voie haute vitesse pour l'aiguiller sur la bonne voie basse vitesse.

4.2. **le multiplexage temporel** partage dans le temps l'utilisation de la voie haute vitesse en l'attribuant successivement aux différentes voies basse vitesse même si celles-ci n'ont rien à émettre. Suivant les techniques, chaque intervalle de temps attribué à une voie lui permettra de transmettre 1 ou plusieurs bits.

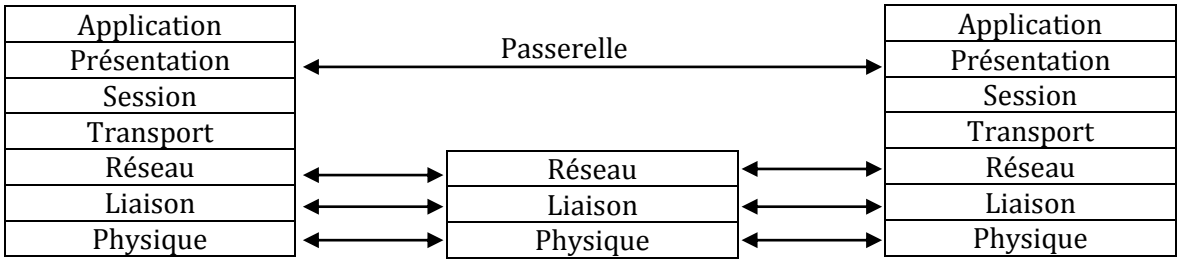
4.3. **le multiplexage statistique** améliore le multiplexage temporel en n'attribuant la voie haute vitesse qu'aux voies basse vitesse qui ont effectivement quelque chose à transmettre. En ne transmettant pas les silences des voies basses, cette technique implantée dans des concentrateurs améliore grandement le débit global des transmissions mais elle fait appel à des protocoles de plus haut niveau et est basée sur des moyennes statistiques des débits de chaque ligne basse vitesse.

LE MODELE DE REFERENCE

INTRODUCTION

Le transport des données d'une extrémité à une autre d'un réseau nécessite un support physique ou hertzien de communication . Pour que ces données arrivent correctement au destinataire avec qualité de service (QoS: Quality of Service) exigée, il faut alors une architecture logicielle.

Le modèle d'architecture proposé par l'ISO (International Standard Organisation) pour l'interconnexion des systèmes ouverts dit MODELE DE REFERENCE appelée également OSI (Open system Interconnexion),est constitué de couches de protocoles.



PRINCIPES DE LA STRUCTURATION EN COUCHES

Le modèle OSI est composé de sept couches.
Chaque couche peut interagir uniquement avec les deux couches adjacentes.

Une couche N est constituée d'un ensemble d'entités formant un sous-système de niveau N. Elle ne peut dialoguer qu'avec une couche de même niveau N sur une autre machine. Les communications se font donc entre entités homologues.

INTERACTIONS ENTRE COUCHES

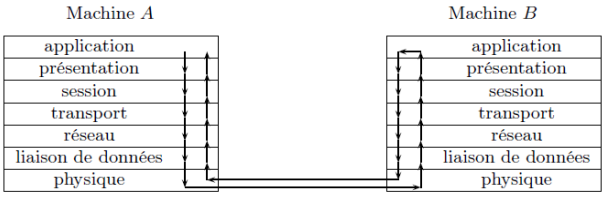
1. **Protocoles et Services**

Les notions de protocole et de service sont fondamentales.

Un protocole est un ensemble de règles et formats, syntaxiques et sémantiques prédéfinis pour les entités d'un même niveau N de deux machines différentes.

Un service est fourni par une couche de niveau N à la couche de niveau N + 1 d'une même machine. La couche de niveau N + 1 communique à la couche N les caractéristiques du service attendu.

Les services fournis par une couche N sont identifiés par des SAP (Service Access Point) ou ports.
La figure suivante décrit la communication entre les 7 niveaux de couches de deux entités communicantes A et B.



2. Encapsulation, PDU et SDU

Les messages échangés par un protocole de niveau N sont appelés des **PDU_N** (Protocol Data Unit de niveau N).

Les messages échangés entre la couche N et la couche inférieure N−1 qui sont les **PDU_N** deviennent des **SDU_{N-1}**(Service Data Unit de niveau N − 1).

De plus, un protocole de niveau N ajoute au **SDU_N** qu'il a reçu des informations de contrôle visant à contrôler la bonne exécution du protocole. Ces informations de contrôle sont appelées **PCI_N** (Protocol Control Information de niveau N).

On a par conséquent :

$$PDU_N = SDU_N + PCI_N$$
$$SDU_N = PDU_{N+1}$$

On dit alors que le PDU_N encapsule le SDU_N.

Au lieu d'indexer le PDU ou le SDU par le numéro de la couche, on le fait souvent précéder de la première lettre du nom de la couche (en anglais). Par exemple, NPDU = PDU_3, où le N indique la couche réseau (network).

3. Primitives de service

Il existe 4 primitives de service : requête, indication, réponse et confirmation.

Une **requête** est initialement envoyée par la couche N à la couche N−1 d'une même entité. Ensuite, une **indication** est transmise de la couche N−1 à la couche N de l'autre entité communicant.

La **réponse** est envoyée par la couche N à la couche N−1 de cette seconde entité. Enfin, une **confirmation** est transmise de la couche N−1 à la couche N de l'entité ayant émis la requête.

Ceci est illustré dans la figure suivante.

LES COUCHES DU MODELE DE REFERENCE

COUCHE I: LE NIVEAU PHYSIQUE (couche de physique)

Le niveau physique fournie les moyens mécaniques, électriques fonctionnels et procéduraux nécessaires à l'activation, au maintient et à la désactivation des connexions physiques destinées à la transmission des éléments binaires entre entités de liaisons.

Dans cette couche, on trouve tous le matériel et les logiciels nécessaires au transport des éléments binaires, notamment:

- les interfaces de connexion des équipements informatiques (jonctions)
- les modems (modulateur / démodulateur)
- Les nœuds de transfert
- divers équipements spécifiques du réseau pour assurer la continuité du chemin physique ex: satellites.

COUCHE II: LE NIVEAU TRAME (couche liaison)

Maintenant que les machines sont reliées entre elles physiquement, il reste à voir comment ces machines s'identifient pour échanger des informations sur le réseau. En général, chaque machine se voit attribuer une adresse physique, unique sur le réseau, qui permet de l'identifier.

Le niveau trame fournit les fonctions nécessaires pour transporter un bloc d'informations appelé trame d'un nœud de transfert vers un autre nœud de transfert. La fonction de base consiste à reconnaitre le début et la fin du bloc de sorte qu'il puisse être transmis sur le support physique et capté correctement par le récepteur.

COUCHE III : LE NIVEAU PAQUET (couche réseau)

Le rôle du niveau paquet est de transporter les paquets (flots) d'un utilisateur jusqu'à un récepteur connecté au même réseau via des nœuds de transfert intermédiaires. En d'autres termes, le niveau paquet, que l'on appelle la couche réseau permet d'acheminer correctement des paquets d'information jusqu'au récepteur connecté au réseau, en transitant par des nœuds de transfert intermédiaires. Si l'émetteur et le récepteur ne sont pas situés sur le même réseau, un premier niveau paquet transporte les données d'un émetteur vers une passerelle. Un autre niveau paquet qui peut être le même que le premier, achemine les paquets sur le deuxième réseau traversé et ainsi de suite jusqu'à arriver au récepteur.

Le paquet ne donne pas la possibilité de reconnaitre son début et sa fin, pour cela il faut l'encapsuler dans une trame.

Pour mettre en place et développer les fonctionnalités de la couche réseau, il est possible de choisir entre les deux grandes méthodes d'accès :

• **Le mode avec connexion,** dans lequel l'émetteur et le récepteur se mettent d'accord sur un comportement commun et négocient les paramètres et les valeurs à mettre en œuvre.

• **Le mode sans connexion**, qui n'impose pas de contrainte à l'émetteur par rapport au récepteur.

COUCHE IV : LE NIVEAU MESSAGE (couche transport)

Le niveau message assure le transport des messages d'un client vers un client de destination.

La fonction de base (logiciel simple) du niveau transport s'appelle fragmentation (segmentation)/ réassemblage. c'est une opération qui consiste à fragmenter les messages en paquets puis à les réassembler à la sortie pour retrouver le message de départ.

COUCHE V : LE NIVEAU SESSION (couche session)

Le niveau session fournit les moyens nécessaires à l'organisation et à la synchronisation du dialogue entre les clients en communication .

Session: mise en communication de deux ou plusieurs extrémités de façon à gérer leur dialogue.

Point de synchronisation: état de la communication sur lequel l'émetteur et le récepteur se mettent d'accord pour redémarrer en cas de problème.

Ce niveau a pour but d'ouvrir et de fermer des sessions entre utilisateurs.

COUCHE VI : LE NIVEAU PRESENTATION (couche présentation)

Le niveau présentation prend en charge la syntaxe des informations que les entités d'application se communiquent: la couche 6 met en forme les données pour les rendre compréhensibles par le destinataire.

Deux aspects sont définis dans cette couche: la représentation des données transférées entre entités d'application et la structure des données à laquelle des entités se réfèrent au cours de leur communication.

En résumé, le niveau présentation s'intéresse à la syntaxe tandis que le niveau application se charge de la sémantique.

COUCHE VII : LE NIVEAU APPLICATION (couche application)

Le niveau application fournit aux processus d'application le moyen de s'échanger des informations par le biais du réseau sous-jacent. Par exemple, un utilisateur peut envoyer un message électronique à son correspondant en utilisant les couches de protocole donnant accès au réseau.

Il s'intéresse particulièrement à la sémantique.

Le niveau application est structuré principalement par les catégories d'application suivantes: Messagerie électronique , les services d'annuaire qui répertorient les équipements adressables et donnent les adresses des destinataires, requêtes sur BD réparties, utilisation de Terminal Virtuel,...

TRANSMISSION DE DONNEES

La couche physique fournit les moyens mécaniques, électriques, fonctionnels et procéduraux nécessaires à l'activation, au maintien et à la désactivation des connexions physiques destinées à la transmission de bits entre deux entités de liaison de données.

La transmission de plusieurs bits peut s'effectuer en série ou en parallèle.

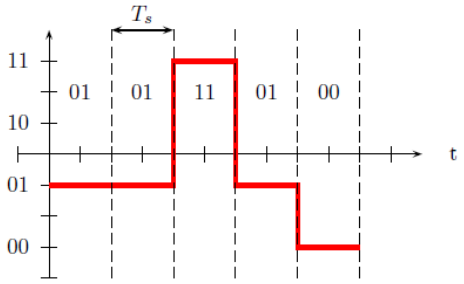
En série, les bits sont envoyés les uns derrière les autres de manière synchrone ou asynchrone. Dans le mode **synchrone** l'émetteur et le récepteur se mettent d'accord sur une base de temps (un top d'horloge) qui se répète régulièrement durant tout l'échange. À chaque top d'horloge (ou k tops d'horloge, k entier fixé définitivement) un bit est envoyé et le récepteur saura ainsi quand lui arrive les bits. Dans le mode **asynchrone**, il n'y a pas de négociation préalable mais chaque caractère envoyé est précédé d'un bit de start et immédiatement suivi d'un bit de stop. Ces deux bits spéciaux servent à caler l'horloge du récepteur pour qu'il échantillonne le signal qu'il reçoit afin d'y décoder les bits qu'il transmet.

En parallèle, les bits d'un même caractère sont envoyés en même temps chacun sur un fil distinct, mais cela pose des problèmes de synchronisation et n'est utilisé que sur de courtes distances (bus par exemple).

Quel que soit le mode de transmission retenu, l'émission est toujours cadencée par une **horloge** dont la vitesse donne le débit de la ligne **en bauds**, c'est-à-dire le nombre de tops d'horloge en une seconde.

Ex: une ligne d'un débit de 100 bauds autorise 100 émissions par seconde. Si à chaque top d'horloge, un signal représentant 0 ou 1 est émis, alors dans ce cas le débit en bit/s est équivalent au débit en baud.

Cependant, on peut imaginer que le signal émis puisse prendre 4 valeurs distinctes (0, 1, 2, 3)(00,01,10,11) dans ce cas le signal a une **valence** de 4 et le débit en bit/s est le double de celui en baud.

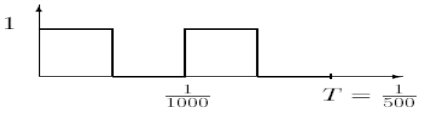


La **valence** est le nombre de niveaux utilisés pour coder la suite binaire.

Un codage sur V niveaux implique un regroupement des bits. Par exemple, pour réaliser un codage sur 4 niveaux (valence 4), les bits doivent être regroupés deux par deux ; dans ce cas un groupe de 2 bits est appelé un symbole.

1. Transmission en bande de base

La transmission en bande de base consiste à envoyer directement les suites de bits sur le support à l'aide de signaux carrés constitués par un courant électrique pouvant prendre 2 valeurs (5 Volts ou 0 par exemple). L'émetteur envoie sur la ligne un signal carré du type de celui de la figure suivante pour la séquence de bits 1010 par exemple.



En considérant ce signal y(t) comme périodique, on peut le décomposer en une série de Fourier de la forme:

$$y(t) = a.\sin(2\pi ft + \varphi)$$

où

- a : amplitude
- f : fréquence (= 1/T)
- φ : phase

Cependant, le câble sur lequel est émis le signal possède une bande passante qui est l'intervalle des fréquences possibles sur ce support, donc à la réception on ne retrouve pas toute la richesse du signal initial et dans la plupart des cas le signal carré sera très déformé.

Débit d'un signal numérique D : représente la quantité d'information émise par unité de temps par une source, exprimé en bit/s. Il dépond des caractéristiques du support de transmission et des techniques de transmission: **D = 1/T**

Capacité du canal: un canal de transmission n'est jamais parfait, le signal réellement transmis est la somme du signal à transmettre et d'un bruit additif. Le bruit additif d'un dispositif électronique est caractérisé par le rapport signal/ bruit (S/N):

$$\mathbf{S/N_{db}=10log_{10}(S/N_{wt})}$$

S: puissance du signal
N: puissance du bruit

On définit la capacité d'un canal comme le débit binaire théorique maximum que ce canal peut supporter.

La capacité C d'une ligne de transmission peut être définie à l'aide de la formule de Shannon par:

$$\mathbf{C= B\ log_2(1+S/N)}$$

B: la largeur de bande passante
Remarque : $\log_2(x)= \log_{10}(x)/\log_{10}(2)$, $\log_{10}(2)=0,3$

La rapidité de modulation : le débit de symboles est appelé rapidité de modulation et est noté R. R s'exprime en bauds (du nom de Baudot, inventeur du télex).
On a donc

$$\mathbf{R = 1/Ts =1/(log_2\ (V) \times Tb) = D/log_2\ (V)}$$

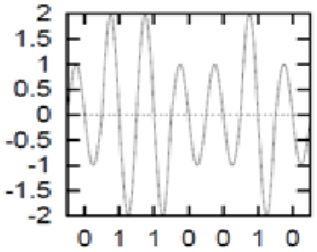
On retiendra que : $\mathbf{D = R \times log_2\ V\ .}$

Ts : temps symbole
Tb: temps d'un bit

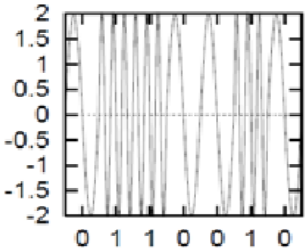
2. Transmission modulée

Sur les longues distances on émet un signal sinusoïdal qui sera facilement décodable par le récepteur. Ce signal sinusoïdal est obtenu grâce à un modem (modulateur-démodulateur)qui est un équipement électronique capable de prendre en entrée un signal en bande de base pour en faire un signal sinusoïdal (modulation) et l'inverse à savoir restituer un signal carré à partir d'un signal sinusoïdal (démodulation). Autrement dit, il permet de passer des signaux numériques discrets (0 ou 1) à des signaux analogiques continus.

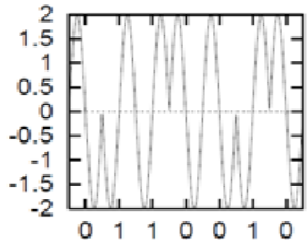
Il existe trois types de modulation décrits dans la figure suivante:



- **la modulation d'amplitude:** envoie un signal d'amplitude différente suivant qu'il faut transmettre un 0 ou un 1. Cette technique est efficace si la bande passante et la fréquence sont bien ajustées. Par contre, il existe des possibilités de perturbation (orage, lignes électriques...), car si un signal de grande amplitude (représentant un 1) est momentanément affaibli le récepteur l'interprétera à tort en un 0.



- **la modulation de fréquence:** envoie un signal de fréquence plus élevée pour transmettre un 1. Comme l'amplitude importe peu, c'est un signal très résistant aux perturbations (la radio FM est de meilleure qualité que la radio AM) et c'est assez facile à détecter.



- **la modulation de phase** change la phase du signal (ici de 180°) suivant qu'il s'agit d'un 0 (phase montante) ou d'un 1 (phase descendante).

Dans les exemples donnés ci-dessus on a seulement 2 niveaux possibles à chaque fois, donc on a uniquement la possibilité de coder 2 valeurs différentes à chaque instant, dans ce cas 1baud=1bit/s.

De manière plus sophistiquée il existe des modems capables de moduler un signal suivant plusieurs niveaux, par exemple 4 fréquences différentes que le modem récepteur saura lui aussi distinguer. Dans ce cas, chaque signal envoyé code 2 bits donc 1 baud = 2bit/s.

3. La numérisation: l'intérêt de la numérisation est le traitement des données non informatiques telles que voix, images et sons.

Elle consiste à convertir un signal analogique en un signal numérique, il faut échantillonner le signal analogique avec une fréquence f = 1/T où T représente un intervalle de temps.
Elle se fait en trois étapes:

a) **Echantillonnage:** consiste à prélever les valeurs du signal continu (échantillon) à des instants régulièrement espacés. En d'autres termes: transformer une fonction continue en une fonction discrète.

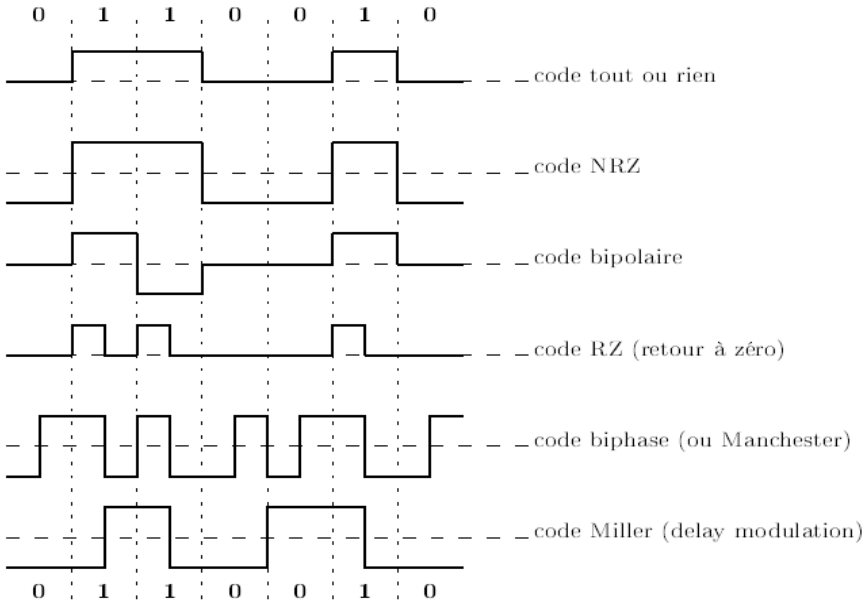
Brièvement, c'est découper l'espace temporel sur des instants réguliers.

b) **La quantification :**
Détermine une valeur pour chaque échantillon sur une échelle numérique. La loi la plus simple consiste à diviser l'ordonnée en segments égaux. Le nombre de segments dépond du nombre de bits choisis pour la numérisation. Par exemple, un codage sur 8bits engendre 2^8 segments. La bande passante est divisée en 256 segments . Le chois de la valeur de l'échantillon s'effectue simplement en sélectionnant la valeur la plus proche.

c) **Le codage**: consiste à affecter une valeur numérique aux échantillons obtenus lors de la première phase. Ces valeurs sont ensuite transportées dans le signal numérique.

4. Codage de l'information

Dans la figure suivante nous trouvons quelques exemple de codage de l'information pour une transmission en bande de base.



- **le code tout ou rien** : c'est le plus simple, un courant nul code le 0 et un courant positif indique le 1
- **le code NRZ (non retour à zéro):** pour éviter la difficulté à obtenir un courant nul, on code le 1 par un courant positif et le 0 par un courant négatif.
- **le code bipolaire** : c'est aussi un code tout ou rien dans lequel le 0 est représenté par un courant nul, mais ici le 1 est représenté par un courant alternativement positif ou négatif pour éviter de maintenir des courants continus.
- **le code RZ** : le 0 est codé par un courant nul et le 1 par un courant positif qui est annulé au milieu de l'intervalle de temps prévu pour la transmission d'un bit.
- **le code Manchester :** ici aussi le signal change au milieu de l'intervalle de temps associé à chaque bit. Pour coder un 0 le courant sera négatif sur la première moitié de l'intervalle et positif sur la deuxième moitié, pour coder un 1, c'est l'inverse. Autrement dit, au milieu de l'intervalle il y a une transition de bas en haut pour un 0 et de haut en bas pour un 1.
- **le code Miller** : on diminue le nombre de transitions en effectuant une transition (de haut en bas ou l'inverse) au milieu de l'intervalle pour coder un 1 et en n'effectuant pas de transition pour un 0 suivi d'un 1. Une transition est effectuée en fin d'intervalle pour un 0 suivi d'un autre 0.

DETECTION ET CORRECTION DES ERREURS

Le support matériel utilisé par la couche physique n'est pas fiable à 100%. Il est par conséquent nécessaire de pouvoir détecter des erreurs parmi la suite de bits reçue, et éventuellement les corriger. Pour cela, la couche liaison de données de l'émetteur ajoute des bits au message à transmettre, qui permettent à la couche liaison de données de l'entité réceptrice du message de vérifier la cohérence de ce qu'elle a reçu.

La couche liaison de données construit ainsi des LPDU, encore appelées trames, qui comportent en particulier un FCS (Frame Check Sequence).

La problématique des erreurs comporte 3 aspects :

– la détection d'une erreur ;
– la localisation de l'erreur détectée ;
– la correction de l'erreur trouvée.

Pour répondre à ces problèmes, on utilise des codes qui sont appliqués au message à transmettre. Ils permettent de détecter certaines erreurs, mais pas nécessairement toutes, et peu permettent la correction. Ces techniques ne sont donc pas complètement fiables, d'autant que le FCS, utilisé pour vérifier et corriger le message, peut lui aussi être erroné.

1. Un code simple : la répétition

Une approche naïve consiste à dupliquer (c'est-à-dire répéter) le message à transmettre. Supposons que le message effectivement transmis soit le double du message réel. Par exemple, pour envoyer 11100010, on transmet 1110001011100010. La détection et la localisation des erreurs sont alors simples : on cherche les différences entre la première et la seconde moitiés du message. Par contre, il est impossible de corriger une erreur détectée : le bit erroné est différent dans les deux copies, et rien ne permet de dire lequel est le bon.

Pour remédier à ce problème, on peut envoyer le message en 3 exemplaires au lieu de 2. Dans ce cas, un bit a soit la même valeur dans toutes les copies, ou la même valeur dans deux d'entre elles et l'autre valeur dans la troisième copie. Le bit correct est probablement celui qui apparaît en deux exemplaires : on peut cette fois corriger l'erreur.

2. Codes à contrôle de parité

Les codes à contrôle de parité sont de parité soit paire, soit impaire. Dans le premier cas, on va protéger une séquence de bits en ajoutant un nouveau bit de telle sorte que le nombre de bits ayant la valeur 1 (dans la séquence protégée plus le bit introduit) soit pair. Dans le second cas, ce nombre doit être impair.

2.1. VRC (Vertical Redundancy Check)

C'est la technique la plus simple. Un code ASCII étant défini sur 7 bits, on utilise le 8ème bit de l'octet pour introduire le code vérificateur.

Exemple : Pour transmettre la chaîne de caractères IUT, on code chaque lettre en ASCII, puis on ajoute le code de parité.

Lettre	ASCII	VRC pair	VRC impair
I	1001001	11001001	01001001
U	1010101	01010101	11010101
T	1010100	11010100	01010100

Pour envoyer le message avec un code de parité pair, on transmet (avec l'ordre d'envoi des bits de gauche à droite) : 11001001 01010101 11010100

Ce code permet de détecter les erreurs en nombre impair sans pouvoir corriger. Il est peu efficace.

2.2. LRC (Longitudinal Redundancy Check)

Le principe est similaire à celui du VRC, mais au lieu de protéger les caractères un par un, on protège l'ensemble des bits de même rang de tous les caractères. On obtient alors un code de protection sur 7 bits.

Exemple : Pour protéger IUT, on calcule le code :

I	1001001
U	1010101
T	1010100
LRC pair	1001000
LRC impair	0110111

Pour envoyer le message avec un code de parité pair, on transmet :
1001001 1010101 1010100 1001000

2.3. LRC et VRC

On peut également combiner les deux techniques précédentes. On protège alors chaque caractère par un code VRC et l'ensemble des bits par un code LRC. On obtient donc un LRC sur 8bits. La parité des LRC et VRC utilisés est la même (tous les deux pairs ou tous les deux impairs).

Exemple : Pour transmettre la chaîne de caractères IUT, on code chaque lettre en VRC puis en LRC :

Lettre	ASCII	VRC pair	VRC impair
I	1001001	11001001	01001001
U	1010101	01010101	11010101
T	1010100	11010100	01010100
LRC		01001000	00110111

Pour envoyer le message avec un code de parité pair, on transmet :
11001001 01010101 11010100 01001000

3. Codes polynomiaux

les codes VRC et LRC s'appliquent sur des blocs de données de tailles équivalentes, si les blocs de données sont de tailles variables, ces codes deviennent inutiles.
Les codes polynomiaux possèdent un avantage : ils peuvent opérer sur des blocs de taille variable. C'est un cas qui se présente souvent en transmission, puisque la taille des données peut varier.

Un code polynômial est basé sur l'utilisation d'un polynôme générateur G(x). Les polynômes manipulés sont binaires : tous les coefficients sont 0 ou 1. Par conséquent, un polynôme générateur de degré k s'écrit sous la forme :

$$G(x) = a_0 + a_1x + a_2x^2+...+a_kx^k$$

Le polynôme G(x) est associé à une valeur binaire.

Exemple : La valeur binaire associée au polynôme $G(x) = x^3 + x + 1$ est 1011.

Soit M le message (séquence de bits) à protéger. Un polynôme M(x) lui est associé :

$$M = m_n \ldots m_2m_1m_0$$
$$M(x) = m_nx^n + \cdots + m_2x^2 + m_1x + m_0$$

Exemple : Au message M = 1101 est associé le polynôme $M(x) = x^3 + x^2 + 1$.

Codage

Le calcul du CRC (Cyclic Redundancy Code)s'effectue dans le corps Z/2Z, c'est-à-dire que :

- $1 + 1 = 0$
- $x + x = 0$
- $x = -x$

Soient :
– G(x) un polynôme générateur de degré k ;
– M(x) le polynôme associé au message M à transmettre.

La procédure de codage consiste à :

- calculer: $P(x) = M(x)*x^k$.

  Ceci correspond à un décalage de k bits (vers la gauche) du message M. La longueur du CRC calculé sera aussi de k bits. Cette opération de décalage revient à préparer la place nécessaire pour ces *k* bits de CRC.

- diviser le polynôme P(x) par G(x). Soient Q(x) et R(x) les polynômes quotient et reste ainsi obtenus :

  $$P(x) = Q(x)*G(x) + R(x)$$

- le CRC est le reste R(x) calculé. On remarque que le reste est forcément au maximum de degré $k-1$.

- le message effectivement transmis est associé au polynôme:

$$M'(x) = P(x) + R(x).$$

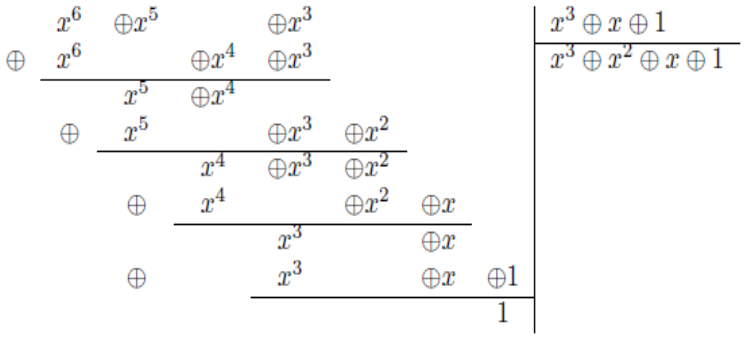
Il est par conséquent composé du message initial M suivi de la séquence de *k* bits correspondant à R(x).

Exemple : Soient le polynôme générateur $G(x) = x^3 + x + 1$ et le message à envoyer M = 1101.

Le polynôme correspondant au message est $M(x) = x^3 + x^2 + 1$. Le degré de G(x) est 3. Donc,

$P(x) = M(x) * x^3 = x^6 + x^5 + x^3$.

Effectuons la division de P(x) par G(x) :



Le quotient est donc $Q(x) = x^3 + x^2 + x + 1$,
et le reste $R(x) = 1$.

Le message transmis a alors pour polynôme $M'(x) = x^6 + x^5 + x^3 + 1$,

d'où M' = 1101001.

PROTOCOLE DE NIVEAU LIAISON DE DONNES

(HDLC)

1. Définition

La couche liaison de données est définie comme étant l'ensemble des équipements et des logiciels fournissant les moyens fonctionnels nécessaires pour acheminer des données avec un taux d'erreurs garanti.

Objectif: fiabiliser la transmission physique et offrir un service à la couche réseau pour acheminer les bits remis par le processus réseau vers leur destination.

Unité d'information: blocs de bits appelés Trame ou L-PDU.

2. Fonctions d'un protocole de liaison de données:

Un protocole de liaison de données a pour tâches de :

- préciser la structure syntaxique (format) des trames valides.
- La place et la signification des différents champs dans une trame.
- Le critère début et fin de trame.
- La technique de détection des erreurs à utiliser.
- L'algorithme de contrôle de flux

3. Le protocole HDLC:

3.1 . Définition:

Le HDLC (High-Level Data Link Control) est un protocole de niveau 2 (couche liaison) du modèle OSI. Son but est de définir un mécanisme pour délimiter des trames de différents types, en ajoutant un contrôle d'erreur. Il est défini par l'organisation internationale de normalisation sous la spécification ISO3309. Il offre un service de transfert de données fiable et efficace entre deux systèmes adjacents.

3.2. Caractéristiques:

- Transmission synchrone
- Liaison point-à-point ou multipoint
- Full duplex
- Contrôle de flux par mécanisme d'anticipation
- Fenêtre d'anticipation: possibilité d'envoi de plusieurs messages (trames) en séquence sans attendre d'acquittement
- Piggybacking (superposition) une trame peut contenir des données et des informations de service (ex: ACK)
- Protection des trames par un FCS(Frame Check Sequence)
- Fanion (Flag): délimiteur de début et de fin de trame
- Transparence vis-à-vis des données transportées

3.3. Modes de Liaison: Liaison HDLC équilibré (LAP-B: Balanced Link Access Procedure)

Deux (2) stations mixtes:

- à la fois primaires et secondaires
- émettre des commandes ou des réponses
- responsabilités égales

Chaque station mixte peut prendre différents états:

- ✓ **Mode asynchrone équilibré ABM** (Asynchronous Balanced Mode) ou **BAC** (Balanced Asynchronous Class) suite aux commandes suivantes
  - **SABM** Set Asynchronou Balanced Mode
  - **SABME** Set Asynchronous Balanced Mode Extended
- ✓ **Mode déconnecté** suite à la commande:
  - **DISC** : Disconnect
- ✓ **Mode d'initialisation** suite à la commande:
  - **SIM** : Set Initialization Mode

3.4. Définition d'une trame: c'est l'unité de données du protocole de niveau Liaison de données (L-PDU) . Elle est composée de :

- une suite de bits (d'une longueur variable mais bornée)
- Le début et la fin de la trame sont souvent identifiés par des délimiteurs

On distingue souvent Trois (03) ensembles de champs: l'en-tête (header), le champ de données et la terminaison (trailer).

3.4.1. Format général d'une trame HDLC

Fanion	Adresse	Commande	Information	FCS	Fanion
01111110				$x^{16}+x^{15}+x^2+1$	01111110
8bits	8bits (ou +)	8 bits (ou16)	>= 0	16 bits	8 bits

A. Le Fanion (Flag)

Délimite la trame: dedans/dehors, la trame est de longueur variable puisque le champ de données est de longueur variable.

Sa valeur est fixe: 01111110 (binaire) 7E (hexadécimal)

Unicité du fanion:

Etant donnée que la champ de données de la trame peut comporter n'importe quel octet (le transport des données est transparent), il doit y avoir ne assurance de l'unicité de la configuration binaire du fanion à l'intérieur de la trame par:

- **Transcodage:** la trame (sauf les fanions) est transcodée lors de la transmission; toute suite de 5 bits consécutif à 1 est transcodée en une suite de 5bits à 1 et d'un bit à 0.

- **Rencodage :** opération inverse au récepteur.

B. Adresse: Identifier le sens des émissions des trames de commandes et de réponses ainsi que leurs émetteurs.

C. Commande

Par le champ commande, on distingue Trois (03) types de trames

- ✓ Les trames d'information (**I** Information)
- ✓ Les trames de supervision (**S** Supervisory)
- ✓ Les trames non numérotées (**U** Unnumbered)

Type de trame	Champ commande								
I	0	N(S)			P/F	N(R)			
S	1	0			P/F	N(R)			
U	1	1	M	M	P/F	M	M	M	

Deux formats du champ de commande existent:

- Le format normal (8bits)
- Le format étendu pour les trames numérotées (16 bits) négocié lors de l'établissement de connexion SABME pour avoir un champs de commande plus grand et ainsi la numérotation modulo 128.

C.1. Les trames d'information:

- ✓ Transportent les données utilisateurs
- ✓ Acquittement- retransmission (piggybacking)
- **N(S) :** numéro de la trame d'information courante (modulo 8 ou 128)
- **N(R) :** numéro de la prochaine trame d'information attendue (modulo 8 ou 128):
  - Acquitte toute les trames de numéros strictement inférieurs à N(R)
  - La perte d'un acquittement peut ainsi être compensée par le prochain acquittement.
- **Le bit P/F** signifie Poll/Final (invitation à émettre / fin)il est positionné s'il a la valeur 1. Par convention, le bit positionné vaut P si la trame est une commande et F si la trame est une réponse. L'émission d'une commande avec P=1 exige une réponse immédiate avec F=1.

A la réception d'une trame avec le bit P/F positionné, le bit vaut F si on attend une réponse à une commande déjà envoyée et il vaut P si aucune commande n'a été envoyée.

C.2. Les trames de supervision:

- ✓ codées dans le sous-champ Type du champ de commande
- ✓ Commande ou réponse

Il y a 4 types de trames de supervision:

RR(Received and Ready) 00 : Acquittement

- confirme la réception des trames de données de n°< N(R)
- demande la transmission des trames suivantes

RNR (Received and Not Ready) 10 : Contrôle de flux

- confirme la réception des trames de données de n°< N(R)
- interdit la transmission des trames suivantes.

REJ (Reject) 01 : Protection contre les erreurs

- confirme la réception des trames de données n°< N(R)
- demande la retransmission des trames de n° >= N(R)

SREJ (Selective Reject) 11 : Protection contre les erreurs

- confirme la réception des trames de données n°< N(R)
- demande la retransmission des trames de n° = N(R)

C.3. Les trames non-numérotées U:

Ces trames transportent des commandes ou des réponses de la gestion de la liaison (établissement, rupture, choix d'un mode de réponse....).

Commandes:

- **SABM:** Set Asynchronous Balanced Mode : 1111P/F100: demande de connexion
- **SABME :** identique à SABM mais en mode étendu
- **DISC :** Disconnect : 1111P/F010: Libération de connexion

Réponses:

- **UA :** Unnumbred Acknowledgement : 1100P/F110 : acquittement de trame non-numérotée
- **FRMR :F**RaMe Reject 1110P/F001: Rejet de trame
- **DM :** Disconnect Mode **:** 1111P/F000 : le terminal est déconnecté

D. FCS (Frame Check Sequence): Séquence de détection d'erreurs

FCS est le résultat d'une opération mathématique de type polynomial effectuée sur toute l'étendue de la trame sauf délimiteurs par la machine émettrice de cette trame.

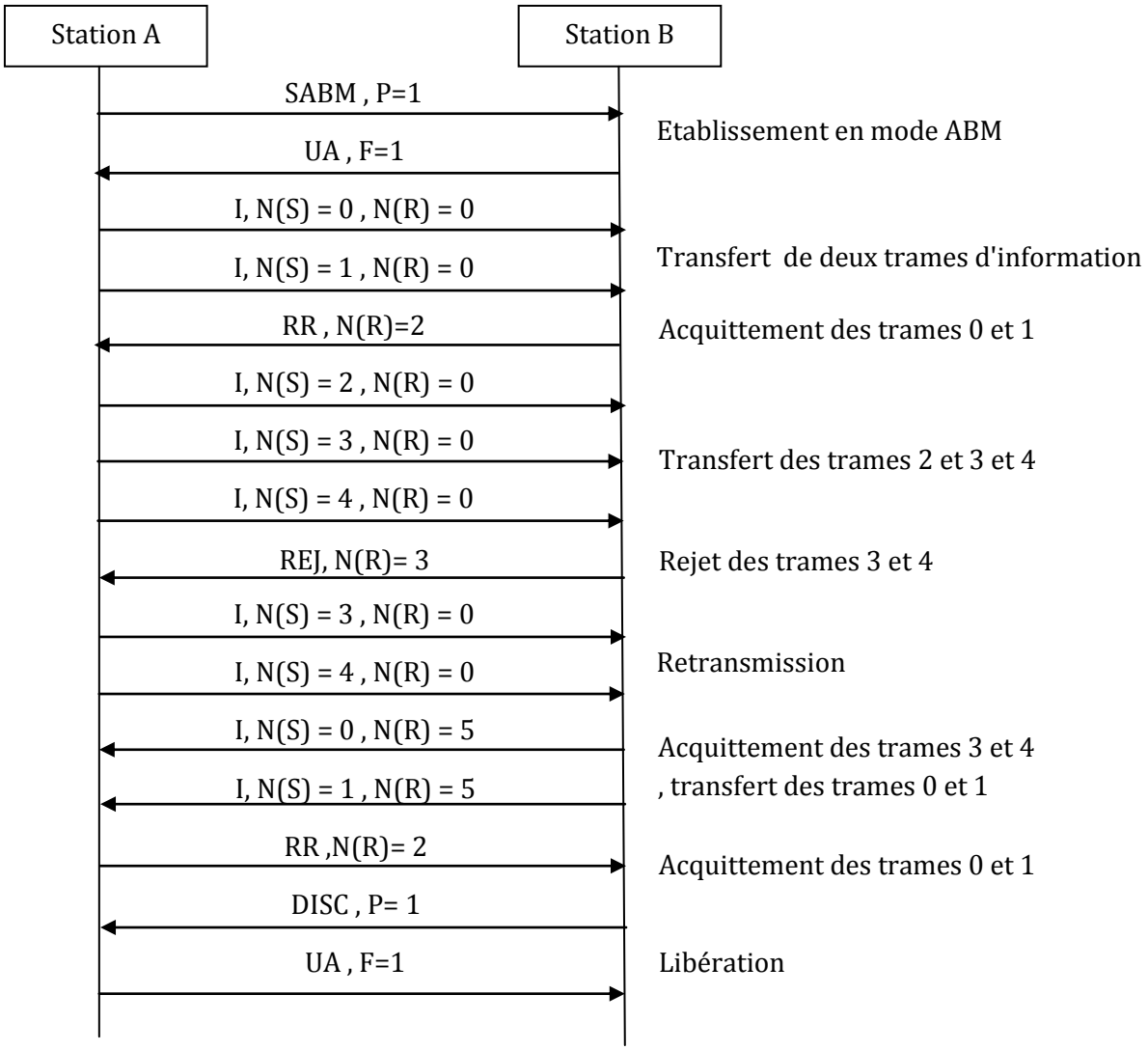
Le polynôme générateur du FCS retenu est $x^{16}+x^{15}+x^2+1$

3.5. Description des échanges :

- Etablissement de la liaison par émission des trames non-numérotées SABM et UA . Le bit P/F est positionné à 1 dans la trame de commande SABM invitant la station B à répondre, celle ci émet tout de suite un acquittement UA avec le bit F à 1.

- Transmission bidirectionnelle de trames I avec gestion des numéros de séquence N(R) et N(S). L'acquittement des trames I émises jusqu'au numéro N(S)= x est réalisé par renvoi dans une trame RR ou I de N(R) = x+1, x+1 est le numéro de la trame attendue (exemple: N(R) = 4 acquitte les trames N(S) <4) La trame REJ avec N(R) = x signifie le rejet des trames N(S) >= x . La gestion des numéros N(S) et N(R) est symétrique pour les deux stations , le transfert des trames I peut donc être réalisé en duplex intégral.

-Libération de la liaison par émission des trames non numérotées DISC et UA.



Exemple d'échange HDLC

INTERCONNEXION DE RESEAUX

(Le Niveau Réseau)

Dans un réseau, les paquets doivent être transportés d'une extrémité à une autre. Le niveau paquet (couche 3 du modèle OSI)a la responsabilité de cet acheminement. Les paquets proviennent de la fragmentation des messages que les utilisateurs souhaitent échanger, ils doivent être encapsulé dans des trames pour être transportés sur une ligne physique.

Au troisième niveau de l'architecture internet, se trouve l'implantation du protocole IP (Internet Protocol). Ce protocole en mode datagramme, va offrir les fonctions de routage, l'interconnexion des machines et gère la fragmentation des données.

1. La notion d'adressage:

Une adresse est une suite de caractères désignant sans ambiguïté un point physique de raccordement à un réseau ou identifiant un processus ou une machine.

on désigne par technique d'adressage l'ensemble des moyens utilisés pour identifier les correspondants. Pour assurer la communication, le système d'extrémité source doit fournir au réseau l'adresse de l'extrémité destination (adresse destinataire) et celui-ci doit pouvoir identifier son correspondant (adresse source).

On peut trouver différents types d'adressage:

- ✓ **Adressage à plat ou global**: l'adresse correspond à un numéro unique attribué sans aucune règle de structuration(adresse MAC dans les réseaux locaux : deux champs: le premier sur 3 octets désigne le constructeur de la machine et le deuxième sur 3 octets correspond à un numéro de séquence attribué par le constructeur à une machine unique).
- ✓ **Adressage hiérarchique (Logique, adressage IP)**: utilisé dans les grands réseaux d'interconnexion, identifie un point d'accès au réseau. Il désigne le réseau et les points d'accès participant à l'acheminement des informations.

2. Le protocole IP:

C'est un protocole de niveau réseau, responsable de :

- la transmission des données en mode non connecté;
- l'adressage et le routage des paquets entre stations via des routeurs;
- la fragmentation des données;
  Lors de l'émission; les fonctionnalités assurées sont:
  - identification du paquet;
  - détermination de la route à suivre;
  - vérification du type d'adressage (station ou diffusion);
  - fragmentation de la trame si nécessaire.
    à la réception, les fonctionnalités sont:
    - vérification de la longueur du paquet;
    - contrôle des erreurs;
    - réassemblage en cas de fragmentation

- transmission du paquet réassemblé au niveau supérieur.

2.1. Le format du paquet IP

Le paquet IP ou datagramme IP est organisé en champs de 32bits :

<table>
<tr><td colspan="8">31 23 15 7 0</td></tr>
<tr><td colspan="2">Version</td><td colspan="2">Longueur</td><td colspan="2">Type de service</td><td colspan="2">Longueur totale</td></tr>
<tr><td colspan="6">Identificateur</td><td>Drapeaux</td><td>Position du fragment</td></tr>
<tr><td colspan="4">Durée de vie</td><td colspan="2">Protocole</td><td colspan="2">Checksum de l'en-tête</td></tr>
<tr><td colspan="8">Adresse station source</td></tr>
<tr><td colspan="8">Adresse station destinatrice</td></tr>
<tr><td colspan="7">Options éventuelles</td><td>Bourrage éventuel</td></tr>
<tr><td colspan="8">Données couche 4</td></tr>
</table>

- Version: numéro de la version du protocole IP (4 ou 6)(4bits)
- Longueur: longueur de l'en-tête codée sur 4bits
- Type de service (TOS) désigne la qualité de service qui doit être utilisée par le routeur. (ex: privilégier le débit par rapport au délai de transmission)(8bits)
- Longueur totale: longueur totale du fragment (en-tête +données) exprimé en nombre d'octets;(16bits)
- Identificateur : identifie le paquet pour la fragmentation (tous les fragments d'un même paquet portent le même numéro)(16 bits)
- Drapeaux: gère la fragmentation sur trois bits suivant le format:
  - DF MF;
  - le bit DF(Don't Fragment) demande au routeur de ne pas fragmenter le paquet quand il est positionné à 1;
  - le bit MF (More Fragment)est positionné à 1 dans tous les fragments , sauf le dernier(MF= 0 : soit c'est le dernier fragment soit c'est le fragment unique" le paquet n'a pas été refragmenté par le routeur) .
- Position du fragment: (Fragment Offset): indique par multiple de 8octets la position du fragment dans le paquet courant. Tous les fragments du paquet, sauf le dernier, doivent avoir pour longueur des multiples de 8octets. Avec codage sur 13 bits, le maximum pour un paquet est de 8192 fragments.
- Durée de vie (TTL: Time To Live) indique en nombre de sauts le temps pendant lequel un paquet peut rester dans le système. Si ce champ contient la valeur 0, le paquet doit être détruit, Sa valeur est décrémentée à chaque passage dans un routeur.(8bits)
- Protocole: numéro de SAP destinataire du paquet, indique le protocole de la couche supérieure.(8bits)
- Checksum de l'en-tête: contrôle d'intégrité sur l'en-tête (16bits)
- Options: utilisées pour le contrôle ou la mise au point.
- Données (maximum sur 64koctets)

2.2 l'adressage IP:

chaque machine susceptible d'être connectée à l'extérieur de son réseau local possède une adresse IP unique.

Une autorité internationale , le NIC(Network Information center) attribue des numéros à chaque réseau. Les adresses codées sur 32 bits comportent deux parties: le numéro du réseau (Net_id)et le numéro de la machine sur le réseau (Host_id). Le NIC n'affecte que les numéro de réseau. L'affectation des numéros complets est à la charge des administrateurs réseaux. suivant l'importance du réseau , plusieurs classes d'adressage sont possibles:

0	Net_id(sur 7bits)	Host_id (sur 24bits)	Classe A
10	Net_id (sur 14 bits)	Host_id (sur 16bits)	Classe B
110	Net_id(sur 21bits)	Host_id(sur 8bits)	Classe C
1110	Adresse Multicast(28bits)		Classe D

Les adresses sur 32bits sont exprimées par octet (soit 4 nombres compris entre 0 et 255) notées en décimal et séparées par des points

Les différentes classes d'adresse correspondent donc à des nombres appartenant aux plages suivantes:

- **Classe A:** 1.0.0.0 à 126.0.0.0, soit 126 réseaux et 16 777 214 machines par réseau(les réseaux de grande envergure: ministère de la défense, réseau d'IBM....);
- **Classe B:** 128.1.0.0 à 191.254.0.0, soit 16 382 réseaux et 65 535 machines par réseau.(Les réseaux moyens: université, centre de recherche....)
- **Classe C:** 192.0.1.0 à 223.255.254.0 , soit 2 097 150 réseaux et 254 machines par réseau(Les petits réseaux régionaux).
- **Classe D:** 224.0.0.1 à 239.255.255.255, soit 268 435 455 adresses de groupe(ne désigne pas une machine particulière mais un ensemble de machines désirant partager la même adresse).

Remarques:

- L'adresse dont la partie basse est constituée de bits à 0 est une adresse réseau ou sous réseau: 212.92.27.0 pour une classe C.
- L'adresse dont la partie basse est constituée de bits à1 est une adresse de diffusion (broadcast) : 157.42.255.255 pour une classe B.
- 127.0.0.1 est une adresse de bouclage (localhost, loopback)et permet l'utilisation interne de TCP/IP sans aucune interface matérielle.
- 0.0.0.0 : une adresse non encore connue utilisée par une machine ne connaissant pas son adresse IP au démarrage.

2.3. l'adressage de sous-réseaux (subneting)

La partie de l'adresse IP administrée localement (host_id) peut être découpée en deux parties: adresse de sous réseau et numéro de machine.

Un masque de sous réseau ou subnet mask a le même format qu'une adresse IP. Les bits à 1 désignent la partie sous-réseau de l'adresse et les bits à 0 la partie numérotation des machines sur le sous-réseau:

adresse IP: 192.44.77.79 =1100 0000 . 0010 1100 . 0100 1101 . 01 00 1111

netmask: 255.255.255.192 =1111 1111 . 1111 1111 . 1111 1111 . 11 00 0000

adresse de sous-réseau: 192.44.77.64=1100 0000 . 0010 1100 . 0100 1101 . 01 00 0000

00 1111 numéro de machine 15

dans cet exemple, un réseau de classe C , les deux bits de poids fort des 8 bits disponibles sont utilisés pour identifier le sous-réseau.(4 adresses de sous-réseaux possibles: 192.44.77.0, 192.44.77.64, 192.44.77.128, 192.44.77.192; la première à exclure pour ne pas confondre avec l'adresse du réseau/sous réseau, et la dernière pour des raisons de symétrie/diffusion)

Les masques de réseau par défaut pour les classes standard:

- Classe A: 255.0.0.0
- Classe B: 255.255.0.0
- Classe C: 255.255.255.0

3. L'acheminement dans le réseau:

acheminer les informations dans un réseau consiste à assurer le transit des blocs d'un point de sortie à un point d'entrée désigné par son adresse. chaque nœud du réseau comporte des tables dites tables de routage qui indiquent la route à suivre pour atteindre le destinataire (un triplet <adresse destination>/<Route à prendre>/<Coût>). Il faut alors avoir un algorithme de routage qui spécifie les échanges entre les nœuds, le mode de calcul de la route et du coût.

Si un paquet émis par une machine ne trouve pas sa destination dans le réseau ou sous-réseau local, il doit être dirigé vers un routeur qui rapproche le paquet de son objectif. Il faut par conséquent que toutes les stations du réseau possèdent l'adresse du routeur par défaut. La machine source applique le masque de sous-réseau pour savoir si le routage est nécessaire.

Chaque routeur doit alors connaitre l'adresse du routeur suivant lorsque la machine de destination n'est pas sur les réseaux ou sous-réseaux qui lui sont raccordés. Un routeur intègre au moins deux interfaces réseau avec une adresse IP dans chaque réseau connecté, il doit gérer une table de routage de manière statique ou dynamique.

3.1. Les protocoles de routage:

3.1.1. **Routage statique ou fixe:** consiste à construire une table indiquant pour chaque nœud destination, l'adresse du nœud suivant(aucun bouclage de chemin, pratique pour les petits réseaux).

3.1.2. **Routage par diffusion**: Lorsqu'une information doit être routée vers plusieurs destinataire, il faut dupliquer le message en autant d'exemplaire que de destinataires.

3.1.3. **Routage par inondation**: chaque nœud envoie le message sur toutes ses lignes de sortie sauf celle d'où provient le message.

3.1.4**. Routage par le chemin le plus court ou au moindre coût (routage dynamique):** Les routeurs doivent envoyer régulièrement la liste des réseaux ou de sous-réseaux que l'on peut atteindre par eux. Ce qui permet aux autres routeurs de mettre à jour leurs tables de routage. Ils évaluent dynamiquement la meilleure route vers chaque réseau ou sous-réseau. Chaque lien a un coût calculé ou affecté. A partir de ces informations(nombre de sauts, distance réelle en Km, temps de latence dans les files d'attente,...), le routeur détermine le chemin optimal à emprunter .

Deux types d'algorithmes de routage dynamique existent:

- **algorithme à vecteur de distance**: (vector-distance) pour lesquels les informations échangées permettent pour chaque routeur de retenir la plus courte distance (le plus petit nombre de sauts) pour atteindre une destination.
- **algorithme à état des liens** (link-state) : basé sur la transmission d'une carte complète des liens possibles entre les routeurs, ceux-ci doivent ensuite localement calculer les meilleures routes pour une destination.

LES PROTOCOLES DE TRANSPORT

1. Le protocole TCP

1.1. Caractéristiques

TCP est un protocole de la couche Transport au sens du modèle OSI. Il s'exécute au dessus du protocole IP . TCP est un protocole **orienté connexion** qui garantit que les données sont remises de façon **fiable**.

Les fonctionnalités de TCP sont donc principalement :

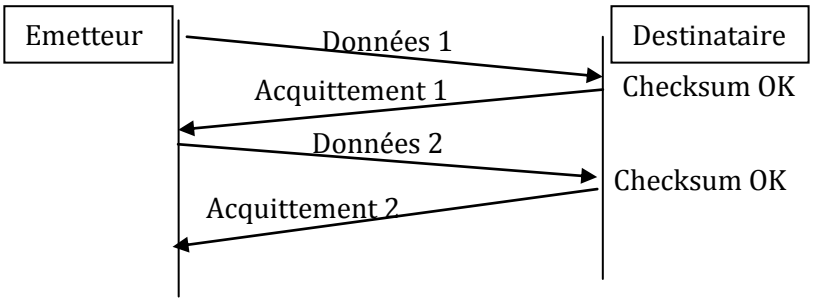
- **Etablissement** d'une **connexion**
- Transmission fiable des données en effectuant un **contrôle des données** et en effectuant une **réémission** pour les données qui n'ont pas pu être transférées correctement.
- **Réordonnancement** des informations transférées. En effet, les informations seront en fait transmises dans des datagrammes IP qui peuvent éventuellement emprunter des chemins différents donc ne pas arriver dans l'ordre d'émission.
- Gérer le **multiplexage**, c'est à dire que plusieurs applications peuvent utiliser simultanément les services du protocole TCP, exemple : un client courrier qui s'exécute en même temps qu'une navigation sur le web et un téléchargement de fichier.
- **Segmentation et séquencement des données:** Lorsque de l'information doit être envoyée d'un émetteur vers un récepteur par le protocole TCP, cette information est découpée en **segments** qui peuvent être de **taille variable**. Mais pour des raisons de fiabilité chaque octet d'un segment va être numéroté avec un **numéro de séquence**, où chaque segment est reconnu par le n° de séquence du premier octet. Les autres numéros (en fait c'est le numéro du dernier qui est intéressant) seront calculés en ajoutant ce numéro au nombre d'octets présents dans la partie "données" du segment. Ce numéro de segment est codé sur 32 bits ce qui permet de numéroter les segments jusqu'à la valeur 2^{32} = 4 294 967 296.

1.2. Mécanisme d'acquittement

La transmission doit être fiable, TCP utilise donc le mécanisme classique d'**acquittement** mais dans une version dite **cumulative**.

1.2.1. Le principe de l'acquittement

Le principe général est assez simple : Lorsqu'un segment est reçu par le destinataire, celui-ci vérifie que les données contenues dans ce segment sont correctes (consultation du checksum) et si c'est le cas envoie un message d'**acquittement positif** (ACK) vers l'expéditeur.



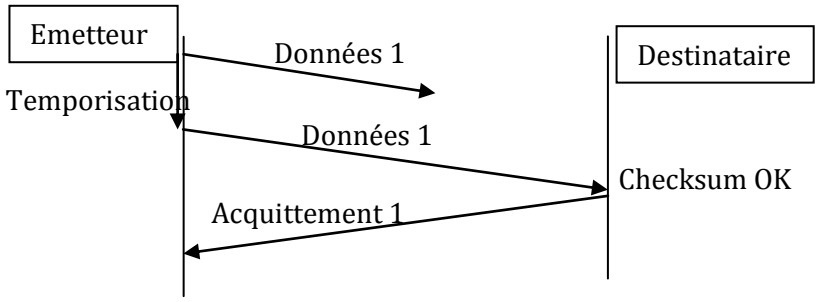
Deux types de problèmes peuvent se produire :

- Les données du segment sont **endommagées**.
- Le segment **n'arrive jamais** à destination.

Pour détecter ce type de problème, chaque fois qu'il envoie un segment l'expéditeur effectue 2 opérations :

- Il stocke dans un buffer une **copie** du segment qu'il vient d'envoyer
- Il arme une **temporisation**

Si au bout d'un certain délai aucun acquittement positif n'a été reçu du destinataire le segment est renvoyé en utilisant la copie présente dans le buffer, si par contre un acquittement est reçu pour ce segment, la copie est supprimée du buffer.

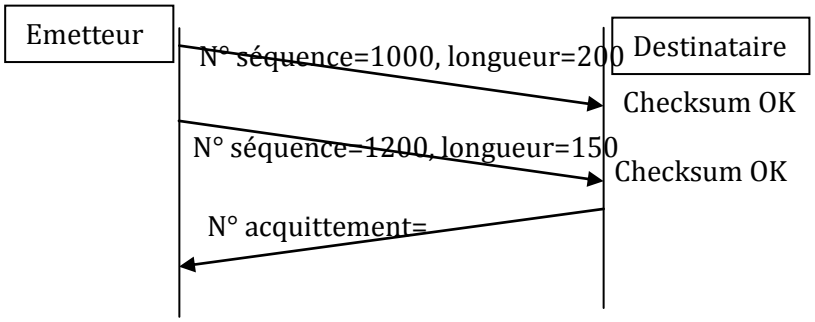


1.2.2. L'acquittement cumulatif

Le protocole TCP utilise le principe de **l'acquittement cumulatif**, c'est à dire que comme les données sont envoyées par segments de taille variable mais comportant le n° de séquence du premier octet du segment, si le contrôle du checksum est satisfaisant, le récepteur déduit à partir du n° de séquence du premier octet et du nombre d'octets reçus le n° de séquence du dernier octet et accuse réception pour cet octet, ce qui implique de façon implicite que tous les octets dont le n° de séquence est inférieur à ce n° de séquence ont été bien reçus.

Il se peut même que pour des raisons d'optimisation, le récepteur attend la réception de plusieurs segments avant d'envoyer un acquittement, ceci a pour but de diminuer le nombre de segments d'acquittements circulant sur le réseau.

Cette technique d'acquittements cumulés a pour principal avantage d'éviter la retransmission de données si un paquet d'acquittement s'est perdu.



1.3. Etablissement d'une connexion

TCP est un **protocole orienté connexion**, cela signifie qu'il va établir et **maintenir** une connexion entre deux machines et **surveiller** l'état de cette connexion pendant toute la durée du transfert.

TCP fonctionne en **full duplex**, c'est à dire que lorsqu'une connexion est établie les données vont pouvoir transiter simultanément dans un sens et dans l'autre.

La demande de connexion peut s'effectuer de 2 manières :

- **passive** (Passive Open), ceci signifie que la machine accepte une connexion entrante. C'est le cas d'un serveur FTP par exemple qui va se mettre en attente de demande d'établissement de connexion de la part d'un client FTP.
- **active** (Active Open) pour demander l'établissement de la connexion.

L'initialisation d'une connexion se fait toujours par ce qui s'appelle une "**Poignée de main à 3 voies**" qui est la traduction littérale de "**Three Way Handshake**", cette initialisation se déroule donc en 3 étapes. Ces 3 étapes ont pour but essentiel de synchroniser les numéros de séquence des 2 machines :

- La machine A envoie un segment de type "ouverture de connexion" avec le n° de séquence X (dans ce segment ne figure aucune donnée)

- La machine B renvoie un segment de type "ouverture de connexion" avec le n° de séquence Y et en acquittant la séquence X envoyée par A
- La machine A renvoie un acquittement à B du segment n° Y

De cette façon chaque machine connaît le n° de séquence de l'autre et l'échange d'information peut débuter.

1.4. Structure des segments TCP

Le segment TCP c'est l'unité de transfert du protocole TCP, il est utilisé indifféremment pour établir les connexions, transférer les données, émettre des acquittements, fermer les connexions.

De façon classique, la structure d'un segment TCP comprend un **entête** de taille variable qui utilise un format en mot de 32bits suivi d'une zone de données.

<table>
<tr><td colspan="4">Port source (16 bits)</td><td colspan="2">Port destination (16 bits)</td></tr>
<tr><td colspan="6">Sequence number (4 octets)</td></tr>
<tr><td colspan="6">Ack number</td></tr>
<tr><td rowspan="2">Data offset
(4bits)</td><td rowspan="2">Réservé
(6 bits)</td><td colspan="6">U A P R S F
R C S S Y I
G K H T N N</td><td rowspan="2">Window
(16 bits)</td></tr>
<tr><td colspan="6">(6 bits)</td></tr>
<tr><td colspan="4">Checksum</td><td colspan="2">Pointeur urgent</td></tr>
<tr><td colspan="6">Options</td></tr>
</table>

- **Source Port et Destination Port (2 x 16 bits)** Ces deux champs de 16 bits chacun contiennent les **numéros de port** de la source et de la destination. Certains numéros de ports sont dédiés à un protocole particulier (par exemple le port 80 est dédié à http).
- **Sequence Number (32 bits)** Ce n° sur 32 bits correspond au **numéro de séquence du premier octet** de données de ce segment de données, en effet le protocole TCP numérote chaque octet envoyé. Si le drapeau SYN vaut 1, ce champ définit le numéro de séquence initial (ISN).
- **Acknowledgment Number (32 bits)** Ce champ sert lorsque le segment est un **segment d'acquittement** (le drapeau **ACK** du champ Flags est à 1), il indique le numéro de séquence du prochain octet attendu (c'est à dire le n° de séquence du dernier octet reçu + 1), tous les octets précédents cumulés sont implicitement acquittés.
- **Data Offset (4 bits)** Ce champ donne la **taille en mots de 32 bits de l'entête du segment**. Si le champ Options est vide, cette taille est égale à 5 (entête de 20 octets).
- **Flags (6 bits)** Ce champ comprend 6 drapeaux qui indique le rôle du segment TCP :
  - ✓ ACK : Indique un segment d'acquittement
  - ✓ SYN : Ouverture de la connexion
  - ✓ FIN : Fermeture de la connexion
  - ✓ RST : Réinitialisation de la connexion pour cause d'erreurs non récupérables
  - ✓ PSH : Demande de remise immédiate des données au processus de la couche supérieure
  - ✓ URG : Données urgentes
- **Window (16 bits) Taille de la fenêtre**, c'est à dire le nombre d'octets que le récepteur est en mesure d'accepter à partir du numéro d'acquittement.
- **Checksum (16 bits)** Le Checksum permet de contrôler si le paquet TCP n'a pas été modifié lors de son transport.
- **Urgent Pointer (16 bits)** Donne la position d'une **donnée urgente** en donnant son décalage par rapport au numéro de séquence. Ce champ n'est utilisé que si le drapeau URG est positionné. Les données urgentes devront passer devant la file d'attente du récepteur, c'est

par exemple avec ce mécanisme qu'il est possible d'envoyer des commandes d'interruption au programme Telnet.

- **Options (variable)** Utilisé à des fonctions de test.
- **Padding (variable)** Octets de bourrage qui permettent de terminer l'en-tête TCP.

1.5. Numéros de port usuels

N° Port	Mot-clé	Description
21	FTP	File Transfer (Control)
23	TELNET	Telnet
25	SMTP	Simple Mail Transfer
80	HTTP	WWW
110	POP3	Post Office Protocol - version 3

2. Le protocole UDP

2.1. Caractéristiques

Le protocole UDP est une alternative au protocole TCP. Comme TCP, il intervient au dessus de la couche IP, au niveau **Transport** au sens des couches OSI.

Les caractéristiques du protocole UDP sont les suivantes :

- identifie les processus d'application à l'aide de **numéros de ports UDP** (distincts des numéros de port TCP)
- possède un **contrôle d'erreurs assez rudimentaire**, il est donc destiné aux réseaux fiables
- UDP est un protocole qui n'est **pas orienté connexion**
- peut éventuellement vérifier l'**intégrité** des données transportées
- Les données ne sont **pas séquencées** donc rien ne permet de vérifier que l'ordre d'arrivée des données et le même que celui d'émission. Ceci le destine plutôt aux réseaux locaux où le mode d'acheminement des informations ne risque pas d'inverser l'ordre des données mais également aux applications qui véhiculent des informations de petites tailles qui peuvent tenir en un seul datagramme.
- De par sa structure UDP est **plus rapide** que TCP, mais **moins robuste**

UDP est donc un **protocole orienté commande/réponse.** UDP peut être utile pour les applications qui nécessitent une **diffusion** d'informations car dans ce cas il serait pénalisant d'utiliser un protocole comme TCP orienté connexion qui devrait gérer (ouvrir et fermer) autant de connexion que de nœuds auxquels l'information est destinée.

2.1. Structure du paquet UDP

0	**16**	**31**
Source Port	Destination Port	
Length	Checksum	
Data		

- **Source Port et Destination Port** : port source et destination
- **Length** : Longueur du paquet UDP
- **Checksum** :champ de contrôle des données.

LES PROTOCOLES D'APPLICATION

Internet est un ensemble de réseaux interconnectés utilisant tous les mêmes protocoles de routage et de transport TCP/IP. Il permet d'accéder à des services comme la messagerie électronique (e-mail), le transfert de fichiers (FTP) ou les serveurs d'informations en ligne(serveurs Web).
Dans l'organisation d'internet, on distingue:

- Les opérateurs (câblage et transport);
- les prestataires de services ou fournisseurs d'accès aux services(ISP: Internet Service Provider);
- les services et les protocoles associés.

1. Les opérateurs:
ils disposent de leur réseau et assurent le transport des informations d'un point à un autre. Ces réseaux sont organisés en réseaux régionaux, interconnectés par des réseaux nationaux. Les opérateurs fournissent les points de connexion sur leur réseau aux entreprises et aux prestataires de services qui ont obtenu des adresses IP d'un organisme agréé.

2. Les prestataires
Les prestataires, connectés à un réseau Internet, fournissent:

- des adresses IP aux particuliers qui ne peuvent obtenir d'adresse auprès de l'InterNIC (Les adresses ne peuvent être attribuées que par blocs de 256 au minimum);
- des services tels que la messagerie, la connexion aux serveurs Web ou l'hébergement de pages Web;
- des services de connexion utilisant les réseaux d'opérateurs de télécommunication.

Les prestataires sont également des opérateurs Internet.

3. Les services
3.1. Service de messagerie
connu sous le nom de "e-mail", permet d'échanger des messages et des fichiers. Il nécessite un serveur de messagerie accessible à partir d'Internet. Le serveur dispose d'une boîte à lettre pour chaque client géré par la messagerie.
Les messages sont stockés par le serveur de messagerie, en attendant que le client vienne consulter sa boîte aux lettres, le message peut alors être lu.
Pour la mise en forme des messages (jeu de caractères, encodage, codage des fichiers joints...) le protocole le plus utilisé est MIME (Multipurpose Internet Mail Extensions). Il permet aussi la mise en forme du texte(mots soulignés, caractères en gras...)

3.2. Service de transfert de fichiers
permet à un client de récupérer des fichiers auprès d'un serveur de fichiers. La connexion et le dialogue entre la station du client et le serveur utilisent le protocole FTP (File Transfert Protocol).
Après s'être connecté au serveur, celui-ci demande un nom de compte et un mot de passe au client.

3.3. service Web
permet d'accéder à des documents au format HTML (Hyper Text Markup Language) en utilisant pour la connexion est les échanges le protocole HTTP (Hyper Text Transfert Protocol).

Les documents sont accessible par un URL(Uniform Ressource Location)comportant le nom du serveur http contenant le document, le chemin d'accès au document et le nom de celui-ci.
Pour accéder aux serveurs Web, les stations doivent être équipées de navigateurs.

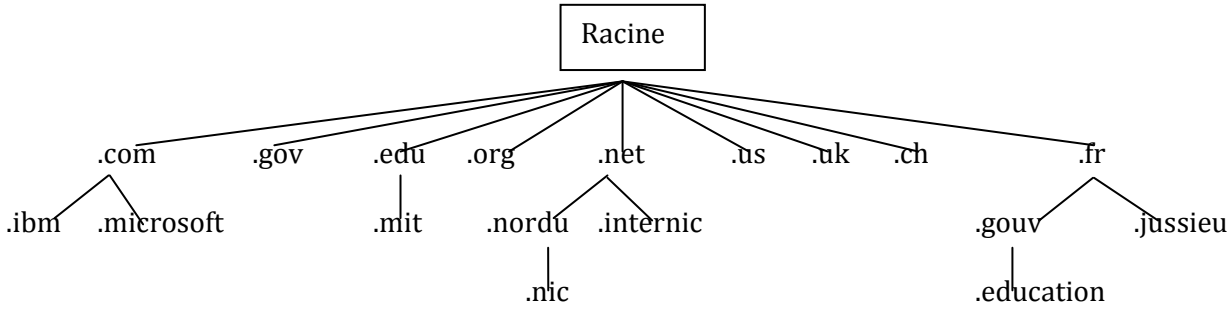
4. Les protocoles

4.1. DNS

Pour simplifier l'identification, un service de résolution permettant d'utiliser des noms symboliques de machines à la place des adresses IP est utilisé sur tous les réseaux TCP/IP.
La méthode la plus simple passe par l'utilisation d'un fichier sur la machine émettrice qui comprend des noms et les adresses IP correspondantes. Cette méthode impose la mise à jour d'un même fichier sur toutes les machines (envisageable que pour des réseaux de quelques postes).

Pour les réseaux de grandes envergures tel que Internet, une méthode consiste à centraliser les gestion des noms sur des machines spécifiques (les serveurs de noms)à l'aide d'un service permettant une organisation hiérarchisée: Les DNS(Domain Name Service). Ce service travaille suivant une organisation arborescente en divisant le réseau global en un ensemble de domaines primaires, secondaires, ...

Exemple de nommage hiérarchique (contrairement à l'adressage IP pour lequel l'identifiant de la machine est situé sur la partie droite, le nom DNS présent le nom de la machine à gauche suivi des noms de domaines d'importance croissante.



ex: PC_du_Ministre.educatin.gouv.fr
.fr: domaine racine
.gouv: domaine de niveau 2
.education: domaine de niveau 3
PC_du_Ministre: sous domaine ou nom de machine

Chaque serveur de nom DNS gère une ou plusieurs zones du réseau. Chacune des zones possède au moins un serveur de noms ayant la connaissance complète des adresses des machines de la zone. Chaque serveur de nom connait également l'adresse d'au moins un autre serveur de noms.
Coté client, chaque machine possède au moins l'adresse d'un serveur DNS afin de pouvoir résoudre une adresse symbolique en une adresse réseau (IP).

4.2. SMTP

SMTP signifie **Simple Mail Transfer Protocol,** c'est le protocole standard permettant de transférer du courrier d'un serveur (on parle de serveur SMTP) à un autre. Le protocole SMTP

fonctionne en **mode connecté**, le déroulement d'une connexion comprend toujours les étapes suivantes :
1. L'ouverture de la session symbolisé par la commande HELO ou EHLO sur les versions plus récentes;
2. Un envoi MAIL FROM qui indique qui est l'expéditeur du message;
3.Un envoi RCPT TO qui indique le destinataire (s'il y a plusieurs destinataires, cette commande est répétée autant de fois que nécessaire)
4. Un envoi DATA qui correspond au corps du message .

Le port utilisé par défaut est le port 25.
La spécification de base du protocole SMTP indique que tous les caractères transmis dans un mail soient codés en ASCII sur 7 bits. Afin de pouvoir envoyer des caractères de la table ASCII étendue (accents, etc...) il est donc nécessaire d'utiliser des système de transcodage (spécifications **MIME** : Multipurpose Internet Mail Extensions) pour coder les fichiers attachés et les caractères spéciaux contenus dans le corps du message.

Il existe d'autres protocoles liés à la messagerie électronique :

- **POP** (**Post Office Protocol**)

Ce protocole permet d'aller récupérer son courrier sur un serveur distant. Ce protocole est indispensable pour les personnes qui ne sont pas connectés directement à l'internet, afin de rapatrier leurs mails sur leur machine. La version POP3 gère l'authentification par nom d'utilisateur et mot de passe, par contre le cryptage n'est pas utilisé.

- **IMAP** (**Internet Mail Access Protocol**)

Ce protocole considéré comme une alternative à POP offre des possibilités supplémentaires. La principale est que le client de courrier peut demander les en-têtes des messages au serveur, ou les corps de certains messages, ou la recherche de messages répondant à certains critères ce qui permet une plus grande souplesse d'utilisation (lecture du message directement sur le serveur, pas besoin de le télécharger sur sa machine).

4.3. FTP:

FTP signifie **File Transfer Protocol,** c'est un protocole de transfert de fichier qui utilise **TCP**. Le mode de fonctionnement est de type client-serveur. FTP permet les opérations suivantes :

- transfert de fichiers du serveur vers le client (download) ;
- transfert de fichiers du client vers le serveur (upload);
- renommage et suppression depuis le client de fichiers stockés sur le serveur;
- listage depuis le client de répertoire situés sur le serveur.

Le mode opératoire standard de FTP est le suivant :

- ✓ Le client FTP initie une connexion avec le serveur FTP. Il doit s'authentifier auprès de ce serveur c'est à dire fournir un identifiant et un mot de passe;
- ✓ Le serveur FTP étant en écoute permanente de demande de connexion, il reçoit donc cette demande de connexion . Le serveur initie donc une connexion TCP dite de connexion de contrôle qui servira à transférer les commandes TCP;
- ✓ Chaque fois que le client FTP exécute une commande de transfert de données, il envoie au serveur FTP cette demande accompagnée du n° de port local utilisé;
- ✓ Si le serveur TCP reçoit une commande de transfert de données, il initie une connexion dite connexion de données.

Généralement un transfert de fichier par ftp se fait en utilisant côté serveur les ports 21 pour les opérations de contrôle et le port 20 pour les données.

- **TFTP**

TFTP signifie **Trivial File Transfer Protocol**, c'est un autre protocole de transfert de fichier mais qui utilise **UDP** comme protocole de transfert. Le protocole TFTP est plus simple à implanter que FTP mais il ne permet pas l'utilisation d'un répertoire utilisateur sur le serveur, ni celle d'un mot de passe garantissant une la protection des données.

4.4. HTTP:

HTTP signifie **H**yper**T**ext **T**ransport **P**rotocol, c'est un protocole **léger** et **rapide** utilisé pour délivrer des fichiers **multimédia** et **hypertextes**, appelés plus généralement "ressources", en utilisant **internet**. Ces ressources sont identifiées par un **URL** : **U**niform **R**essource **L**ocator. Une ressource peut être un fichier (fichier texte codé en HTML par exemple ou image de type jpeg, gif, etc...) ou du texte **HTML** (**H**ypertext **T**ransfer **P**rotocol) généré dynamiquement par un script **CGI** (**C**ommon **G**ateway **I**nterchange).
HTTP se base sur un **système requête/réponse** entre un **client HTTP** (un **navigateur** en fait) et un **serveur HTTP.**
Par défaut, le n° de port utilisé par le serveur HTTP est le **port 80 .**

- **Les URL**

Un URL permet de localiser des ressources sur le Web. Il se compose de plusieurs éléments :

- ✓ Le **protocole** (ou **classification**): http ou **https** (http sécurisé);
- ✓ Le **nom d'hôte** : Le nom du serveur ou éventuellement son adresse IP;
- ✓ Le **numéro de port** (facultatif) : Par défaut ce n° de port est 80 pour HTTP et 443 pour HTTPS, mais il est possible d'utiliser un autre n° de port (précédé par :);
- ✓ Le **chemin** : C'est l'emplacement de la ressource demandée, cela n'a pas forcément de lien avec un chemin disque dur car les serveurs gèrent des **alias** qui permettent de remplacer une portion de chemin par un littéral;
- ✓ La **chaîne de requête** (facultatif) : Permet de passer des paramètres supplémentaires aux scripts. Cette chaîne lorsqu'elle existe est constituée d'un ensemble de paires **nom=valeur** séparées par le symbole **&**, la chaîne elle même débute par le caractère **?;**
- ✓ L'**identificateur de fragment** (facultatif) : Désigne une section spécifique d'une ressource, un début de paragraphe dans une page web par exemple. Cet identificateur n'est utilisé que par les navigateurs afin de décaler sa fenêtre de visualisation en conséquence.

- **Les Requêtes-Réponses**

Lorsqu'un navigateur demande une page (l'utilisateur a cliqué sur un lien ou a tapé un URL dans la fenêtre de navigation par exemple), celui-ci envoie au serveur une **requête** HTTP. Le serveur lui répondra par une **réponse** HTTP qui généralement (mais pas toujours..) contient la ressource demandée par le serveur. Le protocole HTTP jusqu'à la version 1.0 était un protocole qui fonctionne en mode **non connecté** (stateless) c'est à dire qu'après l'échange requête-réponse la connexion n'est pas maintenue, depuis la version 1.1, la connexion peut désormais être maintenue.

La structure des requêtes et des réponses est la même, elle est constituée de :

- ✓ un entête
- ✓ un corps

Structure de l'entête : une ligne initiale + couples (champ d'entête : valeur)

La **ligne initiale** : de forme différente si l'on est dans une requête ou dans une réponse.
Dans une requête, la ligne initiale est composée de 3 parties :

- La **méthode de requête** (exemple : **GET,POST,HEAD,...**)
  - GET: demande au serveur la ressource indiquée;
  - HEAD:ne demande que les en-tête mais pas la ressource complète

- POST: demande au serveur de modifier l'informations qu'il stocke
- L'**URL** de la ressource concernée;
- Le **protocole** utilisé HTTP/x.x (exemple HTTP/1.1)

Exemple : GET /index.htm HTTP/1.1

Dans une réponse, la ligne initiale appelée **ligne status**, est composée de 3 parties :

- Le **protocole** utilisé HTTP/x.x (exemple HTTP/1.1);
- Le **code d'état** a 3 chiffres;
- La **version textuelle** en langue anglaise correspondant à l'état

Exemple : HTTP/1.0 404 Not Found.

Les principaux codes d'état :

Il existe 5 classes de codes d'état :

- 1xx : utilisés à bas niveau lors des transaction HTTP ;
- 2xx : la requête s'est bien passée (200 : OK);
- 3xx : la requête est correcte mais la ressource n'est plus là ou le serveur ne veut pas l'envoyer comme c'est le cas pour le code 304 qui signifie que la ressource n'a pas été modifiée depuis le dernier envoie donc il est inutile de la renvoyer;
- 4xx : la requête est incorrecte (exemple 404 : la ressource n'existe plus);
- 5xx : erreur du serveur (exemple script côté serveur de syntaxe incorrecte)

Les champs d'entête : Les champs d'entête sont de la forme:" **champ d'entête : valeur"**. Il en existe 16 différents pour HTTP 1.0 et 46 pour HTTP 1.1. Le seul requis pour le protocole HTTP 1.1 est le champ **HOST**.

Ex: **Accept-Language :** Langages acceptés par le navigateur
Authorization : Nom et mot de passe de l'utilisateur demandant la ressource
Content-Length : Longueur du contenu de la requête.
Host : Nom de l'hôte cible

La structure du corps :

Pour une requête, le corps contiendra des couples de nom-valeur dans le cas de la méthode POST, ou le contenu d'un fichier dans le cas d'un upload de fichier vers le serveur. Pour une réponse, ce corps contient généralement une ressource destinée au navigateur comme du code HTML par exemple.

4.5. TELNET

Ce protocole est utilisé pour émuler une connexion de terminal à un hôte distant. Le but de ce protocole est donc de transmettre les informations du clavier du client vers l'hôte distant et, dans l'autre sens, d'afficher les informations en retour sur l'écran du client. Telnet utilise **TCP** comme protocole de transport. Le mode de fonctionnement est de type client-serveur. Généralement côté serveur c'est le port 23 qui est utilisé.

Une connexion TELNET débute toujours par une phase de **négociation** qui a pour but de déterminer la configuration du client utilisé comme par exemple la façon dont les données vont être groupées avant d'être envoyées (ligne par ligne ou caractère par caractère).

Telnet utilise le concept de terminal virtuel qui permet de s'affranchir de la multiplicité des terminaux. Un terminal virtuel consiste à se doter d'une base de communication standard comprenant le codage des caractères ASCII et de quelques caractères de contrôle.