Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed Khider - Biskra
Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie
Département d'informatique

$2^{ème} année LMD$

Cours de Réseaux de Communication

Dr. Abdelhamid DJEFFAL
Site web : www.abdelhamid-djeffal.net

Année Universitaire 2017/2018

Plan du cours

Références

[1] Douglas Comer. Tcp/ip : Architecture, protocoles, applications. troisième édition. intereditions, 1996.

[2] Dominique Dromard, Danièle et Seret. *Architecture des réseaux.* Pearson Education France, 2009.

[3] Guy Pujolle. *Les réseaux : Edition 2014.* Editions Eyrolles, 2014.

[4] Pierre Rolin. *Réseaux locaux : normes et protocoles.* Hermès, 1993.

[5] A. Tanenbaum. *Réseaux : Architectures, Protocoles et Applications.* InterEditions, 1995.

Chapitre 1

Introduction aux réseaux informatiques

Les réseaux informatiques de nos jours sont devenus indispensables dans, pratiquement, dans tous les domaines de la vie : banques, assurance, sécurité, internet, santé, administration, transport, ...

Les besoins de communication de données informatiques entre systèmes plus ou moins éloignés sont multiples : transmission de messages (messagerie), partage de ressources (imprimante, disque dur, internet), transfert de fichiers (FTP), consultation de bases de données, gestion de transactions, télécopie ...

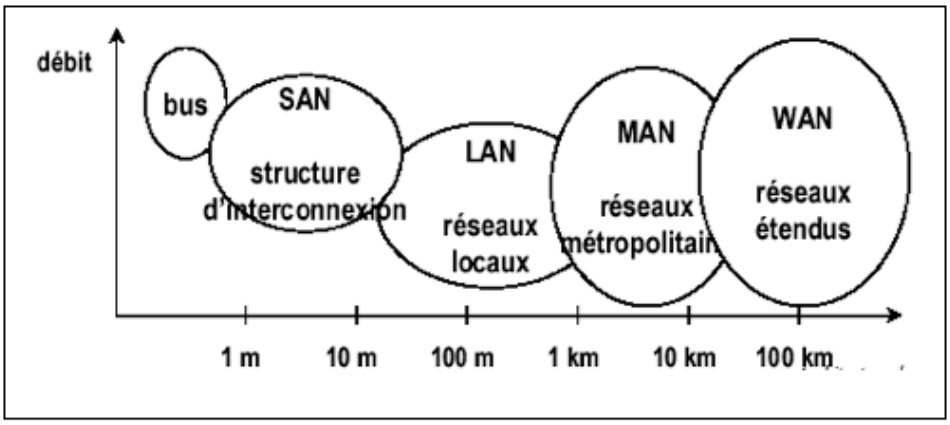
1.1 Définition d'un réseau informatique

C'est un ensemble d'ordinateurs et de périphériques autonomes connectés entre eux et qui sont situés dans un certain domaine géographique.

1.2 Types de réseaux

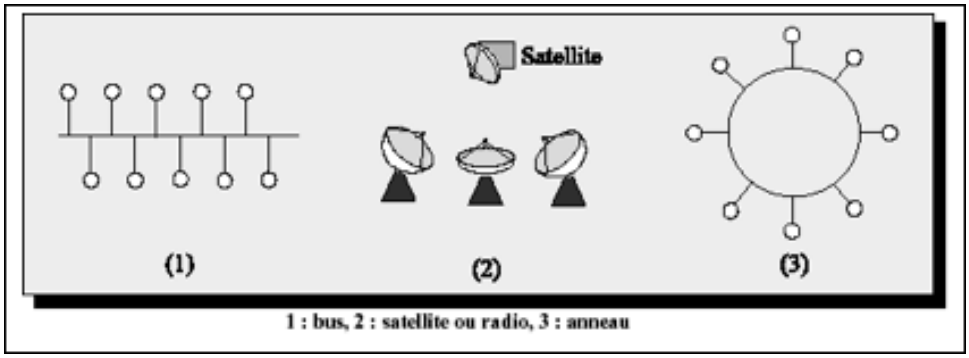
- Bus : Communication entres composants < 1m
- Architectures parallèles (réseaux d'interconnexions) >10m
- Réseaux locaux (LAN) : correspondent par leur taille à des réseaux intra - entreprises. La distance de câblage est de quelques centaines de mètres
- Réseaux métropolitain (MAN) : Correspondent à une interconnexion de quelques bâtiments se trouvant dans une ville (Campus).
- Réseaux étendus (WAN) destinés à transporter des données à l'échelle d'un pays.

Ces réseaux peuvent être terrestres (Utilisation d'infra - structure au niveau : câble, fibre, ...) ou satellite (Mise en place d'engins spatiaux pour retransmettre les signaux vers la terre).

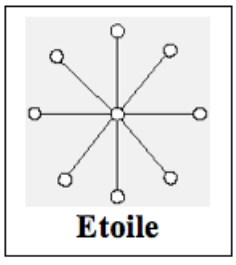


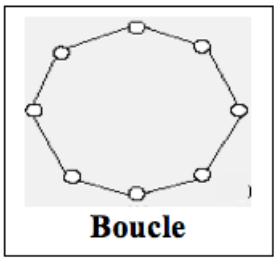
1.3 Modes de communication

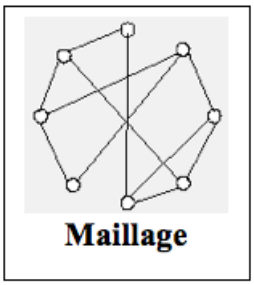
– Mode diffusion : Un émetteur ⇒ Plusieurs récepteurs



– Mode point à point : Un émetteur ⇒ Un récepteur







1.4 Fonctionnement des communications

Quelque soit l'architecture, on a deux modes de communications :

– **Avec connexion** (permanente) ⇒ demande de connexion

1. Emetteur demande,
2. Si récepteur refuse ⇒ pas de communication.
3. Sinon circuit virtuel,
4. Transfert de données,
5. Libération de la connexion
6. Lourde si peu d'informations (gaspillage du réseau)
7. Difficulté des communications multiples.

– **Sans connexion** (Sans demande de connexion)

1. Sans vérification que le récepteur est actif : boites aux lettres.
2. Les organes du réseau gèrent les communications.
3. Utilisations des buffers si le récepteur n'est pas actif

1.5 Différentes techniques de commutation

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

Définition : La commutation rassemble toutes les techniques qui réalise la mise en relation de 2 abonnés quelconques.

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.

– **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.

– **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.

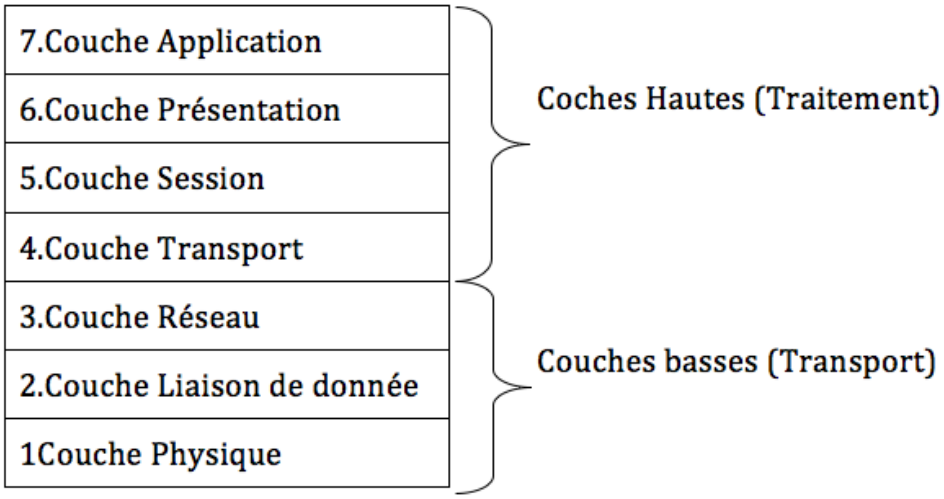
– **Commutation de cellule :** commutation de paquets particulière. Tous les paquets ont une longueur fixe (1 paquet = 1 cellule de 53 octets dans ATM). Un chemin est déterminé pour la transmission des cellules. Commutation de cellule = superposition de 2 types de commutation : commutation de circuit et commutation de paquets.

Chapitre 2

Modèle OSI

A la fin des années 70 on a connu le développement de plusieurs solution réseaux indépendantes (SNA d'IBM, DECNET de DEG, DSA de Bull...) et on avait besoin d'une norme internationale pour inter communiquer.

L'ISO (International Standard Organisation) a pris en charge l'établissement de l'OSI (Open system interconnections : norme d'interconnexion des systèmes ouverts), cette norme se présente sous la formes de sept couches :



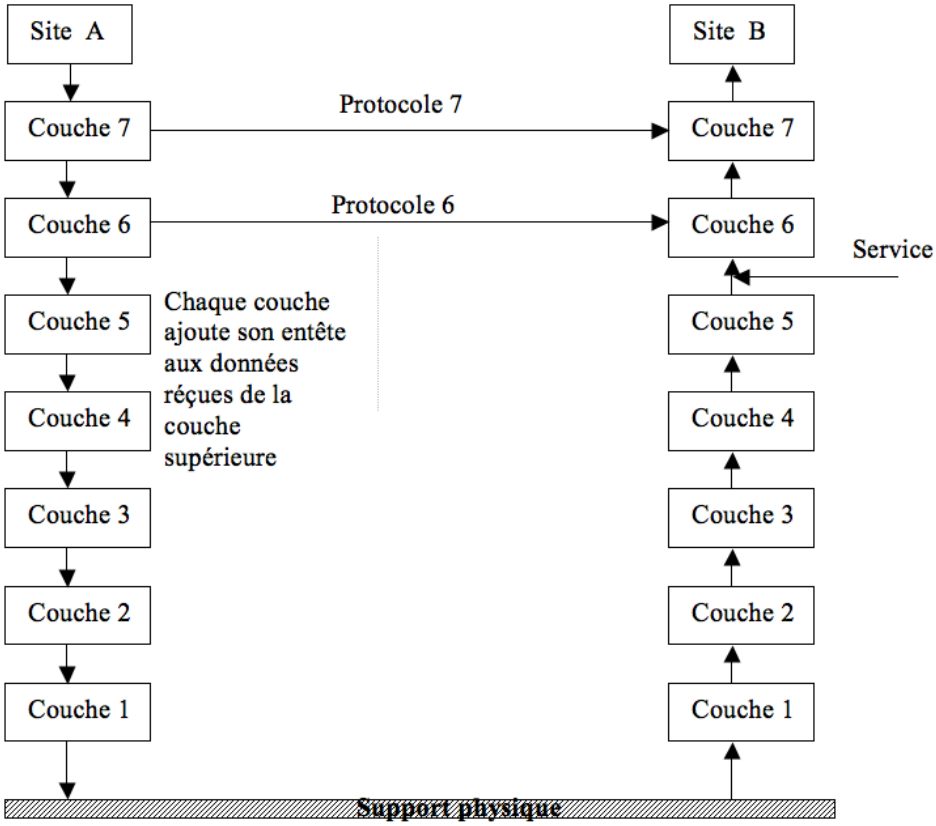
1. **Couche physique :** Etudie les signaux porteurs des informations (modulation, puissance, portée) ainsi que les supports de transmission (câbles, fibres optiques,...).
2. **Couche liaison de donnée :** Responsable de l'établissement, le maintient et la libération des connexions entre les éléments du réseau. Elle est responsable aussi de la détection et la correction des erreurs.
3. **Couche réseau :** Responsable de l'adressage des machines, et le routage des données dans le réseau.

4. Couche transport : Assure un transfert transparent de données entre les utilisateurs, en leur rendant invisible la façon dont les ressources de communications sont mises en œuvre.
5. **Couche session :** Assure l'optimisation et le réglage de quelques problèmes non dus au réseau tel que la reprise de transfert d'un fichier long après une erreur d'accès au disque.
6. **Couche présentation :** Assure des fonctions tel que la compression des données, la représentation des données (Exemple : poids fort à gauche ou à droite).
7. **Couche application :** Permet d'offrir aux logiciels les mêmes principes et standards d'accès aux réseaux (Notion de fichier virtuel).

2.1 Communication par couches

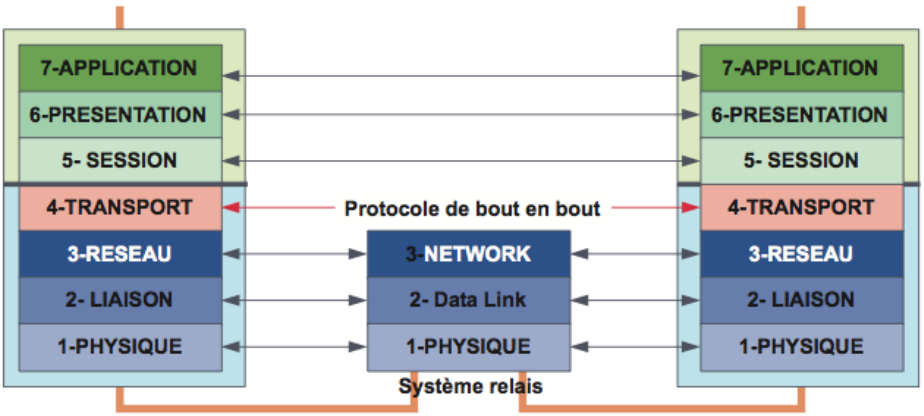
On appelle **protocole**, un dialogue connu par les deux parties, entre deux couches de même niveau. Une couche de niveau (n) ne sera capable de dialoguer qu'avec une autre couche de même niveau qu'elle.

On appelle **service** l'ensemble des fonctions que doit absolument remplir une couche, fournissant l'interface pour transmettre des données de la couche (n) à la couche (n+1).



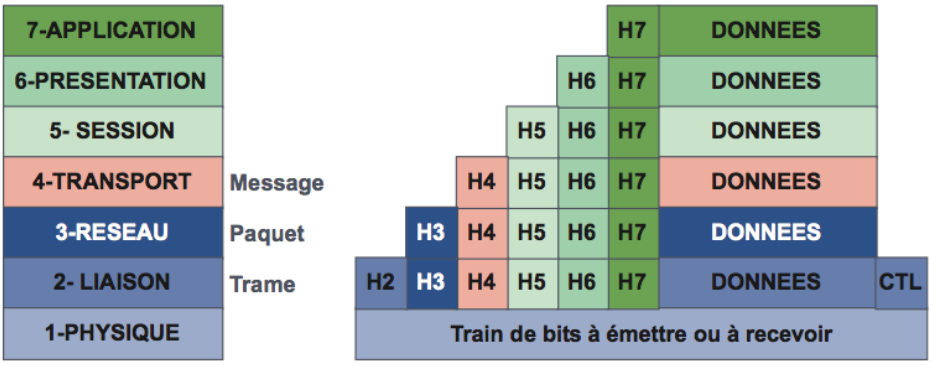
Pour réaliser une communication à travers un ou plusieurs systèmes intermédiaires (relais) il faut :

1. relier les systèmes par un lien physique (couche PHYSIQUE) ;
2. contrôler qu'une liaison peut être correctement établie sur ce lien (couche LIAISON) ;
3. s'assurer qu'à travers le relais (réseau) les données sont correctement acheminées et délivrées au bon destinataire (couche RÉSEAU) ;
4. contrôler, avant de délivrer les données à l'application que le transport s'est réalisé correctement de bout en bout (couche TRANSPORT) ;
5. organiser le dialogue entre toutes les applications, en gérant des sessions d'échange (couche SESSION) ;
6. traduire les données selon une syntaxe de présentation aux applications pour que celles-ci soient compréhensibles par les deux entités d'application (couche PRÉSENTATION) ;
7. fournir à l'application utilisateur tous les mécanismes nécessaires à masquer à celle-ci les contraintes de transmission (couche APPLICATION).



2.2 Encapsulation

Lorsqu'une application envoie des données à travers le modèle OSI, les données traversent de haut en bas chaque couche jusqu'à aboutir au support physique où elles sont alors émises sous forme de suite de bits. Chaque couche ajoute aux données reçues de la couche supérieure des informations appelées "Entête" avant de les transférer à la couche inférieure. Cette entête permet à cette couche d'accomplir son rôle. A la réception, chaque couche retire l'entête, ajouté par la couche du même niveau, des données avant de les transmettre à la couche supérieure.

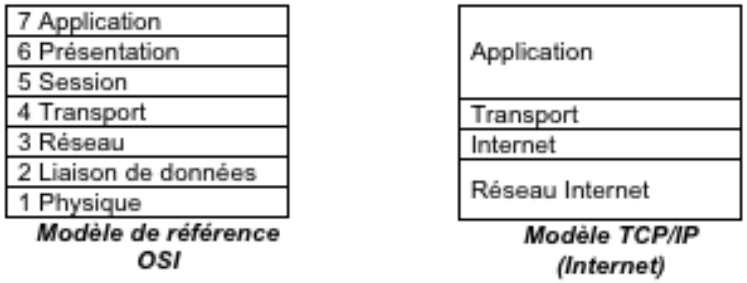


2.3 Modèle TCP/IP

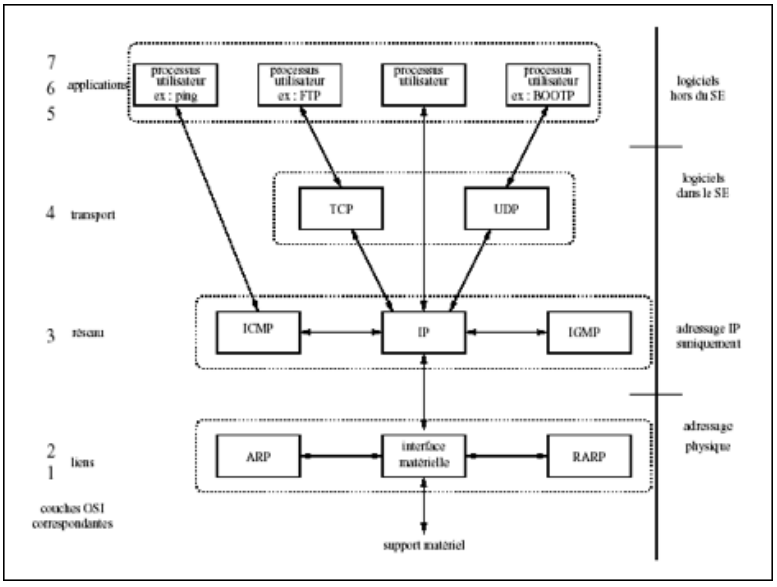
La série des protocoles TCP/IP (Transmission Control Protocol / Internet Protocol) provient du projet DARPA (Defense Advanced Research Project Agency) de l'agence de recherche de projets avancés de la défense des USA vers la fin des années 60. Les protocoles ont évolué dans les années 70 pour passer en 1980 à l'utilisation dans les équipes de recherche et les universités des USA. Le nombre de réseaux utilisant TCP/IP a rapidement

augmenté pour former une grande communauté appelée l'Internet. Aujourd'hui, le bureau d'activités inter réseaux ou l'IAB (Internet Activities Board) se charge du développement et de ratification des protocoles TCP/IP.

L'architecture TCP/IP est similaire au modèle OSI en couche, mais ne dispose que de 4 couches dans la plupart des cas.



- Couche application : FTP, TELNET, HTTP, SMTP.
- Couche Transport : TCP (fiable), UDP (non fiable)
- Couche Internet : IP, ICMP
- Couche réseau Internet : Interface avec le réseau utilisé : ARP.

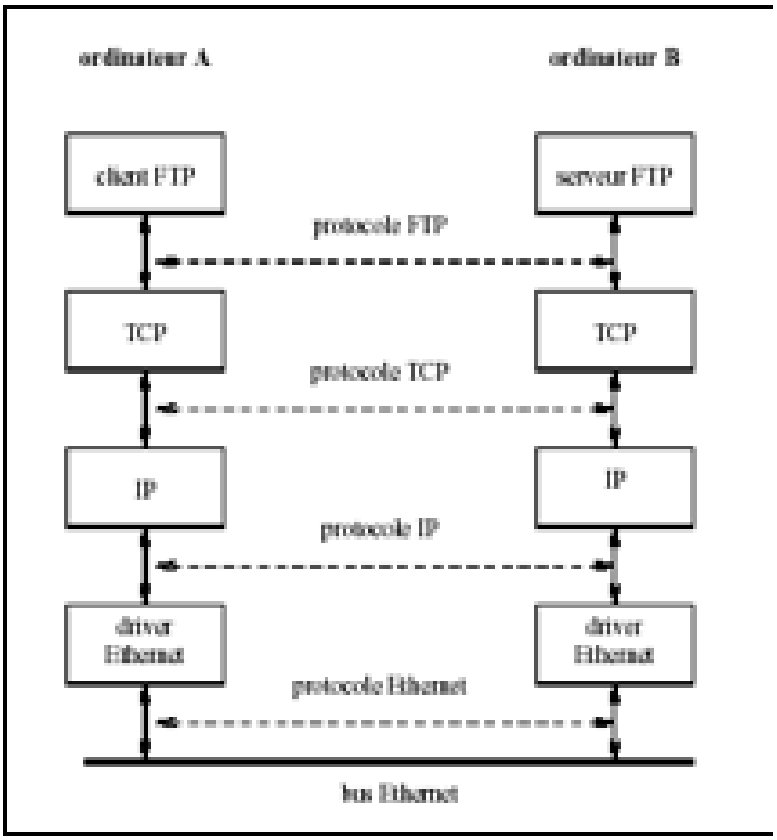


- **La couche de liens** est l'interface avec le réseau et est constituée d'un driver du système d'exploitation et d'une carte d'interface de l'ordinateur avec le réseau (carte réseau, modem,..).
- **La couche réseau** ou couche IP (Internet Protocol) gère la circulation des paquets à travers le réseau en assurant leur routage. Elle comprend aussi les protocoles ICMP (Internet Control Message Protocol) et IGMP (Internet Group Management Proto-

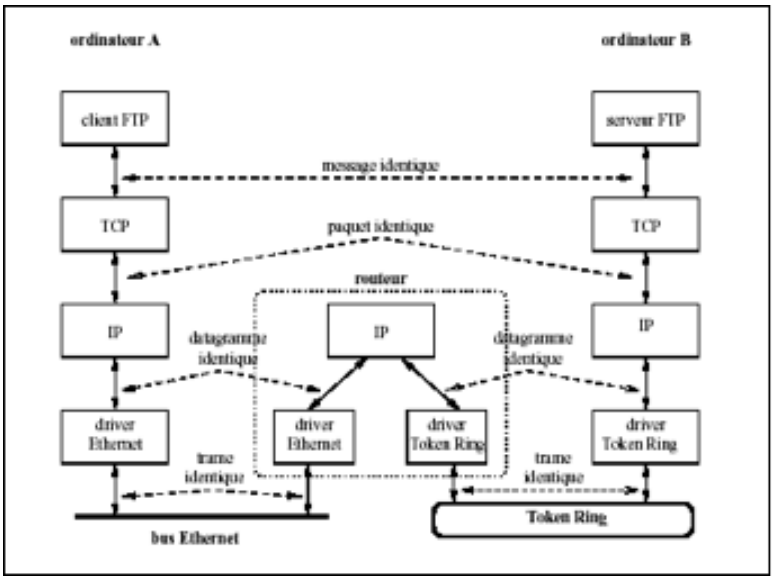
col)

- **La couche transport** assure tout d'abord une communication de bout en bout en faisant abstraction des machines intermédiaires entre l'émetteur et le destinataire. Elle s'occupe de réguler le flux de données et assure un transport fiable (données transmises sans erreur et reçues dans l'ordre de leur émission) dans le cas de TCP (Transmission Control Protocol) ou non fiable dans le cas de UDP (User Datagram Protocol). Pour UDP, il n'est pas garanti qu'un paquet (appelé dans ce cas datagramme) arrive à bon port, c'est à la couche application de s'en assurer.
- **La couche application** est celle des programmes utilisateurs comme telnet (connexion à un ordinateur distant), FTP (File Transfert Protocol), SMTP (Simple Mail Transfert Protocol), etc...

Cette architecture et ces différents protocoles permettent de faire fonctionner un réseau local, par exemple sur un bus Ethernet reliant un ordinateur client A qui interroge un serveur FTP B :



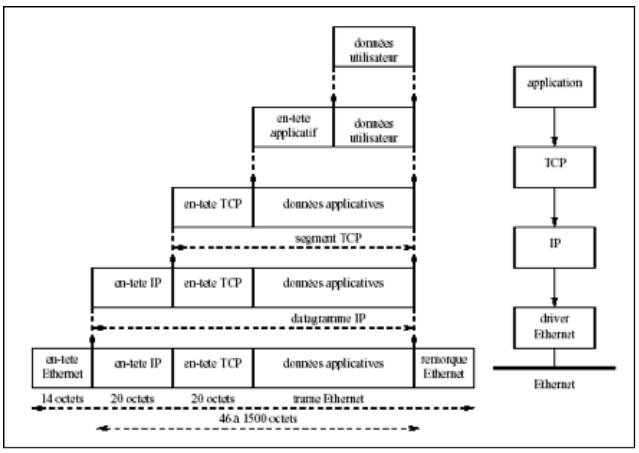
Mais, ceci permet surtout de constituer un Internet, c'est-à-dire une interconnexion de réseaux éventuellement hétérogènes :



Les ordinateurs A et B sont des systèmes terminaux et le routeur est un système intermédiaire. Comme on peut le voir, la remise du datagramme nécessite l'utilisation de deux trames différentes, l'une du réseau Ethernet entre la machine A et le routeur, l'autre du réseau Token-Ring entre le routeur et la machine B. Par opposition, le principe de structuration en couches indique que le paquet reçu par la couche transport de la machine B est identique à celui émis par la couche transport de la machine A.

Lorsqu'une application envoie des données à l'aide de TCP/IP les données traversent de haut en bas chaque couche jusqu'à aboutir au support physique où elles sont alors émises sous forme se suite de bits. Chaque couche ajoute aux données reçues de la couche supérieur des informations appelées "Entête" avant de les transférer à la couche inférieure. Cette entête permet à cette couche d'accomplir son rôle.

A la réception, chaque couche retire l'entête des données avant de les transmettre à la couche supérieur



Chapitre 3

Couche physique

Le rôle de la couche physique est de transformer une suite de bits en signaux (et inversement) pour les adapter au canal de communication et les transmettre d'une machine à une autre. Les bits transformés représentent des informations numérisées (codées) tel que le code ASCII pour les textes, avi pour le multimédia, ...etc.

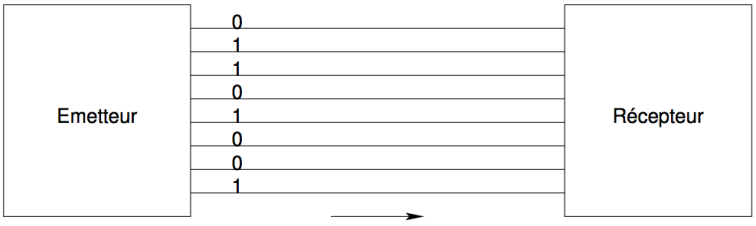
La couche physique détermine la façon selon laquelle les bits sont transportés sur le support physique. Elle permet d'introduire les bits 0 et 1 sur le support sous une forme spécifique, reconnaissable par le récepteur. Plusieurs composants sont utilisés dans cette couche, comme les modems, multiplexeurs, concentrateurs, etc. Ce chapitre étudie les supports de transmission et leurs caractéristiques ainsi que les méthodes utilisées pour transmettre l'information sur ces supports.

3.1 Modes de transmission

Les blocs d'informations transmis sur des fils peuvent l'être en parallèle ou en série.

3.1.1 Transmission en parallèle

Dans la transmission parallèle, les bits d'une même entité (octet, mot, ...) sont envoyés sur des fils distincts pour arriver ensemble à destination. On peut avoir 8, 16, 32 ou 64 fils parallèles.



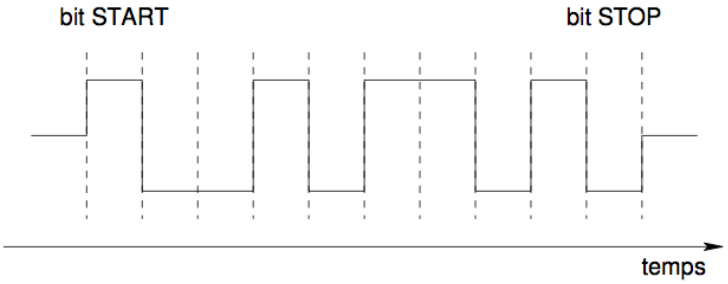
La transmission parallèle pose des problèmes de synchronisation à cause des déphasages possibles entre les différents fils. C'est pour cette raison que ce mode n'est utilisé que sur de très courtes distances tel que le bus d'un ordinateur.

3.1.2 Transmission en série

Dans ce mode de transmission, les bits sont émis les uns après les autres. C'est le mode utilisé dans la réseaux informatiques, il peut être asynchrone ou synchrone.

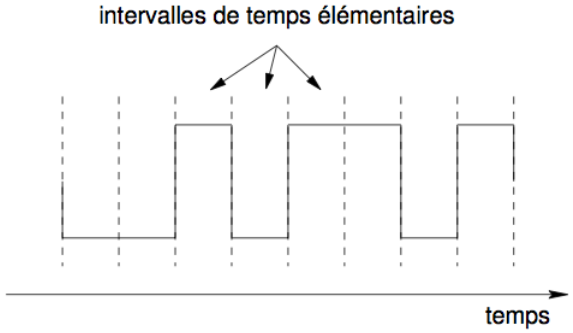
3.1.2.1 Transmission asynchrone

La transmission peut être effectuée n'importe quand, et ne dépend pas d'intervalles de temps précis. Le récepteur commence la réception à l'arrivée d'un bit START et la conclue à l'arrivée d'un bit STOP



3.1.2.2 Transmission synchrone

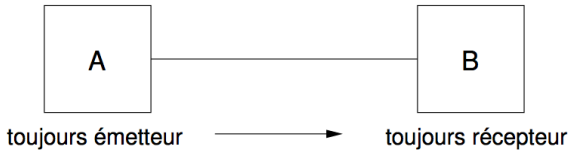
Dans la transmission synchrone, l'émetteur et récepteur sont d'accord sur un intervalle de temps élémentaire constant qui se répète sans cesse. L'émetteur transmet en début d'intervalle pour une durée d'un intervalle par information (ex : 1 bit)



Ce mode est utilisé pour les très hauts débits

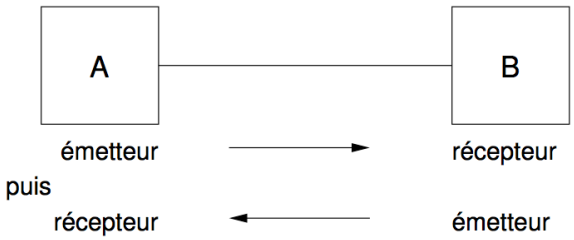
3.1.3 Transmission en simplexe

Dans certains cas d'échange d'information une partie est toujours émettrice et l'autre est toujours réceptrice. Les données circulent toujours dans le même sens. L'exploitation du canal de transmission est appelée dans ce cas en simplexe.



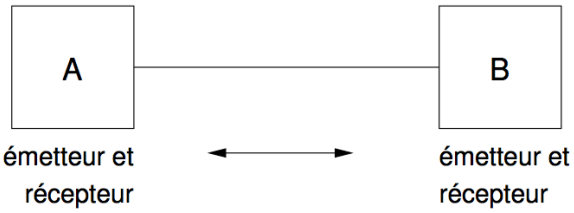
3.1.4 Transmission en half duplex

Dans la transmission en semi-duplex (half-duplex), le canal est exploité à l'alternat pour l'émission : les deux parties émettent tous les deux mais pas en même temps.



3.1.5 Transmission en full duplex

La transmission en full-duplex est bidirectionnelle simultanée. Cela est possible en partageant la bande passante et affecter une partie pour un sens et l'autre pour l'autre sens.



3.2 Signal transmis

Le signal est le véhicule d'information entre deux équipements. Il se propage dans un canal (liaison), matériel ou immatériel sous forme d'onde électromagnétique ou lumineuse. Le signal est une forme ondulatoire résultant de la propagation d'un phénomène vibratoire. Selon la grandeur physique que l'on fait varier, trois types d'ondes sont utilisées :

– ondes électriques (câbles, fils, ...),

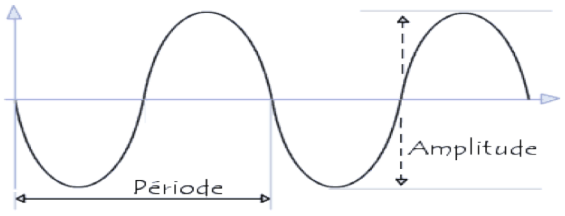
– ondes radio (faisceau hertzien, satellite),

– ondes lumineuses (fibres optiques, infrarouge).

Dans le cas le plus simple une onde est exprimée par une sinusoïde :

$$y(t) = Asin(2\pi ft + \varphi)\,;$$

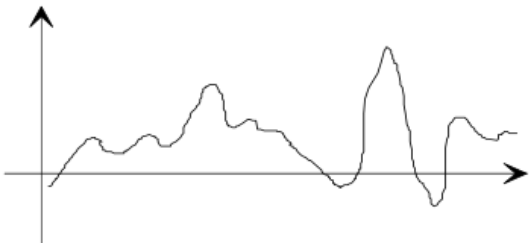
Où A est l'amplitude, f la fréquence et φ la phase.



Les signaux peuvent être de forme analogique ou numérique, les signaux analogiques sont utilisés généralement pour les longues distances, et les signaux numériques pour les courtes distances.

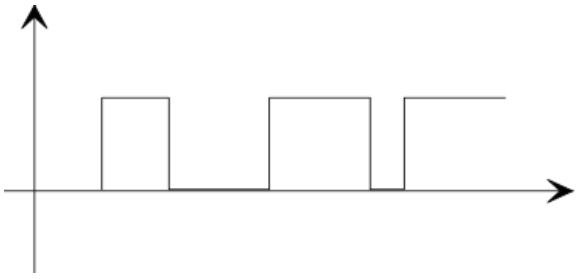
3.2.1 Signal analogique

Un signal analogique est caractérisé par une variation continue, les niveaux de valeurs sont proportionnels aux valeurs de l'information (son, image).



3.2.2 Signal numérique

Le signal numérique est caractérisé par une forme carrée, une variation discontinue et un faible nombre de niveaux de valeurs fixés.



3.3 Caractéristiques d'une ligne de communication

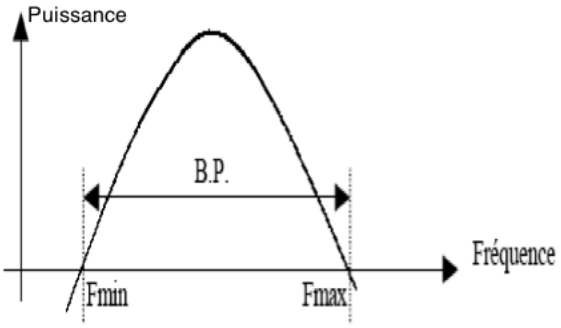
Certaines caractéristiques physiques des supports perturbent la transmission. La connaissance de ces caractéristiques (la bande passante, la sensibilité aux bruits, les limites des débits possibles) est donc nécessaire pour fabriquer de bons signaux, c'est-à-dire les mieux adaptés aux supports utilisés.

3.3.1 La bande passante

La bande passante d'une voie est la plage de fréquence sur laquelle la voie est capable de transmettre des signaux sans que leur affaiblissement soit trop important. Elle est définie par :

$$W = f_{max} - f_{min}$$

Où f_{min} est la fréquence transmise la plus basse et f_{max} la plus haute.



Lorsqu'on parle de la bande passante, on indique une largeur d'intervalle sans préciser les bornes de cet intervalle. Par exemple, la largeur de bande de la ligne téléphonique est 3100Hz.

3.3.2 Rapidité de modulation

La rapidité de modulation R, exprimée en *bauds*, indique le nombre de symboles transmis par unité de temps. Si Δ représente la durée (en secondes) de l'intervalle de temps séparant deux valeurs significatives du signal, alors :

$$R = \frac{1}{\Delta}\ bauds$$

Pour un support de transmission, la rapidité de modulation maximale dépend de sa bande passante (critère de Nyquist). La rapidité de modulation maximale R_{max} est égale au double de la fréquence la plus élevée disponible sur le support :

$$R_{max} = 2F_{max}$$

3.3.3 Taux d'erreur

Il représente la probabilité de perte ou d'altération d'une information (1 bit). On peut la mesurer en calculant pendant un temps significatif le rapport du nombre de bits erronés sur le nombre de bits émis.

3.3.4 Débit binaire

Le débit binaire D est le nombre de bit transmis par unité de temps. Par exemple 512 Kbits/s ou 1 Gigabit/s.

La relation liant la rapidité de modulation au débit binaire est exprimé par la formule :

$$D = R \times log_2(V)$$

Où V désigne la *valence* du signal représentant le nombre des états significatifs que peut prendre le signal.

Une valence de valeur V permet le transports de $P(bits) = log_2(V)$ à chaque baud. Par exemple, pour des modulations simples (des signaux de valence 2) chaque intervalle Δ transporte 1 bit. Les valeurs numériques du débit binaire et de la rapidité de modulation sont alors égales ($R = D$).

Exercice : Si la durée d'un bit est 20ms, quel est le débit binaire ?

3.3.5 Délai de propagation T_p

C'est le temps nécessaire à un signal pour parcourir un support d'un point à un autre. Ce temps dépend de la nature du support, de la distance, de la fréquence du signal,...etc.

3.4 Supports de transmission

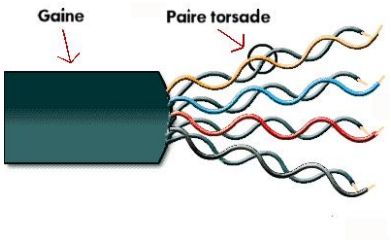
Les supports de transmission sont nombreux et se divisent en deux familles : les supports à guide physique et les supports sans guide physique. Les supports à guide physique, comme les paires torsadées et les câbles coaxiaux, sont les plus anciens, les plus largement utilisés et servent à transmettre des courants électriques. Les supports de verre ou de plastique, comme les fibres optiques, transmettent de la lumière, tandis que les supports sans guide physique des communications sans fil transmettent des ondes électromagnétiques et sont en plein essor.

3.4.1 Supports à guide physique

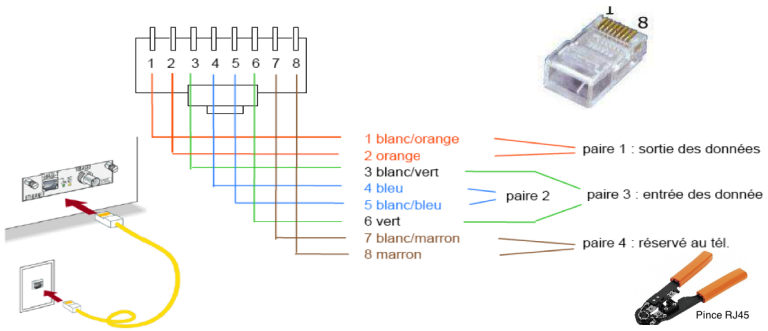
C'est des supports qui utilisent les câbles de différents types pour transmettre l'information.

3.4.1.1 Paires torsadées

La paire torsadée ou enroulée (twisted) est constituée de deux conducteurs identiques torsadés. L'enroulement réduit les conséquences des parasites provenant de l'environnement. L'utilisation la plus courante de la paire torsadée est le raccordement des usagers au central téléphonique (norme RJ11 : Registered Jack). Les réseaux locaux informatiques, où les distances se limitent à quelques kilomètres, utilisent la norme RJ45 utilisant des câbles contenant 4 paires torsadées.



Le raccordement des câbles RJ45 se fait à travers les connecteurs RJ45 permettant de connecter les fils selon le schéma suivant :



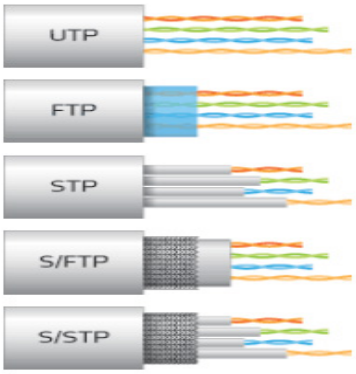
La fabrication manuelle des câbles RJ45 se fait par un pince spéciale appelé "pince RJ45".

Le principal inconvénient des paires torsadées est l'affaiblissement des courants transmis. Elles utilisent souvent, à intervalles réguliers, des éléments appelés répéteurs qui régénèrent les signaux transmis.

Pour les réseaux locaux d'entreprise, la paire torsadée peut suffire. Ses avantages sont nombreux : technique maitrisée, facilité de connexion et d'ajout de nouveaux équipements,

faible coût ainsi qu'elle peut être utilisée en point à point ou en diffusion. Il existe, généralement trois types de câbles :

- UTP (Unshielded Twisted Pairs) : câble à paires torsadées non blindées et non écrantées. Parfois utilisé pour la téléphonie, pas recommandé pour l'informatique.
- FTP (Foiled Twisted Pairs) : paires torsadées entourées dans leur ensemble d'une feuille d'aluminium (écran). C'est le type standard.
- STP (Shielded Twisted Pairs) : paires torsadées entourées chacune par une feuille d'aluminium.
- SFTP (Shielded Foiled Twisted Pairs) et SSTP (Shielded Shielded Twisted Pairs) : câbles FTP ou STP blindés. A utiliser dans les locaux avec fortes perturbations électromagnétiques.

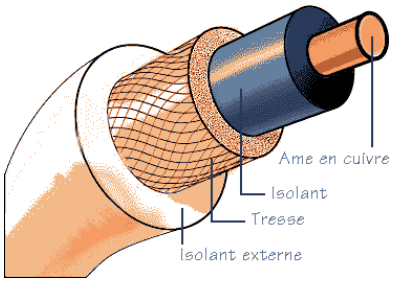


Les câbles à apires torsadées sont normalisés en catégories de Cat1 à Cat7, Les plus utilisées actuellement sont :

- Catégorie 3 : Bande passante 16MHz, utilisée pour la téléphonie.
- Catégorie 5 : Bande passante 100MHz, Débit 100MB/s sur 100m utilisée pour la téléphonie et les réseaux
- Catégorie 6 : Bande passante 250MHz, Débit GB/s sur 100m utilisée pour les réseaux
- Catégorie 6a : Bande passante 500MHz, Débit 10GB/s sur 100m
- Catégorie 7 : Bande passante 600Mhz, Débit 10GB/s

3.4.1.2 Câble coaxial

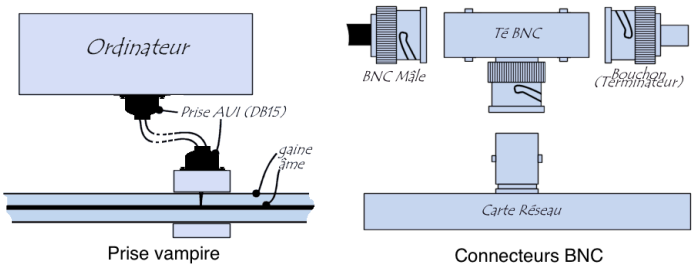
Le câble coaxial est formé de deux conducteurs cylindriques de même axe séparés par un isolant, le tout étant protégé par une gaine plastique.



Il esiste deux types de câble coaxial :

- Le câble 75 Ω, dit "large bande" (broadband) utilisé pour la transmission analogique : c'est le câble de télévision !
- le câble 50 Ω, dit "bande de base" (baseband) généralement utilisé pour transmettre des signaux numériques. Il permet une bande passante de quelques centaines de MHz et des débits allant jusqu'à 2Gbit/s.

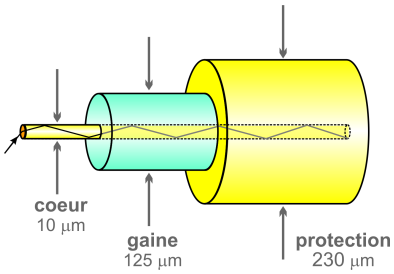
Le câble coaxial est raccordé par des prises vampire pour les gros câbles et les fiches BNC (British Naval Connector) pour les câbles fins.



Le câble coaxial est d'une qualité de transmission et débits meilleures que les paires torsadées et peut être utilisé en point à point ou en diffusion. Cependant, il est un peu plus cher.

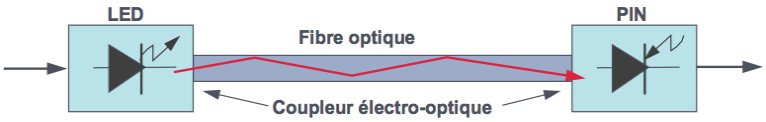
3.4.1.3 Fibre optique

Une fibre optique est constituée d'un fil de verre très fin. Elle comprend un cœur, dans lequel se propage la lumière émise par une diode électroluminescente ou une source laser, et une gaine optique dont l'indice de réfraction garantit que le signal lumineux reste dans la fibre.



Un système de transmission par fibre optique met en œuvre :

- un émetteur de lumière (transmetteur), constitué d'une diode électroluminescente (LED, Light Emitting Diode) ou d'une diode LASER (Light Amplification by Stimulated Emission of Radiation), qui transforme les impulsions électriques en impulsions lumineuses ;
- un récepteur de lumière, constitué d'une photodiode de type PIN (Positive Intrinsic Negative) qui traduit les impulsions lumineuses en signaux électriques ;
- une fibre optique.

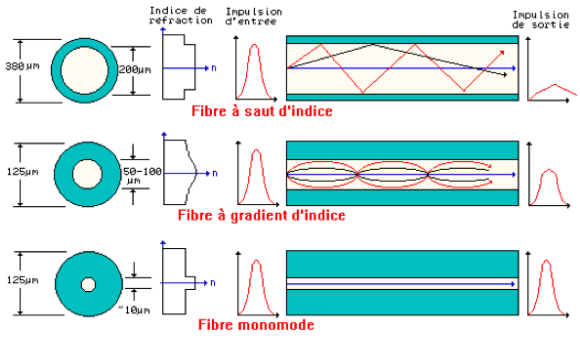


La fibre étant un système de transmission unidirectionnel, une liaison optique nécessite l'utilisation de 2 fibres.

Il existe trois types de fibre optique.

1. Fibre multimode à saut d'indice : le cœur d'indice de réfraction n_1 est entouré d'une gaine d'indice n_2. La variation d'indice entre le cœur et la gaine est brutale (saut d'indice). La propagation s'y fait par réflexion totale à l'interface cœur/gaine. Le diamètre du cœur est important ce qui lui permet d'admettre plusieurs rayons qui se propagent sur des chemins différents ou modes de propagation. La porté des rayons étant de 10 km.
2. Fibre multimode à gradient d'indice : dans ce type, l'indice du cœur décroit de façon continue, depuis le centre du cœur jusqu'à l'interface cœur/gaine suivant une loi parabolique. Tous les rayons sont focalisés au centre de la fibre, ils ont une trajectoire proche de la sinusoïde. La dispersion étant réduite ce qui autorise des portées d'environ 50 km.
3. Fibre monomode : le diamètre du cœur est réduit à 8 μm. Cette réduction, peut être telle que, pour une longueur d'onde donnée, la fibre n'admette plus qu'un seul rayon.

La fibre est alors dite monomode et la distance franchissable est de l'ordre de 100 km.



Le raccordement de la fibre optique utilise des connecteurs de types SC (Subscriber Connector), ST (Straight Tip), FC (Fiber Connector), LC (Lucent Connector).

Malgré que la fibre optique ne permet que les connexion en point à point, ses avantages sont nombreux :

- Débits allant jusqu'à 50 GBit/s (débit théorique 50 TBit/s),
- Transmission simultanée de très nombreux canaux de télévision, de téléphone,...
- Insensible aux parasites électromagnétiques,
- Diamètre extérieure est de l'ordre de 0,1 mm,
- Poids de quelques grammes au kilomètre.
- Difficile à pirater.

3.4.2 Supports sans guide physique

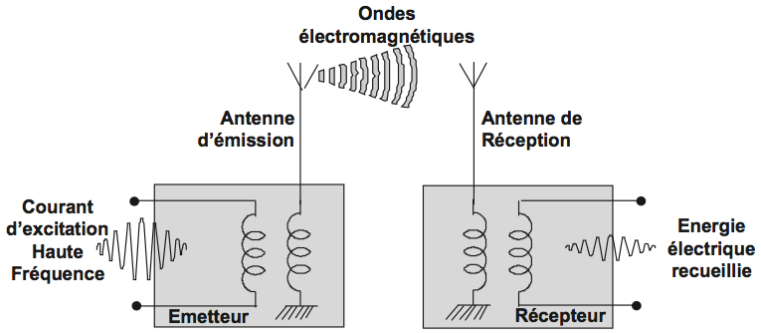
Les supports sans guide physique transmettent des ondes électromagnétiques ou la lumière.

3.4.2.1 Ondes électromagnétiques

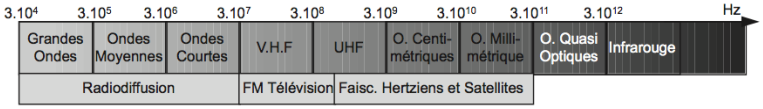
Les ondes électromagnétiques se propagent dans l'atmosphère. L'absence de support matériel apporte une certaine souplesse et convient aux applications comme la téléphonie ou les télécommunications mobiles, sans nécessiter la pose coûteuse de câbles.

Une antenne d'émission rayonne une énergie (onde électromagnétique). Cette énergie

électromagnétique recueillie par un autre conducteur distant ou antenne de réception est transformée en un courant électrique similaire à celui d'excitation de l'antenne d'émission.



Chaque type de liaison ou d'application utilise des bandes de fréquences différentes. L'espace de fréquences utilisables est limité et géré par des organismes nationaux et internationaux. La figure suivante décrit l'utilisation des différents plages de fréquences.



Les hautes fréquences (faisceaux hertziens) sont utilisées pour franchir de grandes distances tandis que les basses (ondes radioélectriques) pour atteindre des récepteurs géographiquement dispersés.

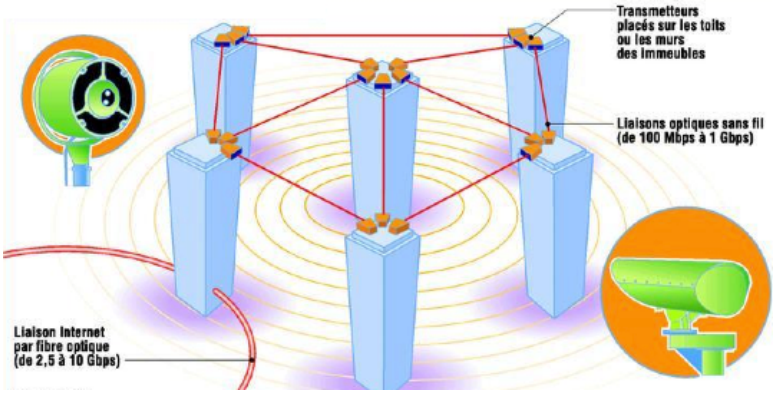
3.4.2.1.1 Faisceaux hertziens Les faisceaux hertziens reposent sur l'utilisation de fréquences très élevées (de 2 GHz à 15 GHz et jusqu'à 40 GHz) et de faisceaux directifs produits par des antennes directionnelles qui émettent dans une direction donnée. La propagation des ondes est limitée à l'horizon optique ; la transmission se fait entre des stations placées en hauteur, par exemple sur une tour ou au sommet d'une colline, pour éviter les obstacles dus aux constructions environnantes. Les faisceaux hertziens s'utilisent pour la transmission par satellite, pour celle des chaînes de télévision ou pour constituer des artères de transmission longue distance dans les réseaux téléphoniques.

3.4.2.1.2 Ondes radioélectriques Les ondes radioélectriques correspondent à des fréquences comprises entre 10 kHz et 2 GHz. Un émetteur diffuse ces ondes captées par des récepteurs dispersés géographiquement. Contrairement aux faisceaux hertziens, il n'est pas nécessaire d'avoir une visibilité directe entre émetteur et récepteur, car celui-ci utilise l'ensemble des ondes réfléchies et diffractées. En revanche, la qualité de la transmission est

moindre car les interférences sont nombreuses et la puissance d'émission beaucoup plus faible.

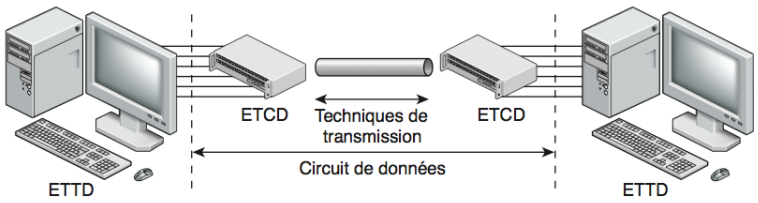
3.4.2.2 Ondes lumineuses

Les liaisons infrarouges et lasers constituent un cas particulier des liaisons hertziennes. Elles sont généralement utilisées, pour interconnecter deux réseaux privés, sur de courtes distances, de l'ordre de quelques centaines de mètres. Elle utilisent des technologies comparables à celles des fibres optiques, mais au lieu d'emprunter un canal en verre, les données prennent la voie des airs à un très hauts débits pouvant dépasse 1 GBit/s. Le signal est numérisé et transmis par un rayon infrarouge ou Laser dans une ligne de visée précise. Le plus souvent, ces liaisons s'effectuent entre des transmetteurs installés au sommets d'immeubles et communiquant en point à point par le biais de faisceaux.



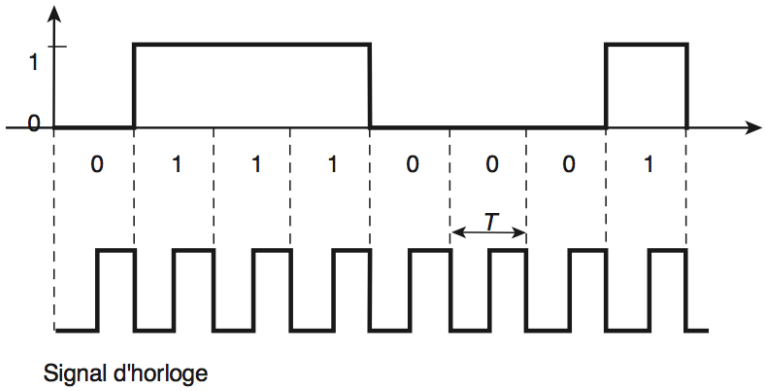
3.5 Codage de l'information

Pour transmettre les données, un équipement spécifique est placé à chaque extrémiste du support : soit un modem (modulateur-démodulateur), soit un codec (codeur-décodeur). Cet équipement assure la fabrication des signaux en émission et leur récupération en réception. Pour émettre les données, le modem reçoit la suite de données binaires à transmettre et fournit un signal dont les caractéristiques sont adaptées au support de transmission. Inversement, en réception, le modem extrait la suite des données binaires du signal reçu. Le support de transmission est ainsi transparent à l'utilisateur. Le support de transmission et les deux modems placés à chacune de ses extrémités constituent un ensemble appelé circuit de données.



L'ISO et l'ITU (Union International des Télécommunications) ont attribué des appellations génériques normalisées aux différents éléments de ce système. Ainsi, le modem et le codec s'appellent des ETCD (équipement de terminaison du circuit de données) et l'ordinateur s'appelle ETTD (équipement terminal de traitement des données).

L'ETTD émetteur fournit à l'ETCD, régulièrement dans le temps, les données à transmettre. L'ETCD les émet sous forme d'un signal à deux valeurs (correspondant à 0 et 1), appelé message de données synchrone.



Les intervalles de temps alloués à chaque symbole sont égaux et coïncident avec les périodes successives d'une base de temps (ou horloge) indispensable à l'interprétation du message de données.

Si la distance entre les deux ETCDs le permet, le signal numérique est transmis directement, la transmission est appelée dans ce cas *bande de base* c'est-à-dire dans la même bande du signal original. Dans le cas contraire, le signal est modulé et la transmission est appelée en *large bande* ou en *bande transposée*.

3.5.1 Transmission numérique (en bande de base)

Lorsque la longueur de la liaison ne dépasse pas quelques centaines de mètres, les informations peuvent être transmises sur le support de liaison sans transformation du signal numérique en signal analogique.

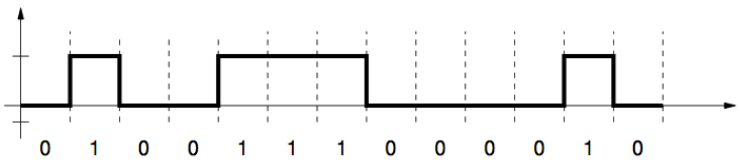
La transmission en bande de base rencontrée principalement dans les réseaux locaux

permet d'obtenir des circuits de données à grand débit et faible portée (débits supérieurs à 1 Mbit/s pour des distances inférieures à 1 Km) en utilisant directement des supports physiques de types métallique (paires torsadées ou câble coaxiaux) ou optique avec éventuellement l'adjonction de répéteurs disposés sur des intervalles allant de 500 mètres à quelques kilomètres.

La transmission de longues suites de 0 ou de 1 (silences) peut rendre difficile la la récupération de l'horloge causer, par conséquent, la perte de la synchronisation entre l'émetteur et le récepteur. Plusieurs types de codage sont utilisés pour introduire des changements d'état fréquents sur le signal pour éviter les silences.

3.5.1.1 Codage unipolaire

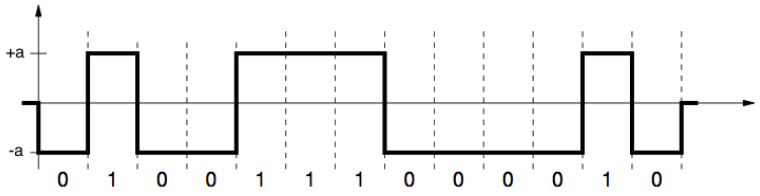
Le signal est transmis sans le moindre changement.



Le problème de ce codage est qu'il ne permet pas de distinguer le cas de 0 du cas d'absence d'information.

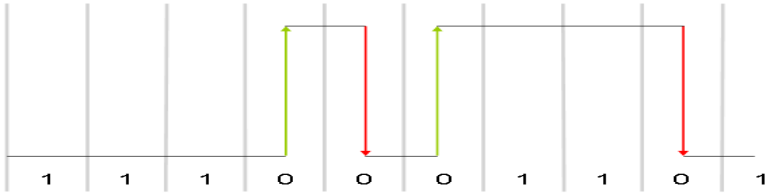
3.5.1.2 Codage NRZ

Pour éviter la valeur nulle, le codage NRZ(No return to zero) utilise une valeur $+a$ du signal pour représenter un 1 et $-a$ pour un 0.



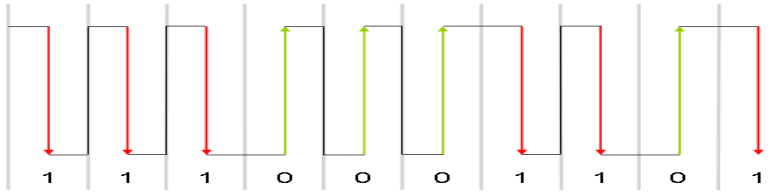
3.5.1.3 Codage NRZI

Le code NRZI (No Return to Zero Inverted) présente les mêmes caractéristiques mais pour éviter les successions de 0, le signal reste dans le même état pour coder un 1 et change d'état pour coder un 0.



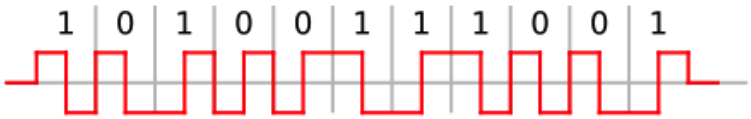
3.5.1.4 Code biphase ou code Manchester

Une opération XOR (OU exclusif) est réalisée entre l'horloge et les données, d'où une transition systématique au milieu de chaque bit du signal binaire.



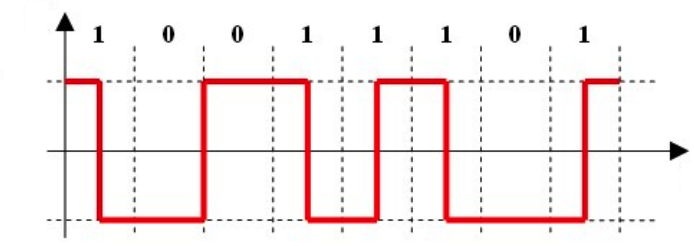
3.5.1.5 Code Manchester différentiel

Une transition systématique est réalisée au milieu de chaque bit. Pas de transition pour coder un bit à 1, une transition pour coder un bit à 0



3.5.1.6 Code Miller

Une transition au milieu du bit pour un 1, pas de transition en milieu de bit pour un 0. Une transition à la fin du bit pour un 0 si le bit suivant est aussi un 0.



3.5.2 Modulation (large bande)

Les techniques en bande de base ne sont pas fiables dès que la distance dépasse quelques centaines de mètres. Pour avoir un signal que l'on puisse récupérer correctement, il faut lui

donner une forme spéciale (sinusoïdale) en le modulant. La transmission par modulation consiste à envoyer une onde sinusoïdale appelée porteuse. Le fait de n'avoir plus de fronts montants ni descendants protège beaucoup mieux le signal des dégradations occasionnées par la distance parcourue par le signal dans le câble puisque le signal est continu et non plus discret. Les opérations de modulation en émission et de démodulation en réception sont réalisées par l'ETCD couramment appelé modem (modulateur-démodulateur).

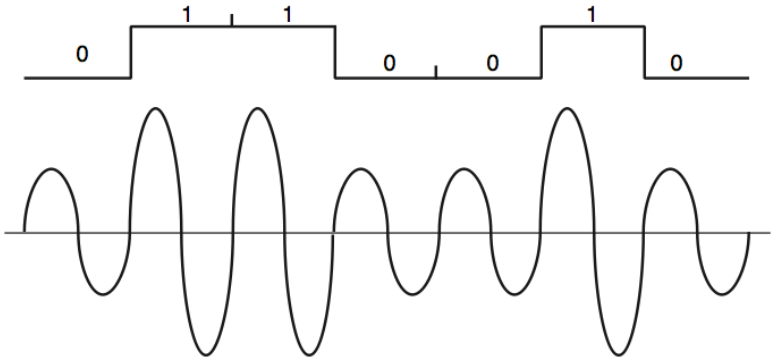
En fonction de la donnée à transmettre, le modem modifie l'un des paramètres de la porteuse (fréquence, phase ou amplitude). On distingue les trois grandes catégories de modulation suivantes :

- modulation d'amplitude, ou ASK (Amplitude-Shift Keying) ;
- modulation de phase, ou PSK (Phase-Shift Keying) ;
- modulation de fréquence, ou FSK (Frequency Shift Keying).

On utilise, souvent, des modulations combinées des trois types précédents.

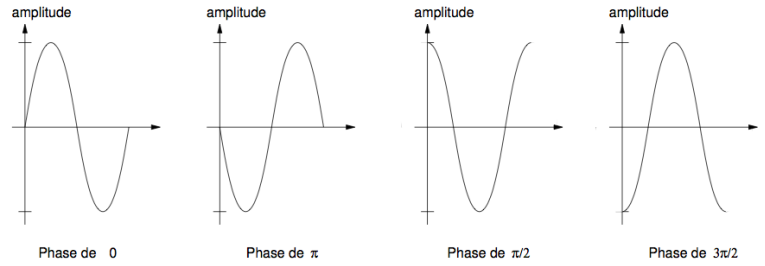
3.5.2.1 ASK (Amplitude-Shift Keying)

Dans la modulation d'amplitude, la distinction entre le 0 et le 1 est obtenue par une différence d'amplitude du signal.

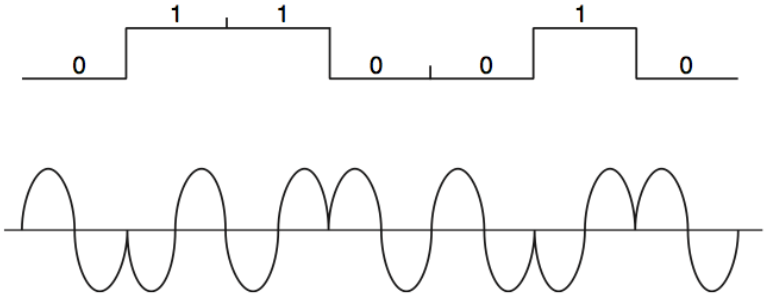


3.5.2.2 PSK (Phase-Shift Keying)

Pour la modulation de phase, la distinction entre 0 et 1 est effectuée par un signal qui commence à des emplacements différents de la sinusoïde, appelés phases.

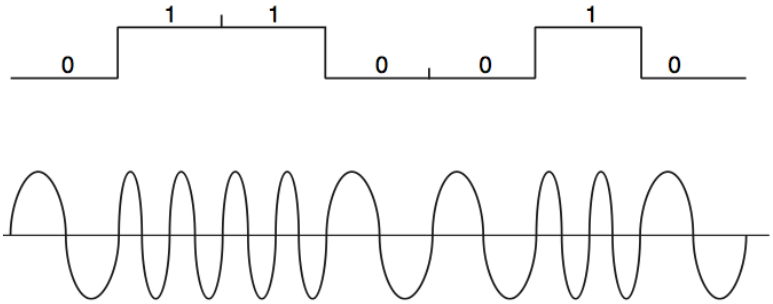


A la figure suivante, les valeurs 0 et 1 sont représentées par des phases respectives de 0 et de π.



3.5.2.3 FSK (Frequency Shift Keying)

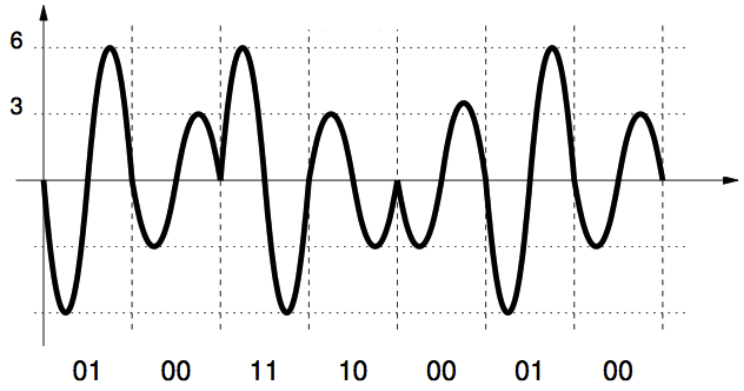
En modulation de fréquence, l'émetteur a la possibilité de modifier la fréquence d'envoi des signaux suivant que l'élément binaire à émettre est 0 ou 1.



3.5.2.4 Modulation de phase et d'amplitude (PSK + AM)

Pour obtenir des vitesses de transmission encore plus élevées dans une modulation de type PSK, il est nécessaire de multiplier le nombre d'états de phase (couramment 4, 8, 16 états ou plus). En combinant une modulation de phase à une modulation d'amplitude, on obtient une meilleure répartition des points sur le diagramme spatial et donc une meilleure immunité au bruit. Par exemple, dans la figure suivante, on combine 2 phases et 2 amplitudes :

– 00 : phase de π et amplitude de 3
– 01 : phase de π et amplitude de 6
– 10 : phase de 0 et amplitude de 3
– 11 : phase de 0 et amplitude de 6



3.5.2.5 Transmission ADSL

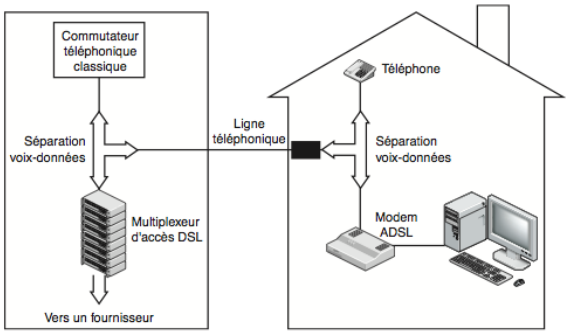
La capacité des lignes téléphoniques en paires torsadées est limitée d'une part par la bande passante, et d'autre part, par le rapport signal/bruit. Toutefois, sur des distances limitées à quelques kilomètres, en améliorant le rapport signal/bruit, il est possible de dépasser les débits de quelques dizaines de kbit/s obtenus avec les modulations précédentes.

La technique utilisée dans l'Asymetric Digital Subscriber Line, permet d'atteindre des débits de plusieurs Mbit/s sur des distances inférieures à 5 km. Cette solution est mise en œuvre pour permettre, entre autres, aux abonnés du réseaux téléphonique commuté RTC (fixe) d'accéder à Internet à des débits élevés. Elle permet de plus, d'assurer une communication téléphonique simultanément aux transferts de données.

Compte tenu des objectifs, les débits dans le sens abonné vers réseau (flux montant ou upstream) sont moins élevés que dans le sens réseau vers abonné (flux descendant ou downstream).

Les valeurs typiques de débit sont de 640 kbit/s et 2 Mbit/s respectivement pour les flux montant et descendant, ce qui correspond à des requêtes sur des serveurs ou des bases de données. Pour obtenir de tels débits, la bande des fréquences utilisées sur les paires téléphoniques va de 0 Hz à 1,1 MHz (pour des lignes supportant de telles fréquences sur des distances courtes). La bande de 0 Hz à 4 kHz est réservée aux communications de type voix analogique. La bande de 64 kHz à 1,1 MHz est utilisée pour la transmission des données en deux bandes distinctes, une pour chaque flux.

Le raccordement, au niveau du réseau et de l'abonné, se fait selon le schéma suivant :



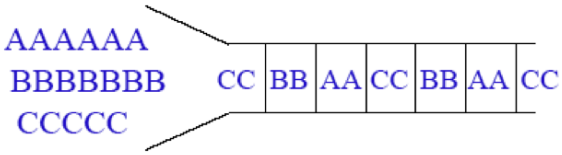
3.6 Multiplexage

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. Chaque émetteur (récepteur) est raccordé à un multiplexeur (démultiplexeur) par une liaison dite voie basse vitesse.

Plusieurs techniques sont possibles :

3.6.1 Multiplexage temporel TDMA (Time Division Multiplexing Access)

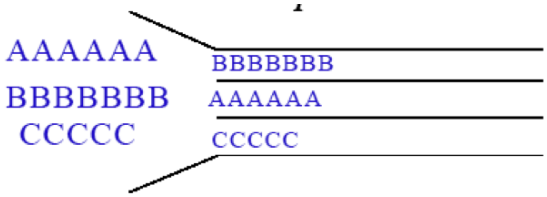
Il partage dans le temps l'utilisation de la voie haute vitesse en l'attribuant successivement aux différentes voies basse vitesse même si celles-ci n'ont rien à émettre. Suivant les techniques chaque intervalle de temps attribué à une voie lui permettra de transmettre 1 ou plusieurs bits.



3.6.2 Multiplexage fréquentiel FDM (Frequency Division Multiplexing)

Il 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 remet 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.



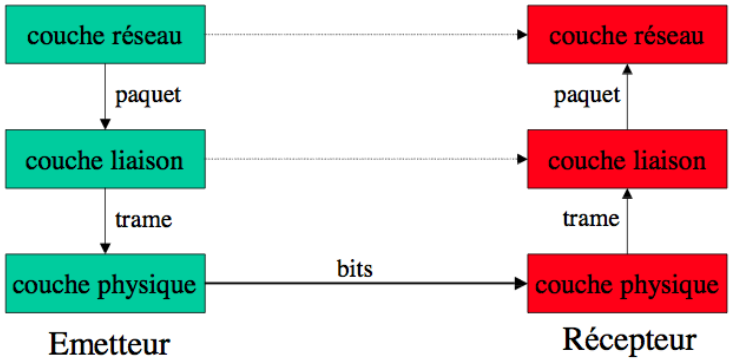
3.6.3 Multiplexage statistique ATDM (Asyncronous Time Division Multiplexing)

Il 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.

Chapitre 4

Couche Liaison de données

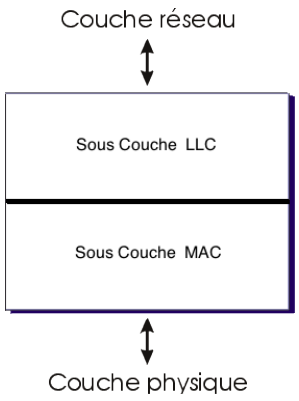
La couche physique permet de transmettre un flot de bits entre deux systèmes distants. La couche liaison (couche n° 2) doit s'assurer de la corrections des bits transmis et les rassembler en paquets pour les passer à la couche Réseaux. La couche liaison récupère des paquets de données de la couche réseau, les enveloppe en des trames qui les envoie une à une à la couche physique.



Pour rendre la transmission fiable, la couche liaison doit assurer :

- La délimitation des blocs de données échangées ;
- Le contrôle de l'intégrité des données reçues ;
- L'organisation et le contrôle de l'échange.
- Le contrôle d'accès à un canal partagé

En fait, la couche liaison se compose de deux sous-couches : LLC (Logical Link Control) et MAC (Media Access Control). La sous-couche LLC Assure les trois premières fonctions tandis que la sous-couche MAC assure la dernière.



4.1 Délimitation de trames

A l'instar des transmissions asynchrones où les bits de start et de stop encadrent les bits d'information, en transmission synchrone une information spéciale permet de repérer le début et la fin des données transmises. Il existe deux méthodes :

4.1.1 Comptage des caractères

On utilise un champ dans l'entête de la trame pour indiquer le nombre de caractères de la trame.

06	'S'	'U'	'P'	'E'	'R'	03	'L'	'E'	06	'C'	'O'	'U'	'R'	'S'

Un problème sérieux peut se poser avec cette méthode si la valeur du champ ajouté est modifiée au cours de la transmission. Généralement, cette méthode est rarement utilisée seule.

4.1.2 Utilisation des fanions

La trame est délimité par une séquence particulière de bits appelée **fanion** ou flag.

Fanion	DONNEES	Fanion

Pour garantir le bon fonctionnement de cette méthode, des bits de transparence sont nécessaires pour qu'une séquence binaire dans la trame ne corresponde accidentellement au fanion. Par exemple, si le fanion est l'octet 01111110, un bit de transparence "0" est inséré après toute séquence de cinq 1 successifs dans la trame.

Exemple :

Données :

01011001111110

Trame :

01111110 010110011111010 01111110

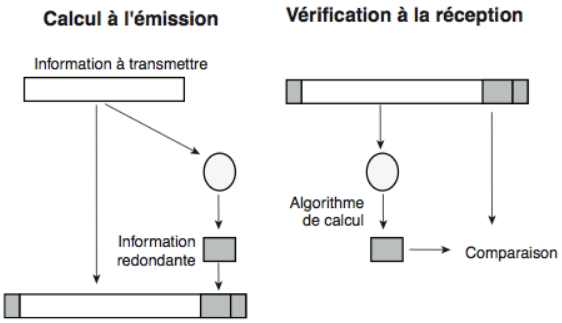
Cette technique est utilisée également en considérant des caractères de délimitation et des caractères de transparence.

L'avantages des fanions est qu'ils permettent de trouver toujours la synchronisation et d'envoyer des trames de tailles quelconques.

4.2 Détection et Correction d'erreurs

D'une manière générale on doit, lors d'une transmission de données, s'assurer que les données reçues n'ont pas été altérées durant la transmission. Plusieurs facteurs peuvent modifier le contenu des données tel que les interférences causées par des rayonnements électromagnétiques ou la distorsion des câbles de transmissions.

Les méthodes de protection exploitent la redondance de données en ajoutant des bits de contrôle aux bits de données. Les bits de contrôle sont calculés, au niveau de l'émetteur, par un algorithme spécifié dans le protocole à partir du bloc de données. À la réception, on exécute le même algorithme pour vérifier si la redondance est cohérente. Si c'est le cas, on considère qu'il n'y a pas d'erreur de transmission et l'information reçue est traitée ; sinon, on est certain que l'information est invalide.



Plusieurs méthodes de protection contre les erreurs peuvent être utilisées :

4.2.1 Duplication de données

Un exemple des bits de contrôle est la duplication des bits transmis (détection par répétition). Le message code est un double exemplaire du message initial, le récepteur sait qu'il y a eu erreur si les exemplaires ne sont pas identiques, il demande alors la retransmission du message. Si la même erreur se passe sur les deux exemplaires, l'erreur ne sera pas détectée.

Si on envoie le message en trois exemplaires, le récepteur pourra même corriger l'erreur en prenant les valeurs des deux copies identiques sans demander la retransmission de l'émetteur.

4.2.2 Code de contrôle de parité

C'est un code dans lequel un bit (le bit de parité) est ajouté au mot initial pour assurer la parité. Son rendement est faible lorsque k est petit.

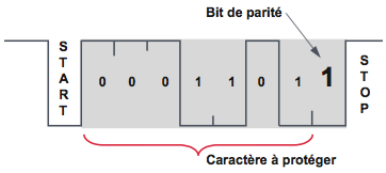
Exemple : Transmission de caractères utilisant un code de représentation (le code ASCII sur 7 bits).

Lettre	Code ASCII	Mot codé (parité paire)	Mode codé (parité impaire)
E	1010001	10100011	10100010
V	0110101	01101010	01101011
A	1000001	10000010	10000011

Ce code est capable de détecter toutes les erreurs en nombre impair mais il ne détecte pas les erreurs en nombre pair. Il permet de détecter une erreur de parité, mais ne permet pas de la localiser.

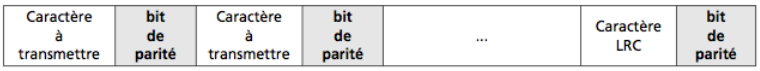
4.2.2.1 Parité verticale

À chaque caractère on rajoute un bit (bit de redondance verticale ou bit de parité, VRC :Vertical Redondancy Check)



4.2.2.2 Parité longitudinale

A chaque bloc de caractère, on ajoute un champ de contrôle supplémentaire (LRC : Longitudinal Redondancy Check)



La parité longitudinale était initialement utilisée pour les bandes magnétiques pour compléter la détection des erreurs de parité verticale.

4.2.2.3 Parité longitudinale et verticale

Le bloc de données est disposé sous une forme matricielle ($k = a \bullet b$). On applique la parité sur chaque ligne et chaque colonne. On obtient une matrice $(a + 1, b + 1)$. Un caractère le LRC est ajouté au bloc transmis. Chaque bit du caractère LRC correspond à la parité des bits de chaque caractère de même rang : le premier bit du LRC est la parité de tous les 1^{er} bits de chaque caractère, le second de tous les 2^e bits... Le caractère ainsi constitué est ajouté au message. Le LRC est lui-même protégé par un bit de parité (VRC).

	H	E	L	L	O	*LRC* →
bit 0	0	1	0	0	1	0
bit 1	0	0	0	0	1	1
bit 2	0	1	1	1	1	0
bit 3	1	0	1	1	1	0
bit 4	0	0	0	0	0	0
bit 5	0	0	0	0	0	0
bit 6	1	1	1	1	1	1
VRC ↓	0	1	1	1	1	0

1001000	0	1000101	1	1001100	1	1001100	1	1001111	1	1000010	0
H		E		L		L		O		LRC	

4.2.3 Codes polynomiaux

La méthode des codes polynômiaux (ou le CRC : Cyclic Redondant Coding) est la méthode la plus utilisée pour détecter des erreurs groupées. Avant la transmission, on ajoute des bits de contrôle. Si des erreurs sont détectées à la réception, il faut retransmettre le message.

Dans ce code, une information de n bits est considérée comme la liste des coefficients binaires d'un polynôme de n termes, donc de degré $n - 1$.

Exemple :

- $1101 \rightarrow x^3 + x^2 + 1$
- $110001 \rightarrow x^5 + x^4 + 1$

– $11001011 \rightarrow x^7 + x^6 + x^3 + x + 1$

Pour calculer les bits de contrôle, on effectue un certain nombre d'opérations avec ces polynômes à coefficients binaires. Toutes ces opérations sont effectuées modulo 2. C'est ainsi que, dans les additions et dans les soustractions, on ne tient pas compte de la retenue : Toute addition et toute soustraction sont donc identiques à une opération XOR. Par exemple :

	1	0	0	1	1	0	1	1
+	1	1	0	0	1	0	1	0
=	0	1	0	1	0	0	0	1

	1	1	1	1	0	0	0	0
-	1	0	1	0	0	1	1	0
=	0	1	0	1	0	1	1	0

La source et la destination choisissent un même polynôme $G(x)$ dit générateur car il est utilisé pour générer les bits de contrôle (Checksum).

L'algorithme pour calculer le message à envoyer est le suivant. Soit $M(x)$ le polynôme correspondant au message original, et soit r le degré du polynôme générateur $G(x)$ choisi :

– Multiplier $M(x)$ par x^r, ce qui revient à ajouter r zéros à la fin du message original

– Effectuer la division suivante modulo 2 :

$$\frac{M(x)x^r}{G(x)} = Q(x) + R(x)$$

– Le quotient $Q(x)$ est ignoré. Le reste $R(x)$ (Checksum) contient r bits (degré du reste $r-1$). On effectue alors la soustraction modulo 2 :

$$M(x).x^r - R(x) = T(x)$$

Le polynôme $T(x)$ est le polynôme cyclique : c'est le message prêt à être envoyé. Le polynôme cyclique est un multiple du polynôme générateur $T(x) = Q(x).G(x)$

A la réception, on effectue la division suivante :

$$\frac{T(x)}{G(x)}$$

– Si le reste = 0, il n'y a pas d'erreur

– Si le reste $\neq 0$, il y a erreur, donc on doit retransmettre

En choisissant judicieusement $G(x)$, on peut détecter toute erreur sur 1 bit, 2 bits consécutifs, une séquence de n bits et au-delà de n bits avec une très grande probabilité.

Exemple

Soit à transmettre le message 1011011 en utilisant le polynôme générateur $G(x) = x^4 + x + 1$. On procède comme suit pour calculer le message à transmettre

1. message original = 1011011 $\Rightarrow M(x) = x^6 + x^4 + x^3 + x^1 + 1$
2. $G(x) = x^4 + x + 1$
3. $M(x).x^4 = x^10 + x^8 + x^7 + x^5 + x^4$
4. Calculer $R(x)$ par division polynomiale

x^{10}	+	x^8	+	x^7	+	x^5	+	x^4	x^4	+	x	+	1
x^{10}	+	x^7	+	x^6					x^6	+	x^4	+	x^2
x^8	+	x^6	+	x^5	+	x^4							
x^8	+	x^5	+	x^4									
x^6													
x^6	+	x^3	+	x^2									
		x^3	+	x^2									

5. $R(x) = x^3 + x^2 = (1100)_2$
6. Le message à envoyer $T(x) = M(x).x^r - R(x) = x^10 + x^8 + x^7 + x^5 + x^4 - x^3 - x^2 = (10110111100)_2$

À la réception, un calcul semblable s'effectue sur le mot reçu, mais il faut, ici, que le reste soit nul. Dans le cas contraire, c'est qu'une erreur est survenue en cours de route.

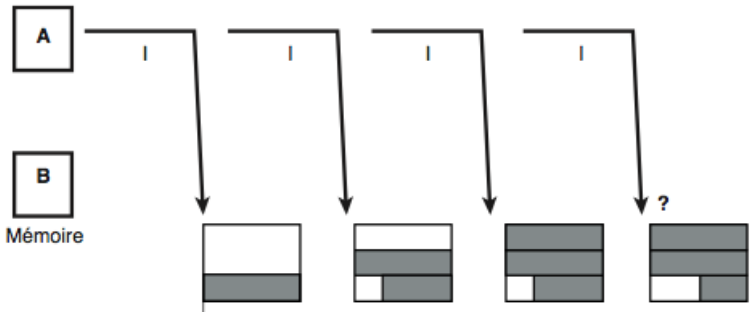
Codes polynomiaux utilisés

Les principaux polynômes générateurs (diviseurs) sont :

- LRCC-8 : $x^8 + 1$
- LRCC-16 : $x^{16} + 1$
- CRC 12 : $x^{12} + x^{11} + x^3 + x^2 + x + 1$
- CRC 16 Forward : $x^{16} + x^{15} + x^2 + 1$
- CRC 16 Backward : $x^{16} + x^{14} + x + 1$
- CRC CITT Forward : $x^{16} + x^{12} + x^5 + 1$
- CRC CITT Backward : $x^{16} + x^{11} + x^4 + 1$

4.3 Contrôle de flux

Dans une transmission d'information d'un émetteur A vers un récepteur B, si l'émetteur produit les données à une vitesse nettement supérieure à la vitesse de consommation du récepteur, ce dernier sera engorgé (saturé ou surchargé) et les informations émises seront perdues. Pour résoudre ce problème, on peut penser à doter le récepteur d'une mémoire tampon lui permettant de stoker les messages en attendant leur traitement. On peut constater rapidement que quelque soit la taille de la mémoire utilisée, elle peut être saturée.



Le contrôle de flux sert à mettre en place un mécanisme de contrôle du rythme d'envoi des informations vers le récepteur. Il peut être réalisé par plusieurs méthodes :

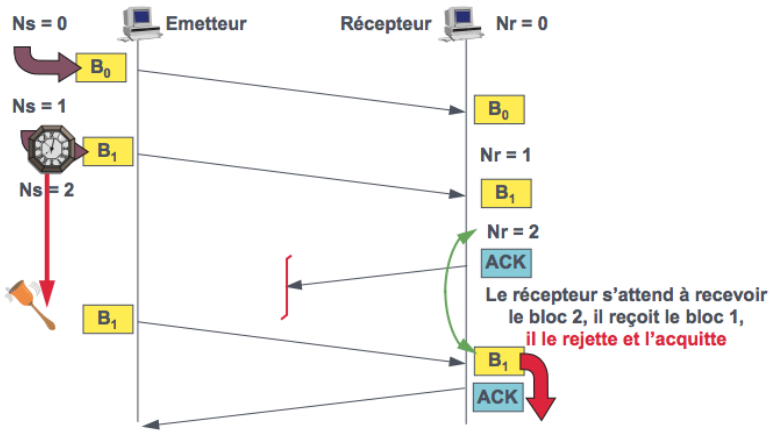
4.3.1 Envoyer et attendre (Sent and Wait)

Après l'envoi d'un bloc d'information, L'émetteur s'arrête dans l'attente d'un accusé de réception. À la réception de l'acquittement, noté ACK pour Acknowledge, l'émetteur envoie le bloc suivant.

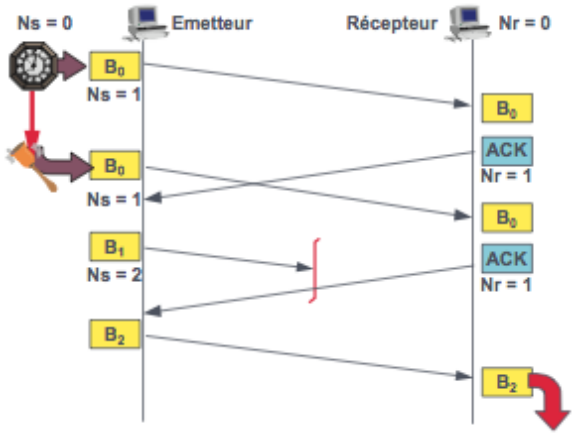
En cas d'erreur de transmission, le bloc reçu est rejeté. Le bloc est dit perdu, il n'est pas acquitté. L'émetteur reste alors en attente. Pour éviter un blocage de la transmission, à l'émission de chaque bloc de données, l'émetteur arme un temporisateur (Timer). À l'échéance du temps imparti (Time Out), si aucun accusé de réception (ACK) n'a été reçu, l'émetteur retransmet le bloc non acquitté.

Une difficulté survient si la perte concerne l'ACK. En effet, bien que les données aient été correctement reçues, l'émetteur les retransmet sur temporisation. Les informations sont ainsi reçues 2 fois. Pour éviter la duplication des données, il est nécessaire d'identifier les blocs. À cet effet, l'émetteur et le récepteur entretiennent des compteurs Ns (Ns, Numéro émis, s pour send) et Nr (Numéro du bloc à recevoir, r pour receive).Les deux compteurs sont initialisés à zéro. Le contenu du compteur Ns est transmis avec le bloc, le récepteur compare ce numéro avec le contenu de son compteur Nr. Si les deux valeurs sont identiques

le bloc est réputé valide et accepté. Si les valeurs diffèrent, le bloc reçu n'est pas celui attendu. Il est rejeté et acquitté s'il correspond à un bloc déjà reçu.



Dans les cas où les délais de consommation sont plus importants, les données peuvent ne pas être acquittées à temps. Par exemple, si A transmet un bloc B0 et B se tarde dans son traitement, A va retransmettre B0 avant de recevoir l'acquittement. Si A transmet un nouveau bloc B1 et se perd, il va considérer l'acquittement du deuxième B0 comme un acquittement de B1.



Pour éviter cette confusion d'interprétation, il est aussi nécessaire de numéroter les ACK.

Le temps d'attente des acquittements rend la méthode send and wait peu efficace. En plus, il est unidirectionnel.

4.3.2 Fenêtre d'anticipation

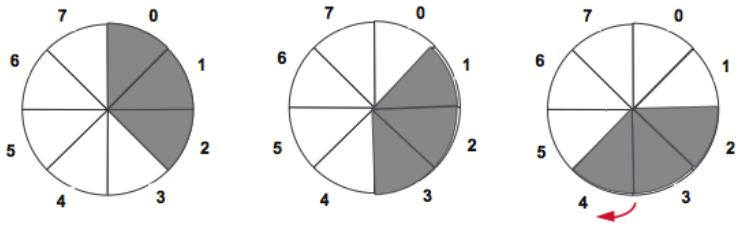
Pour améliorer le protocole précédent et réduire le temps d'attente des acquittements, on émet plusieurs blocs sans attendre les ACK, ce processus se nomme anticipation. Ainsi,

un acquittement n'acquitte plus une seule trame mais un ensemble de trames qui se suivent sans erreur. Le nombre de trames successives qu'on peut émettre sans réception d'acquittement est limité par une valeur notée W, appelée fenêtre (Window). Le principe est d'autoriser l'émetteur à envoyer les trames de numéro de séquence compris entre le numéro r de la prochaine trame attendue (communiqué par le récepteur) et $r + W - 1$:

$$r \leq Ns \leq r + W - 1$$

Remarque : $W = 1$ dans le cas d'une procédure Send-and-Wait.

Pour pouvoir réenvoyer les trames en cas d'erreur, l'émetteur met les blocs non acquittés dans W mémoires tampons. À la réception d'un acquittement d'une trame, l'émetteur libère la mémoire correspondante et envoie une nouvelle trame.



Problème : que se passe-t-il si, de façon temporaire, le récepteur n'est pas prêt à recevoir les W trames d'information de la fenêtre ? L'utilisation d'une fenêtre d'anticipation peut nécessiter la mise en œuvre d'un mécanisme de régulation supplémentaire de type tout-ou-rien. L'idée est d'utiliser une trame de contrôle particulière est utilisée pour indiquer que le récepteur est momentanément dans l'incapacité de continuer à recevoir. L'émetteur recevant cette trame de contrôle cesse immédiatement toute émission (même s'il n'avait pas utilisé toute sa "fenêtre" d'émission). Une autre trame de contrôle est alors nécessaire pour indiquer à l'émetteur que le récepteur est revenu dans un état normal et qu'il est donc prêt à recevoir de nouvelles trames.

4.4 Procédures de gestion des données

Les procédures de gestion de données sont des protocoles de la couche liaison qui mettent en œuvre les techniques précédentes (délimitation des trames, correction des erreurs et contrôle de flux). Elles sont de deux familles :

1. Les procédures orientées caractère : qui fonctionnent généralement à l'alternat (de type send and wait).

2. Les procédures orientées bit : conçues pour les transmissions bidirectionnelles simultanées à hauts débits.

4.4.1 Procédure HDLC

La procédure HDLC (High-level Data Link Control) est une procédure orientée bit, développée par IBM et normalisée par l'UIT en 1976. HDLC est une procédure point à point et multipoints en full duplex, utilisant des trames séparées par des fanions de valeur 01111110 (7E). Trois modes peuvent être exploités par HDLC :

1. le mode de réponse normal (Normal Response Mode ou NRM) : la station secondaire doit attendre un ordre explicite du primaire avant de pouvoir émettre.
2. le mode de réponse asynchrone (Asynchronous Response Mode ou ARM) :la station secondaire a le droit d'émettre des données sans attendre l'invitation du primaire. Ce mode de fonctionnement est également connu sous le nom protocole LAP (Link Access Protocol). Il suppose que les deux stations possèdent à la fois le statut primaire et le statut secondaire.
3. le mode de réponse asynchrone équilibré ou symétrique (Asynchronous Balanced Mode ou ABM) : la liaison est obligatoirement point à point ; comme pour LAP, les deux stations possèdent à la fois le statut primaire et le statut secondaire. Ce mode de fonctionnement est connu aussi sous le nom de protocole LAP-B (Link Access Protocol-Balanced). De nos jours, c'est le seul mode utilisé.

4.4.1.1 Types de trames HDLC

HDLC utilise trois types de trames :

1. les trames diinformation ou trames **I** : assurent le transfert de données ;
2. les trames de supervision ou trames **S** (Supervisor) : assurent la transmission des commandes de supervision (accusé de réception...),
3. les trames non numérotées ou trames **U** (Unnumbered) : supervisent la liaison (connexion, déconnexion).

4.4.1.2 Structure de la trame HDLC

La trame HDLC est organisée comme suit :

8 bits	8 bits	8 bits	n bits	16 bits	8 bits
fanion	adresse	contrôle	information	FCS	fanion

- Fanion (flag) : 01111110
  . délimite les trames : toutes les trames doivent commencer et finir par un fanion.
  . permet la synchronisation des trames : toutes les stations rattachées à la liaison doivent rechercher en permanence cette séquence ;
  . un même fanion peut servir de fanion de fermeture pour une trame et de fanion d'ouverture pour la trame suivante ;
  . mécanisme de transparence au fanion par bits de bourrage : en émission, un 0 est inséré dès que cinq 1 consécutifs apparaissent en dehors des champs F ; ces 0 sont enlevés en réception. Si sept 1 apparaissent n'importe où dans une trame, elle est déclarée en erreur.
- Champ d'adresse : permet d'identifier la trame comme étant une commande ou une réponse. En mode ABM, les valeurs que peut prendre ce champ sont prédéfinies. Quatre valeurs sont suffisantes pour distinguer les commandes et les réponses dans les deux sens de transmission (ex : 11000000, 10000000, 11110000 et 1110000).
- Champ de contrôle : il indique le type de trame avec les paramètres nécessaires.
- FCS (Frame Check Sequence) : calculé sur les champs d'adresse, de commande et d'information, à partir du code polynômial V.41 ($x^{16} + x^{12} + x^5 + 1$).

4.4.1.3 Champ de contrôle et formats de trame

Il existe trois formats de trame qui correspondent à des codages différents du champ de contrôle :

	1	2	3	4	5	6	7	8
Information	0	Ns			P/F	Nr		
Supervision	1	0	S	S	P/F	Nr		
Non numérotée	1	1	M	M	P/F	M	M	M

- Ns : numéro de séquence de la trame émise,
- Nr : numéro de la prochaine trame attendue (acquittement dans les données),
- P/F (Poll/Final) : Ce bit est positionné à 1 par le primaire lorsque celui-ci sollicite une réponse immédiate du secondaire.

La signification des trames de type S dépend des deux bits SS selon le tableau suivant :

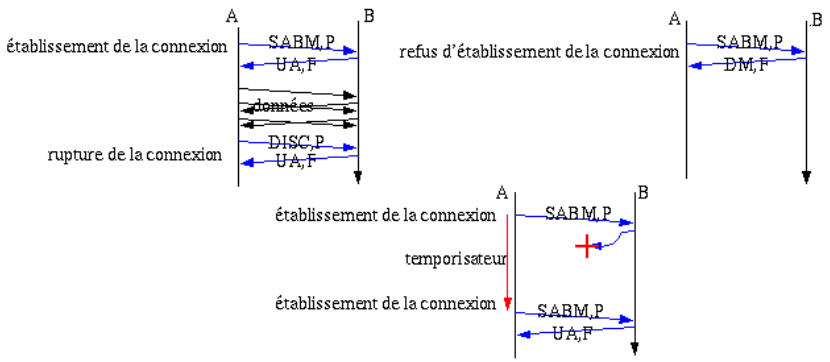
S	S	Commande	Signification
0	0	RR (Receiver Ready)	La station est prête à recevoir la trame numéro Nr et accuse positivement la réception des trames jusqu'à (Nr - 1)
0	1	RNR (Receiver not Ready)	La station n'est pas prête à recevoir des trames mais et accuse positivement la réception des trames jusqu'à (Nr - 1)
1	0	REJ (Reject)	La station rejette les trames à partir du numéro Nr. L'émetteur est obligé de retransmettre (P/F = 1)
1	11	SREJ (Reject)	= REJ mais uniquement pour la trame numéro Nr.

La signification des trames de type U dépend des deux bits M selon le tableau suivant :

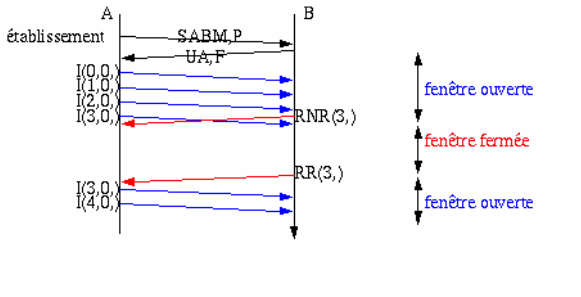
Trame	Commande	Signification
11111100	SABM	Set ABM demande l'établissement en mode ABM
11110000	DM	Disconnect Mode indique que la station se trouve en mode déconnecté
11001010	DISC	Disconnect libère la liaison
11000110	UA	Unnumbered Acknowledge indique la réception et l'acceptation d'une commande non numérotée

4.4.1.4 Exemples des échange

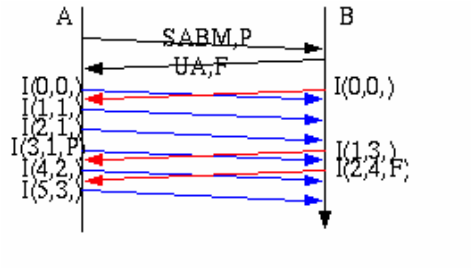
La figure suivante illustre des exemples des échanges entre deux stations utilisant la procédure HDLC pour établir la connexion.



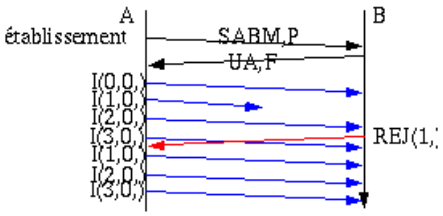
L'exemple suivant, illustre un cas d'échange unidirectionnel.



L'exemple suivant, illustre un cas d'échange bidirectionnel.



L'exemple suivant, illustre un cas d'échange unidirectionnel avec perte de trames.



4.5 Protocole PPP

Le protocole PPP est le protocole de liaison point à point utilisé dans Internet. Il utilise les lignes téléphoniques de l'abonné pour accéder au réseau (la liaison concerne typiquement un ordinateur personnel et le fournisseur d'accès à Internet). Il s'agit d'une version très simplifiée d'HDLC qui ne comprend ni contrôle de flux, ni mécanisme de reprise sur erreurs. La structure d'une trame PPP est donnée dans la figure suivante.

Flag	Address	Control	Data PPP	Flag

Les 8 bits du champ Address sont à 1 (la liaison étant point à point, une seule valeur d'adresse suffit). Le champ Control a la même signification que dans HDLC. Le champ Data PPP commence par deux octets (le champ protocole), qui identifient le protocole

de niveau supérieur auquel est destinée la trame ; il se termine par un champ FCS dont le mode de calcul est identique à celui d'une trame HDLC. La seule trame transportant des données sur une liaison fiable est une trame U de type UI (Unnumbered Information). Cette trame contient un champ d'informations mais n'est pas numérotée (car il n'y a pas de contrôle de flux). L'absence de mécanisme de reprise sur erreur ne signifie pas que le circuit est fiable : le champ FCS sert à valider les trames reçues.

PPP comprend trois composants principaux :

- Une méthode pour encapsuler les datagrammes de plusieurs protocoles.
- Un protocole de contrôle du lien "Link Control Protocol" (LCP) destiné à établir, configurer, et tester la liaison de données.
- Une famille de protocoles de contrôle de réseau "Network Control Protocols" (NCPs) pour l'établissement et la configuration de plusieurs protocoles de la couche réseau.

Deux variantes très connues du protocole PPP :

- **PPPoA** (en anglais point-to-point protocol over ATM) est un protocole utilisé par les connexions haut débit ADSL et câble destinées aux particuliers.
- **PPPoE** (en anglais point-to-point protocol over Ethernet) est un protocole d'encapsulation de PPP sur Ethernet. Il permet de bénéficier des avantages de PPP, notamment sa compatibilité avec les protocoles d'authentification (PAP, CHAP, etc.) et le contrôle de la connexion (débit, etc.), sur un réseau Ethernet. Il est beaucoup employé par les connexions haut débit à Internet par ADSL et câble destinées aux particuliers.

Chapitre 5

Couche Réseaux (2 cours)

Pour pouvoir échanger des informations entre les utilisateurs de plusieurs réseaux locaux, les entités intermédiaires jouent un rôle capital. Elles doivent contenir les moyens nécessaires à l'acheminement des informations entre deux stations quelconques dans le réseaux. Ces moyens sont situés, selon le modèle OSI, au niveau de la couche 3 : la couche réseaux.

La couche réseaux est appelée, donc, à fournir les procédures et les moyens fonctionnels nécessaires à l'échange des informations données par la couche transport. C'est un service de bout en bout qui est responsable de l'acheminement des paquets de données qui peuvent traverser plusieurs nœuds intermédiaires. Le paquet est l'unité de transport de données dans la couche réseaux.

Cette couche est surtout nécessaire dans les réseaux maillés où les paquets doivent traverser plusieurs nœuds avant d'arriver à, leurs destinations ; elle n'est pas prise en compte dans les réseaux constitués par une seule liaison. Les caractéristiques de cette couche ont été déterminées par la réalisation effective de réseaux d'ordinateurs généraux (Internet).

Le rôle de la couche réseau est de transporter d'une extrémité à l'autre du réseau des blocs de données provenant d'une fragmentation des messages du niveau supérieur, le niveau transport.

Pour pouvoir échanger des informations entre deux entités communicantes quelconques à travers un ou plusieurs réseaux, de nombreux services sont fournis par la couche réseau :

1. **Commutation :** pour mettre en relation les deux correspondants.
2. **Adressage et nommage :** pour identifier et localiser chaque correspondant de manière unique sur le réseau.
3. **Routage :** pour acheminer les blocs d'information vers le destinataire.

4. **Segmentation :** pour adapter la taille des unités de données transférées aux capacités du réseau.
5. **Contrôle de congestion :** pour contrôler le trafic admis dans le réseau pour empêcher son effondrement.

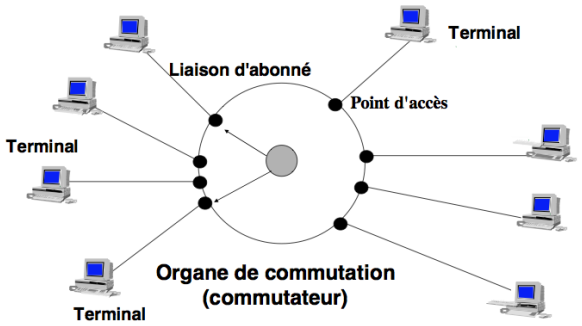
En plus de ces grands services, la couche réseau assure l'intégrité du transport des paquets. Pour ce faire, il est nécessaire de définir un format de paquet et de se donner des moyens pour détecter les erreurs.

Des services facultatifs peuvent être rendus par la couche réseau. Par exemple, la livraison en séquence des paquets reçus par la couche transport, le transfert accéléré de paquets de supervision, le multiplexage de connexions réseau... etc.

La commutation (vue dans le chapitre précédent) est une tâche de base de la couche réseaux, elle peut être de circuit de message ou de paquet. Dans le dernier cas, souvent utilisé, les paquets peuvent être transférés en mode connecté ou non connecté

5.1 Commutation

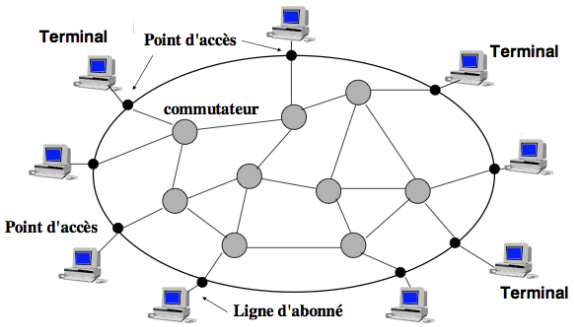
La commutation consiste à mettre en relation de façon temporaire 2 points de connexion, créant ainsi une liaison temporaire entre les 2 terminaux connectés. L'organe de commutation s'appelle un commutateur. Cette solution résout le problème du nombre de lignes d'abonnés (une ligne par abonné). Par contre, si le commutateur est unique, les lignes d'abonnés doivent être très longues, et de plus, à un instant, seuls 2 abonnés peuvent être connectés.



Pour résoudre ce problème, il faut rapprocher les points de connexion des abonnés, offrir la possibilité de construire des chemins multiples entre les points de connexion : il faut créer un maillage entre des commutateurs, c'est à dire un réseau de commutation.

5.1.1 Réseaux de commutation

Les terminaux sont reliés au réseau de commutation par une ligne d'abonné locale (courte). Plusieurs liaisons peuvent être établies simultanément. Plusieurs terminaux peuvent accéder au réseau de commutation par le même point d'accès (non représenté sur le schéma). Les commutateurs réalisent une fonction d'aiguillage et participent au routage des données, pour constituer des connexions entre les terminaux.



Dans ce contexte où la ressource est rare vis-à-vis de la demande potentielle (si simultanément tous les abonnés du réseau désiraient joindre un autre abonné...), il est indispensable de rechercher des techniques particulières pour optimiser le partage des ressources, c'est l'objectif des techniques de commutation.

5.1.2 Techniques de commutation

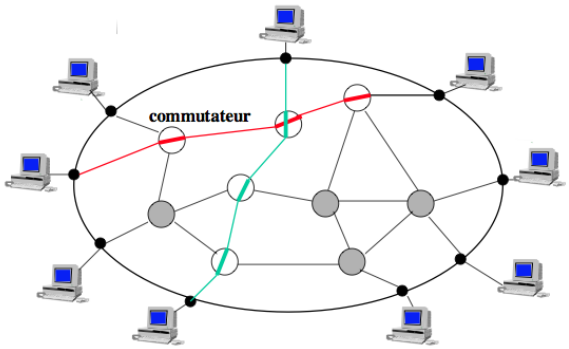
Selon la technique employée pour relier deux utilisateurs, on distingue la commutation de circuits, de messages ou de paquets.

5.1.2.1 La commutation de circuits

Elle est issue des techniques utilisées dans les réseaux téléphoniques (RTC). Elle se déroule en 3 phases :

1. **La connexion :** un chemin est établi entre l'appelant et l'appelé, par commutations successives. Les commutateurs ne remplissent qu'une fonction d'aiguillage. Tout se passe comme s'il n'y avait qu'une seule liaison entre les extrémités.
2. **Le transfert :** Les données (ou la voix) sont transmises de bout en bout sur le "circuit de données". Les ressources (lignes) sont attribuées en permanence, et en volume constant, pendant toute la durée de la communication (les "silences" sont pénalisés).

3. **La libération :** après le transfert, les ressources sont restituées au réseau de commutation, et sont disponibles pour d'autres communications.

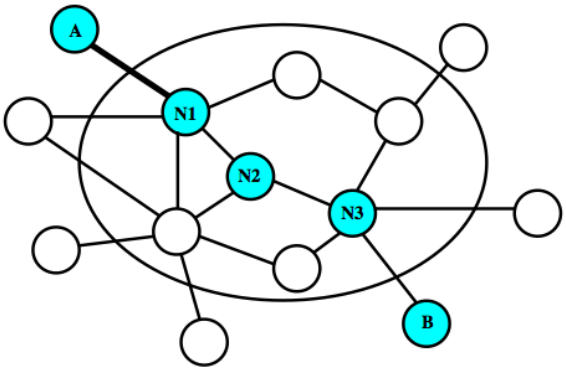


Cette technique soufre de trois inconvénients :

- Chevelure analogique (débit limité)
- Délais de connexion importants
- Signalisation indistincte des données, et mal adaptée

5.1.2.2 La commutation de message

Le principe est très différent de la commutation de circuits. Les données sont structurées en messages. Un message est une suite logique de données qui forment un tout (fichier, enregistrement ...).



Un message est acheminé vers sa destination au fur et à mesure des commutations :

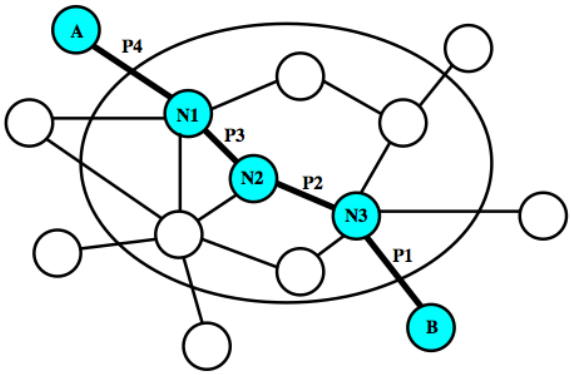
- Le message délivré par A est envoyé vers N1, où il est stocké et vérifié,
- La liaison A-N1 est libérée
- Il est transmis à N2, où il est stocké et vérifié,
- La liaison N1-N2 est libérée,
- Le processus est itératif jusqu'à la livraison du message à B

– Ainsi, les commutateurs (ou nœuds de commutation) sont dotés de mémoire et de capacité de traitement de données (leur fonction va au-delà d'un simple aiguillage).
– Quand un message est enregistré dans un nœud de commutation, la liaison par laquelle il est arrivé est libérée, et ses ressources sont disponibles pour le réseau de commutation.

L'avantage de cette technique est que sur le chemin AB, on n'utilise qu'une seule liaison de données à la fois, les autres étant disponibles pendant ce temps pour d'autres communications. Cependant, si la taille des messages est trop importante, il y a des risques de débordement des tampons de stockage puisqu'un nœud de commutation peut recevoir des messages depuis plusieurs stations). En plus, le délai d'acheminement de bout en bout est excessif (il faut attendre qu'un message soit entièrement stocké dans un nœud avant de le retransmettre vers le nœud suivant),

5.1.2.3 Commutation de paquets

Pour palier aux inconvénients engendrés par les messages de taille trop importante, on fragmente les messages en paquets de taille moyenne. Les paquets sont acheminés de nœud en nœud : après avoir été vérifiés, ils sont réexpédiés, sans attendre la totalité du message.



Cette fragmentation élimine les inconvénients engendrés par la commutation de messages trop longs :

– A débit constant, le délai d'acheminement de bout en bout est plus court,
– La capacité de stockage des nœuds de commutation est moindre,
– A taux d'erreur constant, la probabilité d'erreur sur un paquet (un bit erroné) est plus faible, et en cas d'erreur, seul le paquet erroné doit être retransmis.
– De plus, les paquets peuvent plus facilement être multiplexés sur des liaisons à haut débit.

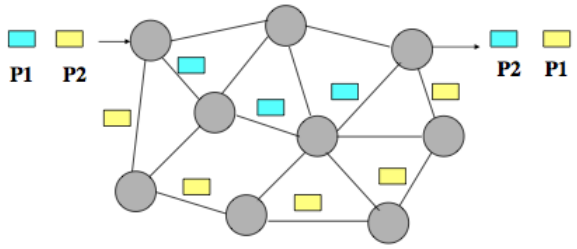
La taille des paquets est fixée par un compromis entre 2 contraintes :

– Il faut réduire leur taille pour profiter au maximum des avantages de la fragmentation,
– Il ne faut pas trop la réduire, à cause de l'encapsulation des données (pour ne pas réduire le débit utile, il faut que la taille des données utiles soit très supérieure à celle de la signalisation et des informations protocolaires).

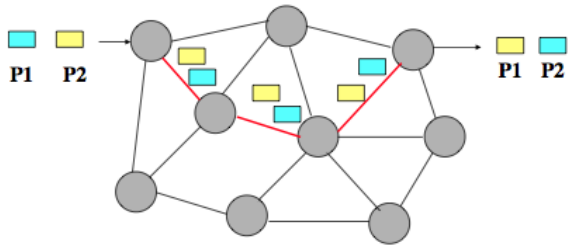
En pratique, elle est fixée par les protocoles de transfert de paquets : Elle est généralement de l'ordre de grandeur du koctet (sauf pour les réseaux à commutations rapides, pour lesquels les messages sont fragmentés en cellules de quelques dizaines d'octets).

La commutation de paquets peut se faire en deux modes :

– Mode non connecté : les paquets (appelés datagrammes) sont transmis de nœud en nœud, au fur et à mesure que la connexion est établie, sans s'assurer que les ressources soient disponibles de bout en bout. Les paquets d'un même message, sont indépendants. Ils peuvent suivre des chemins différents, car les algorithmes de routage sont adaptatifs (chaque paquet subit un routage, et son parcours dépend du trafic). Ils sont livrés dans le désordre.



– En mode connecté, un circuit est établi de bout en bout, avant de transférer le message, comme pour la commutation de circuits. Mais il s'agit bien de la commutation de paquets (le circuit est virtuel). Avant de transférer les paquets de données, un paquet (appelé paquet d'appel) subit un routage adaptatif, pour parcourir un chemin optimal. Au fur et à mesure de son parcours, les ressources sont réservées, et les tables de routage des nœuds traversés sont fixées. Ainsi, tous les paquets suivants d'un même message suivront le même chemin (appelé circuit virtuel commuté ou CVC), et sont livrés dans l'ordre d'émission. Après le transfert des données, un paquet (de libération) parcourt le CVC, et le ferme en restituant ses ressources.



Le transfert des paquets de données est plus rapide, car ils ne sont pas retardés par l'exécution d'un algorithme de routage.

Le contrôle d'erreur dans les réseaux à commutation de paquets peu se faire en deux manières :

1. Un contrôle d'erreur par liaison de données : Le paquet, encapsulé dans une trame, est contrôlé dans chaque nœud de commutation. Cette méthode augmente la fiabilité, mais aussi le délai de transit.
2. Un contrôle d'erreur de bout en bout : Il n'y a pas de contrôle d'erreur dans les nœuds de commutation. Le contrôle est effectué aux extrémités par les usagers du réseau de commutation. Le transfert est moins fiable, mais plus rapide.

5.2 Adressage et nommage

Le but de l'adressage est de fournir un service de communication universel permettant à toute machine de communiquer avec toute autre machine. Les machines doivent être accessibles aussi bien par des humains que par d'autres machines. Une machine doit pouvoir donc être identifiée par :

- une adresse qui doit être un identificateur universel de la machine,
- un nom (mnémotechnique pour les utilisateurs),
- une route précisant comment la machine peut être atteinte.

Les noms sont plus faciles à utiliser par les utilisateurs, mais ne permettent pas l'identification des machines et nécessitent un moyens de faire la correspondance avec les adresses.

Les adresses forment le moyen d'identifier de manière unique chacun de ses utilisateurs.

On utilise dans les réseaux, selon la cas, deux types d'adressage :

- Adressage hiérarchique : l'adresse est décomposée en différentes parties qui permettent d'identifier :
  - le réseau auquel l'utilisateur est rattaché
  - le point d'accès par lequel il est raccordé au réseau

– l'utilisateur dans l'installation locale Le champ adresse diminue au fur et à mesure de la progression des blocs dans le réseau. L'exemple courant de ce type est la numérotation téléphonique

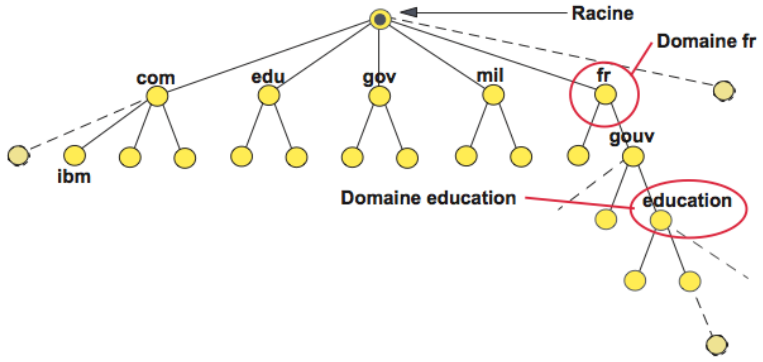
– Adressage à plat : le format de l'adresse n'a aucune signification particulière quant à la localisation de l'entité communicante. L'exemple est l'adresse MAC.

5.2.1 Nommage

La notion de nommage est complémentaire de celle d'adressage, l'un désigne l'objet, l'autre précise sa localisation. Indépendamment qu'il est plus aisé de manipuler des noms que des adresses, l'avantage du nommage est essentiellement de dissocier l'objet de sa localisation géographique. Le déplacement de l'objet nommé est transparent à l'utilisateur. De manière similaire à l'adressage, le nommage utilise deux modes de représentation :

– Le nommage à plat ou horizontal, ce type de nommage impose une démarche rigoureuse pour garantir l'unicité d'un nom sur l'ensemble du réseau. NetBios, protocole allégé mis en œuvre dans les réseaux locaux, utilise un nommage à plat.

– Le nommage hiérarchique ou arborescent, plus souple, organise le nommage en domaines. Cette technique autorise une représentation des objets calquée sur l'organisation de l'entreprise. Chaque nœud peut être un domaine dont la gestion peut être confiée à une autorité particulière. Ce mode de représentation et d'administration convient parfaitement à la gestion d'un annuaire très important comme celui d'Internet.



DNS (Domain Name System) est un système d'annuaire permettant de faire la correspondance entre le nom symbolique et l'adresse IP. Les serveurs de noms, ou serveurs DNS, sont des machines qui assurent la traduction des noms en adresses IP et réciproquement. On parle de résolution d'un nom ou d'une adresse. Pour cela, ils possèdent des informations sur l'architecture de l'arbre et sur les données associées.

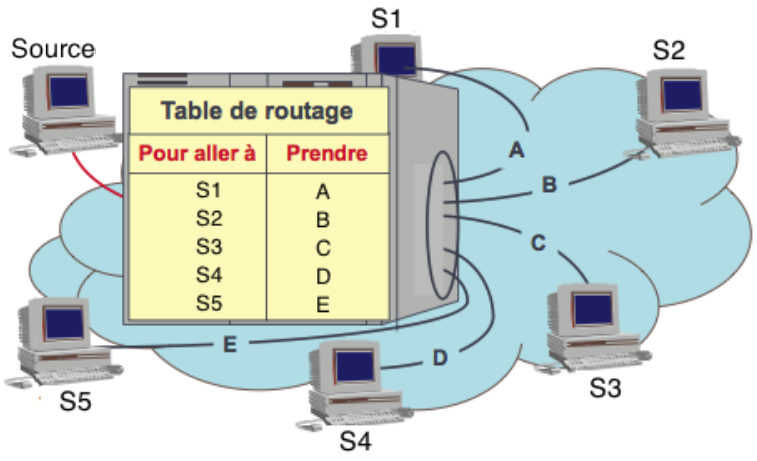
5.3 Routage

5.3.1 Principe

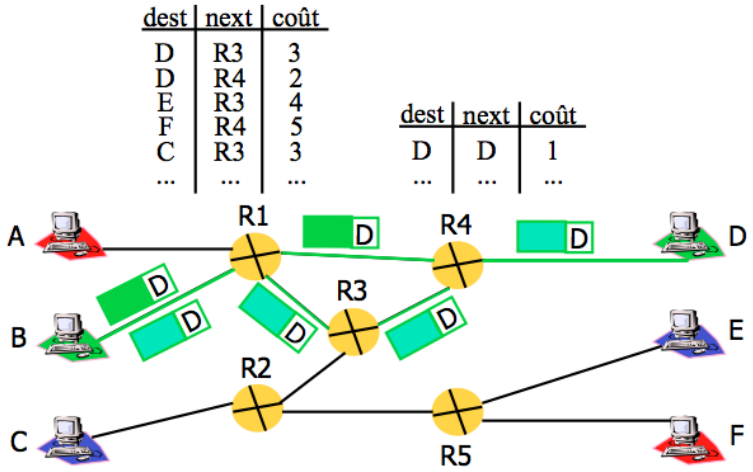
Le but du routage est la détermination d'un chemin à travers le réseau entre une machine émettrice et une machine réceptrice, toutes deux identifiées par leur adresse. Les protocoles de routage établissent des règles d'échange entre routeurs pour mettre à jour leurs tables d'informations selon des critères de coût comme, par exemple, la distance, l'état de la liaison, le débit. Ils améliorent ainsi l'efficacité du routage.

Il y a de très nombreux problèmes à résoudre. L'un des problèmes fondamentaux à éviter réside dans les boucles de routage (le message peut "tourner en rond" dans le réseau et ne jamais atteindre son destinataire). L'autre apparait lorsqu'il y a une panne dans le réseau et qu'il faut optimiser le calcul des nouvelles routes : une fois la panne détectée, il faut transmettre l'information sur l'évènement le plus rapidement possible pour que les différents routeurs recalculent par où faire passer leurs messages en contournant la liaison en panne.

Chaque nœud du réseau comporte des tables, dites tables d'acheminement couramment appelées tables de routage, qui indiquent la route à suivre pour atteindre le destinataire. En principe, une table de routage est un triplet <Adresse destination>/<Route à prendre>/<Coût>.



Chaque paquet doit contenir une information qui permet d'identifier le destinataire pour que chaque routeur puisse prendre une décision (@destination). Le "coût" permet de choisir la route appropriée si plusieurs routes sont possibles (coût pour atteindre la destination). Par exemple dans la figure suivante, le coût et représenté par le nombre de sauts (de routeurs) pour atteindre la destination.



Un protocole de routage commence par la construction des tables de routage au niveau de chaque nœud, ensuite utilise ces tables pour l'acheminement des paquets entre les stations. À priori, aucun routeur n'a une vision globale de la route que prendront les paquets : les paquets sont relayés de proche en proche jusqu'au destinataire. L'émetteur du paquet doit connaître le premier routeur relais, ensuite, chaque routeur est chargé d'acheminer le paquet vers le routeur suivant jusqu'à la destination finale.

Un protocole de routage efficace doit assurer :

- une taille minimale pour les tables de routage,
- Une consultation rapide des tables,
- Une consommation acceptable de mémoire
- Le moins possible d'informations à échanger
- minimiser la fréquence des messages de contrôle générés par l'échange des informations de routage
- être robuste en évitant les boucles,la surcharge de certains routeurs...
- trouver une route optimale

En général des types de routage existent :

5.3.2 Routage statique

Dans le routage statique (ou non adaptatif), la décision de routage ne dépend pas de l'état du réseau ; le choix de la route est défini par l'administrateur une fois pour toute lors de l'initialisation et elles sont mises à jour uniquement quand la topologie du réseau change (cas de panne).

Le routage statique est simple mais ne recherche pas la route optimale et n'est pas adapté à la défaillance d'un lien. Il est généralement adapté aux petits réseaux et aux

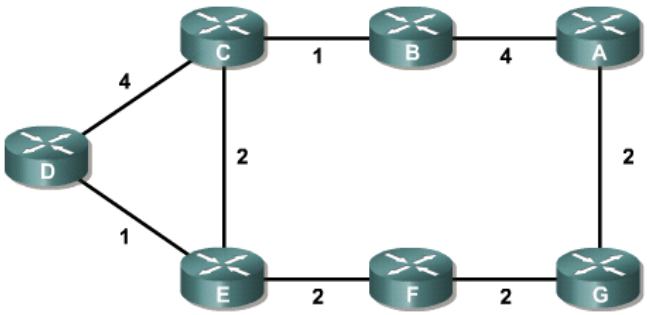
réseaux dans lesquels le choix de la route est limité (routes rentrées manuellement). Cependant, il assure le séquencement des paquets même en mode non connecté car tous les paquets prennent la même route.

Deux variantes de routage statique sont utilisées :

5.3.2.1 Plus court chemin

Dans ce type de routage, l'administrateur calcule les chemins minimums entre chaque deux stations (réseaux) puis introduits les tables correspondant à ces chemins dans chaque routeur. Le réseaux tout entier est représenté par un graphe pondéré. Le coût (métrique) de chaque arrête peut être calculé selon :

- le nombre de saut (nombre de routeurs traversés)
- la distance réelle (en km) entre deux routeurs
- le délai de transmission (temps de latence)
- le nombre de paquets moyen dans les files d'attente
- le taux d'erreurs moyen
- le trafic moyen observé, ...



Le calcul des chemins est effectué en utilisant des algorithmes connus en théorie des graphes tel que Dijkstra.

5.3.2.2 Inondation

Dans le routage par inondation, chaque nœud envoie le message sur toutes ses lignes de sortie, sauf celle d'où provient le message. Pour éviter une surcharge du réseau, chaque message comporte un compteur de sauts. Le compteur est initialisé à l'émission (nombre de sauts autorisés) et décrémenté par chaque nœud. Le message est détruit quand le compteur de sauts est à zéro. Pour éviter les bouclages, les messages sont numérotés, chaque nœud

mémorise cet identifiant et détruit les messages déjà vus.

Ce système est très robuste, il résiste à la destruction de plusieurs lignes et garantit de trouver toujours le plus court chemin ; il est utilisé dans certaines communications militaires et par certains protocoles de routage pour diffuser les informations d'états du réseau. Il est utilisé, également pour découvrir le chemin optimal et en déduire une route statique.

5.3.3 Routage dynamique

Dans le Routage dynamique, les tables de routage sont construites automatiquement sans l'intervention de l'administrateur. Les routeurs échangent régulièrement leurs états et mettent à jour leurs tables de routage. Ce routage est plus complexe que le routage statique et surcharge le réseau par l'échange d'informations de routage et ne garantit pas le séquencement en mode non connecté. Cependant, il permet de choisir la route optimale.

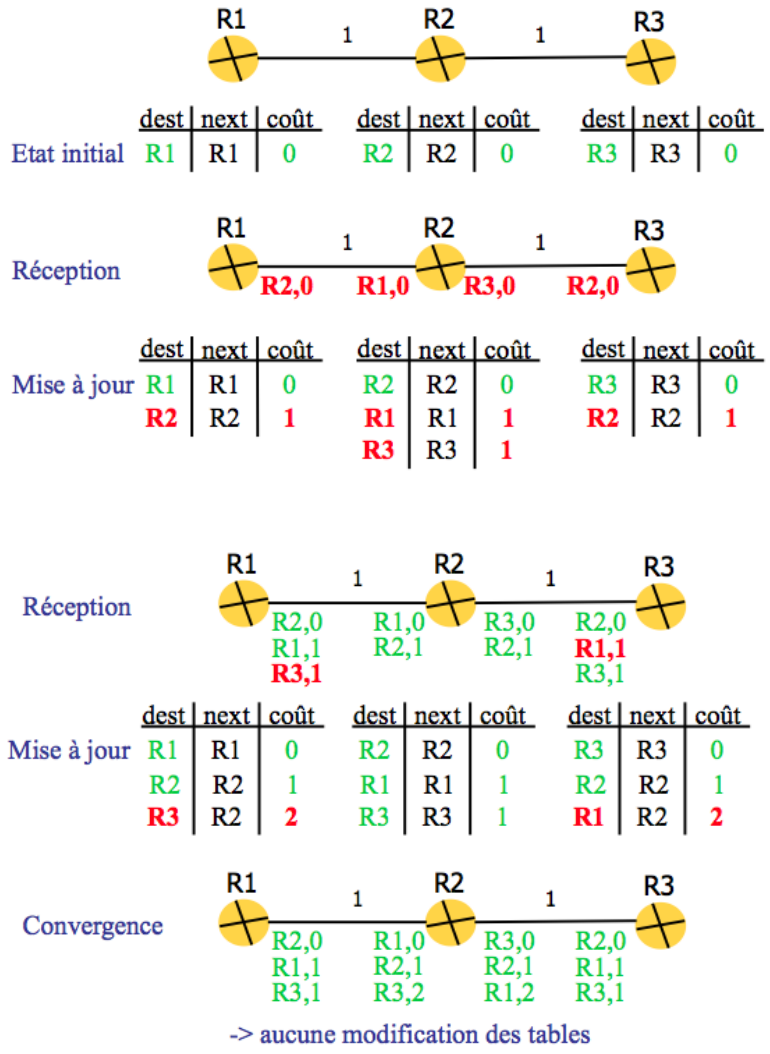
Il existe trois algorithmes de routage dynamique :

5.3.3.1 Routage à Vecteur de distance

Dans le routage à vecteur de distance ou routage de Bellman-Ford (distance vector routing), chaque nœud du réseau maintient une table de routage qui comporte une entrée par nœud du réseau et le coût pour joindre ce nœud. Périodiquement chaque nœud diffuse sa table de routage à ses voisins. Le nœud destinataire apprend ainsi ce que son voisin est capable de joindre. À réception, le nœud compare les informations reçues à sa propre base de connaissance :

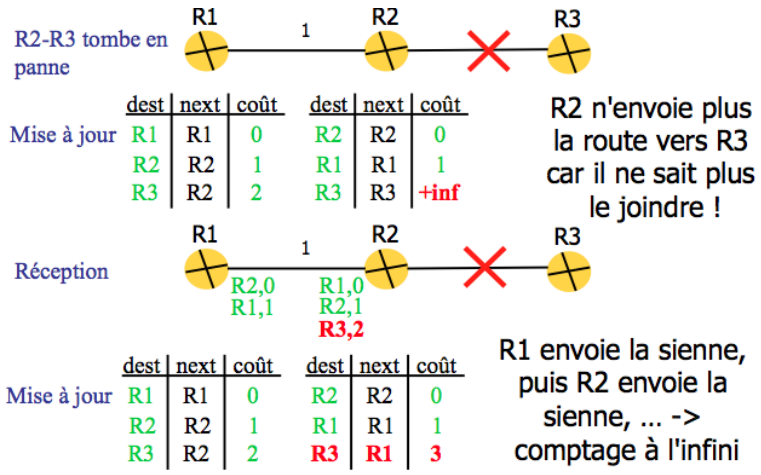
- La table reçue contient une entrée qui n'est pas déjà dans sa propre table, il incrémente le coût de cette entrée du coût affecté au lien par lequel il vient de recevoir cette table et met cette entrée dans sa table. Il a ainsi appris une nouvelle destination.
- La table contient une entrée qu'il connaît déjà. Si le coût calculé (coût reçu incrémenté du coût du lien) est supérieur à l'information qu'il possède, il l'ignore sinon il met sa table à jour de cette nouvelle entrée.

De proche en proche chaque nœud apprend la configuration du réseau et le coût des différents chemins. La convergence des différentes tables peut être assez longue. Les deux figures suivantes en représentent un exemple.



Deux problèmes majeurs peuvent survenir dans cet algorithme en cas de panne d'un lien entre deux routeurs :

1. Les boucles de routage : la panne d'un lien peut conduire à une bouclage infini des paquets dans le réseau. La figure suivante illustre ce phénomène :tous les paquets à destination de R3 oscillent entre R1 et R2



2. l'algorithme ne converge plus : à l'échange suivant, R1 apprend de R2 que désormais le coût pour joindre R3 en passant par R2 est de 3 -> il met sa table à jour (R3, R2, 4) ; de même, R2 va apprendre de R1 que désormais le coût pour joindre R3 est de 4...

Les solutions utilisées consistent à :

- interdire à un nœd de signaler une destination qu'il connaît au routeur par lequel il l'a apprise (split horizon)
- limiter la valeur infinie du coût à une petite valeur (16 dans RIP) -> convergence dés que l'infini est atteint

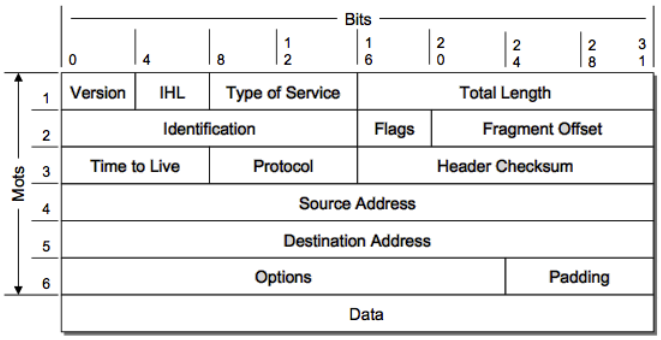
5.4 Protocole IP

Le protocole IP fait partie de la couche Internet de la suite de protocoles TCP/IP. C'est un des protocoles les plus importants d'Internet car il permet l'élaboration et le transport des datagrammes IP (paquets), sans toutefois en assurer la livraison. En réalité, le protocole IP traite les datagrammes IP indépendamment les uns des autres en définissant leur représentation, leur routage et leur expédition.

Le protocole IP détermine le destinataire d'un paquets grâce à trois informations disponible au niveau de chaque machine : l'adresse IP, le masque de sous-réseau, l'adresse de la passerelle par défaut. Les tables de routages établies au niveaux des routeurs permettent de déterminer les routes à suivre par les paquets. Le protocole RIP (routing information protocol) utilisant un algorithme de type vecteur à distance permet de construire et mettre à jour les tables de routage.

5.4.1 Structure d'un paquet IP

Le datagramme IP contient un en-tête IP suivi des données IP provenant des protocoles des couches supérieures.



- Par défaut, la longueur de l'en-tête est de 5 mots de 32 bits (soit 20 octets) ; le sixième mot est facultatif. Puisque la longueur de l'en-tête est variable, elle inclut un champ appelé Internet Header Lenght (IHL - longueur de l'en-tête Internet) en mots.
- Le champ Version fait quatre bits et indique le format de l'en-tête IP (IPv4 ou IPv6).
- Le champ Type of Service (TOS) informe les réseaux de la qualité de service désirée, spécifiant ainsi les délais, le débit et la fiabilité. Les implémentations actuelles ignorent ce champ.
- Le champ Total length (longueur totale) contient la longueur de l'en-tête et des données IP, en octets.
- La durée de vie (Time To Live) représente la durée maximale de vie d'un datagramme sur le réseau. Cette valeur est décrémentée à chaque routeur. Lorsque le champ TTL tombe à 0, datagramme IP est écarté par le routeur.

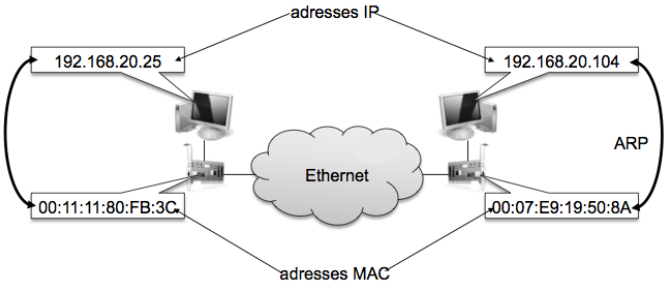
Plusieurs autres protocoles sont utilisés avec le protocole IP dans la couche réseaux notamment :

5.4.2 RIP

RIP (Routing Information Protocol) est un protocole de routage IP de type vecteur distance basé sur l'algorithme de routage décentralisé Bellman-Ford. Il utilise le nombre de sauts (hops) comme métrique de distance entre les machines. Chaque routeur diffuse sa table de routage à ses voisins toutes les 30 secondes. En recevant une table de routage d'un voisin, un routeur met à jour la sienne. Le TTL max étant fixé à 16. Un routeur supprime une information de sa table si elles n'est pas confirmée pour 180 secondes. Il existe deux versions de RIP, la deuxième étant une amélioration de la première : RIPv1 et RIPv2.

5.4.3 ARP

En TCP/IP les interfaces du réseau sont modélisées par un unique identificateur à 32 bits, l'adresse IP. Or la transmission des datagrammes IP sur le réseau physique nécessite que ces datagrammes soient encapsulés dans des trames de couche de liaison de données (couche 2), telles que Ethernet ou Token-Ring, qui elles contiennent des adresses "physiques" (adresses MAC).



Un mécanisme souple, implémenté sous forme d'un protocole distinct et appelé ARP (Address Resolution Protocol) permet de déterminer dynamiquement l'adresse MAC à partir de l'adresse IP d'un hôte.

Pour déterminer l'adresse matérielle de l'hôte B avant de lui envoyer son message, la station A envoie sur le réseau une trame MAC de diffusion, appelée trame ARP de requête. Celle-ci contient les adresses IP et MAC de l'hôte A émetteur, ainsi que l'adresse IP de la destination B. La trame inclut un champ destiné à contenir l'adresse MAC de B. Tous les nœuds du réseau physique reçoivent la trame ARP. Seul l'hôte dont l'adresse IP correspond à l'adresse requise dans la trame de requête ARP répond en encodant sa propre adresse matérielle dans une trame de réponse ARP. L'hôte A initialise alors sa table cache ARP (conservée en mémoire) en utilisant la réponse fournie. Les entrées dans cette table expirent après une temporisation donnée qui peut être configurée dans certaines implémentations de TCP/IP (généralement 15 mn). Le cache ARP est consulté par un hôte juste avant l'envoi d'une requête ARP ; si la réponse se trouve dans le cache, la requête n'est pas effectuée.

5.4.4 RARP

RARP (Reverse ARP) est un mécanisme utilisé par les stations sans disques (terminaux) pour obtenir leur adresse IP auprès d'un serveur distant. Le nœud qui veut connaître sa propre adresse envoie en diffusion une requête RARP. S'il existe plusieurs serveurs RARP, chacun d'eux tente de traiter la requête RARP. Généralement, le client RARP accepte la première réponse reçue et ignore silencieusement les suivantes.

5.4.5 ICMP

Le protocole ICMP (Internet Control Message Protocol) envoie des messages qui réalisent des fonctions de contrôle, de détection des erreurs, et de transmission d'informations pour TCP/IP. Il existe 18 types de messages ICMP.

La commande *traceroute*, par exemple, utilise également des messages ICMP ; elle permet de connaître la route exacte empruntée par les datagrammes. traceroute envoie 3 paquets UDP avec un TTL égal à 1 puis recommence en augmentant le TTL de 1 à chaque envoi. A chaque fois que le TTL arrive à 0, le routeur renvoie un message ICMP d'erreur.

5.4.6 IGMP

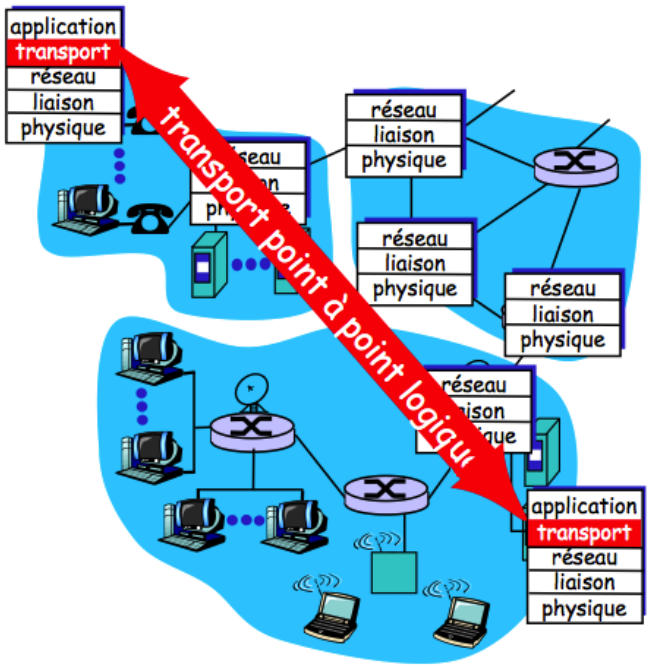
IGMP (Internet Group Management Protocol) permet aux machines de déclarer leur appartenance à un ou plusieurs groupes auprès du routeur multipoint dont elles dépendent soit spontanément soit après interrogation du routeur. Celui-ci diffusera alors les datagrammes destinés à ce ou ces groupes. IGMP comprend essentiellement deux types de messages : un message d'interrogation (Host Membership Query), utilisé par les routeurs, pour découvrir et/ou suivre l'existence de membres d'un groupe et un message de réponse (Host Membership Report), délivré en réponse au premier, par au moins un membre du groupe concerné.

Chapitre 6

Couche Transport (2 cours)

Le service fourni par les protocoles de la couche réseaux (IP par exemple) n'étant pas fiable, il faut implanter par dessus un protocole supplémentaire, en fonction de la qualité de service dont les applications ont besoin. Ce protocole est appelé protocole de transport et appartient à la couche OSI numéro 4, la couche transport.

La couche transport assure une communication entre deux machines dans un réseau quelque soit leur localisation en masquant les détails vus dans la couche réseaux tel que le routage. Elle fournit donc une communication logique de bout en bout entre processus tournant sur des machines différentes. Les protocoles de transport tournent sur les hôtes et non pas sur les routeurs.



Il est important de noter qu'il existe une grande ressemblance entres les fonctionnalités de la couche transport 4 et la couche liaison 2. Cependant, la couche liaison adresse las machines tandis que la couche transport adresse les applications. Les tâche essentielles de la couche transport sont les suivantes :

- Segmentation des messages,
- Séquencement des messages,
- Multiplexage et démultiplexage des données des applications,
- Protection contre les éventuelles erreurs,
- Assurer les qualité de service souhaitée par les applications.

6.1 Segmentation des messages

La couche transport émettrice divise (fragmente) le message reçu de la couche supérieure (session pour OSI et application pour TCP/IP) en segments qui les passe à la couche réseau. L'unité de données manipulée par le ouche transport est donc le segment. La couche transport réceptrice, reforme le message à partir des segments obtenus de la couche réseau.

6.2 Séquencement des messages

Si le service offert par la couche réseau n'est pas fiable, la couche transport doit le fiabiliser en garantissant la reconstruction correcte du message pour les application ayant besoin. Pour cela les segments doivent être numérotés et acquittés exactement comme dans la couche liaison, mais ici entre deux applications distantes.

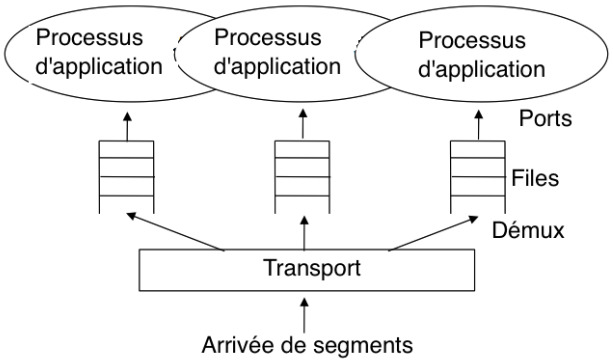
6.3 Multiplexage et démultiplexage

Puisque plusieurs application peuvent tourner sur une machine et peuvent toutes envoyer et recevoir des messages, la couche transport doit pouvoir :

- lors de l'émission, collecter ces données et et les encaplsuler dans le même médium : service multiplexage,
- lors de la réception, livrer les segments reçus aux bonnes applications : service de démultiplexage.

Pour pouvoir distinguer les applications (processus) des unes des autres, elles doivent être identifiées par des adresses. Chaque processus utilisant le réseau doit obtenir une adresse. Par exemple, dans le protocole TCP d'Internet on utilise les numéro de port pour

adresser les applications. Si les trames portent des adresses MAC et les paquets des adresses IP, les segments portent des numéros de port.



Dans le protocole TCP, le numéro de port est un entier représenté sur deux octets. Par exemple, les applications http utilisent le port numéro 80.

6.4 Protection contre les erreurs

En plus de la protection contre les erreurs effectuée à chaque liaison, la couche transport en effectue une autre de bout en bout. Généralement, c'est une somme (checksum) effectuée sur la totalité du segment envoyé. Le champ Checksum est codé sur 16 bits et représente la validité du paquet de la couche 4 TCP.

Pour TCP, le Checksum est constitué en calculant le complément à 1 sur 16 bits de la somme des compléments à 1 des octets de l'en-tête et des données pris deux par deux (mots de 16 bits).

Par example, soit le segment à envoyer est "0x450000304422400080060000 8c7c19acae241e2b". On commence par diviser le segment en blocs de 16 bits chacun puis calculer leurs somme :

$$4500 + 0030 + 4422 + 4000 + 8006 + 0000 + 8c7c + 19ac + ae24 + 1e2b = 2BBCF$$

Puis mettre le résultat en 16 bits par l'ajout du reste au résultat :

$$2 + BBCF = BBD1 = 1011101111010001$$

Finalement, on calcule le complément à 1 du résultat

$$\text{checksum} = \text{complement à un } (1011101111010001) = 0100010000101110 = 442E$$

La vérification chez le récepteur est faite en utilisant le même algorithme avec l'initialisation du checksum à la valeur 442E.

2BBCF + 442E = 2FFFD, alors 2 + FFFD = FFFF

On prend le complément à un de FFFF = 0.

Le segment est correct si le checksum est à 0.

6.5 Protocoles de transport

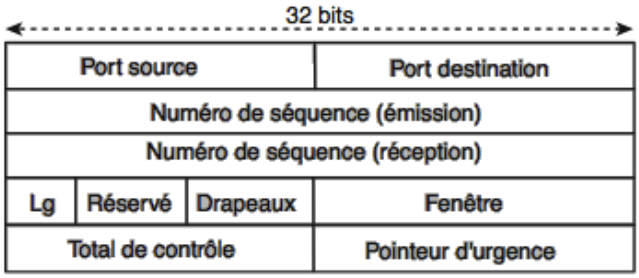
Les applications utilisant les réseaux peuvent avoir besoin de deux type de service : connecté et non connecté. Le service connecté peut être utilisé lorsqu'on connait l'adresse du destinataire au préalable tel que les serveurs http, ou ftp. Tandis que le service non connecté peut être utilisé par exemple pour la diffusion. Le monde d'Internet (TCP/IP) utilise les deux protocoles TCP et UDP correspondant chacun à un service.

6.5.1 TCP

TCP (Transport Control Protocol) assure un service de transmission de données de bout en bout, connecté, fiable avec détection et correction d'erreurs .

6.5.1.1 Format d'un segment TCP

Un segment TCP est constitué comme suit :



- Port Source (16 bits) : Numéro du port utilisé par l'application en cours sur la machine source.
- Port Destination (16 bits) : Numéro du port relatif à l'application en cours sur la machine de destination.
- Numéro d'ordre (32 bits) : numéro du premier octet du flux de données qui sera transmis (Initial Sequence Number)
- Numéro d'accusé de réception (32 bits) : Numéro d'ordre du dernier octet reçu par le récepteur (par rapport à tous les octets du flot de données reçues).

- Longueur en-tête (4 bits) : Il permet de repérer le début des données dans le segment. Ce décalage est essentiel, car il est possible que l'en-tête contienne un champ d'options de taille variable. Un en-tête sans option contient 20 octets, donc le champ longueur contient la valeur 5, l'unité étant le mot de 32 bits (soit 4 octets).
- Réservé (6 bits) : Champ inutilisé.
- Drapeaux ou flags (6 bits) Ces bits sont à considérer individuellement :
  - URG (Urgent) : Si ce drapeau est à 1, le segment transporte des données urgentes dont la place est indiquée par le champ Pointeur d'urgence (voir ci-après).
  - ACK (Acknowledgement) : Si ce drapeau est à 1, le segment transporte un accusé de réception.
  - PSH (Push). Si ce drapeau est à 1, le module TCP récepteur ne doit pas attendre que son tampon de réception soit plein pour délivrer les données à l'application. Au contraire, il doit délivrer le segment immédiatement, quel que soit l'état de son tampon (méthode Push).
  - RST (Reset) : Si ce drapeau est à 1, la connexion est interrompue.
  - SYN (Synchronize) : Si ce drapeau est à 1, les numéros d'ordre sont synchronisés (il s'agit de l'ouverture de connexion).
  - FIN (Final) : Si ce drapeau est à 1, la connexion se termine normalement.
- Fenêtre (16 bits) : Champ permettant de connaître le nombre d'octets que le récepteur est capable de recevoir sans accusé de réception.
- Total de contrôle ou checksum (16 bits). Le total de contrôle est réalisé en faisant la somme des champs de données et de l'en-tête. Il est calculé par le module TCP émetteur et permet au module TCP récepteur de vérifier l'intégrité du segment reçu.
- Pointeur d'urgence (16 bits). Indique le rang à partir duquel l'information est une donnée urgente.
- Options (taille variable).

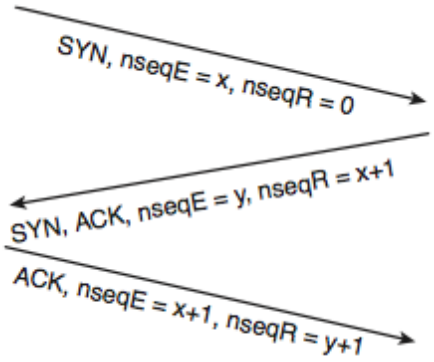
6.5.1.2 Connexion

TCP est un protocole qui fonctionne en mode client/serveur : l'un des utilisateurs est le serveur offrant des services (serveur http : site web), l'autre est le client (client http : navigateur web) qui utilise les services proposés par le serveur.

Le serveur doit être initialisé le premier ; on dit qu'il exécute une ouverture passive. Dès qu'il est opérationnel, il attend les demandes des clients, qui peuvent alors faire une ouverture active. Pour ouvrir une connexion, les clients doivent connaître le numéro de

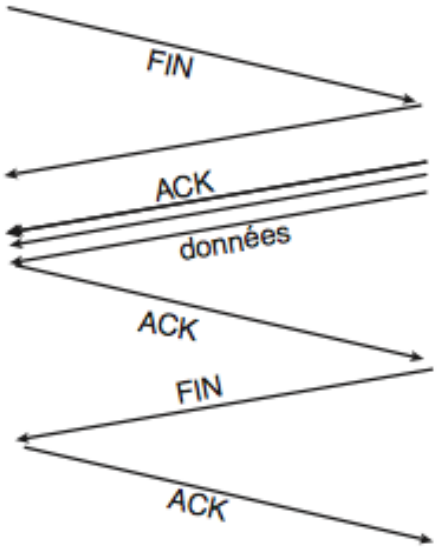
port de l'application distante. En général, les serveurs utilisent des numéros de ports bien connus, mais il est possible d'implanter une application sur un numéro de port quelconque. Il faut alors prévenir les clients pour qu'ils sachent le numéro de port à utiliser.

Le client ouvre la connexion en envoyant un premier segment, parfois appelé séquence de synchronisation. La connexion est établie en trois phases (*three-way-handshake*) en spécifiant les numéro de séquences des octets à échanger.



Une fois la connexion établie, elle peut être utilisée pour échanger les segments dans les deux sens.

La déconnexion est indépendante pour les deux sens ; un échange dans un sens peut être interrompu en gardant celui de l'autre sens. Pour fermer la connexion dans les deux sens, on utilise quatre phases.



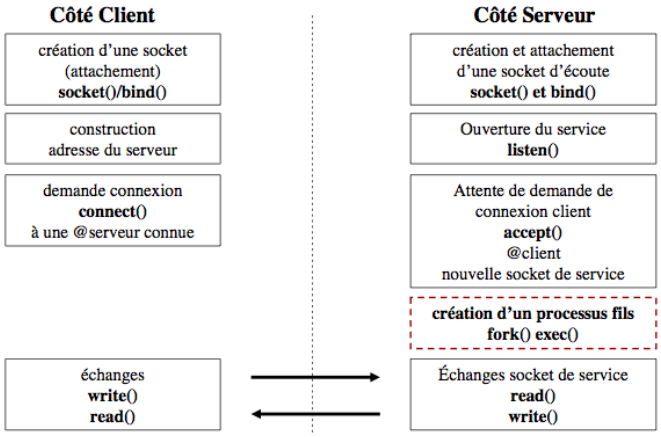
Les applications utilisent un concept appelé socket (initialement développé dans UNIX), permettant d'identifier les différentes communication sur la même machine. Le socket est constitué de la paire :

< adresse IP locale, numéro de port local >

Pour identifier de manière unique l'échange de données avec le processus applicatif distant, le protocole de transport utilise un ensemble de cinq paramètres formé par le nom du protocole utilisé, le socket local et le socket distant :

< protocole, adresse IP locale, numéro de port local ; adresse IP distante, numéro de port distant >

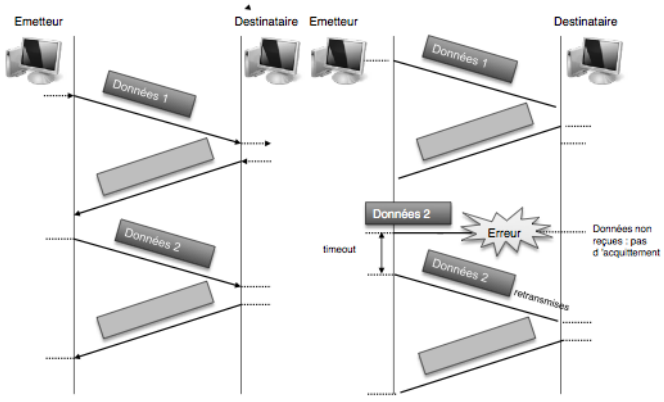
Les serveur peut accepter plusieurs connexions simultanément en créant pour chacune son propore soket avec un processus différent pour sa gestion.



Dans le modèle OSI, les sockets sont gérés par la couche session.

6.5.1.3 Fiabilité

L'utilisation d'un mécanisme appelé PAR (Positive Acknowlegment with Retransmission, Accusé de réception positif avec la retransmission) permet à TCP de garantir des transmissions fiables.



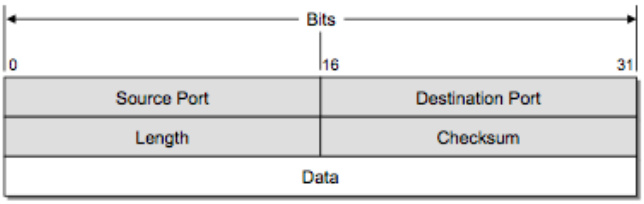
Chaque segment envoyé contient un checksum que le destinataire utilise pour vérifier que les données n'ont pas été endommagées pendant leur transmission. Si le segment est correctement reçu, le récepteur renvoie un accusé de réception positif à l'émetteur. Dans la négative, le récepteur élimine ce segment de données. Après un délai d'attente déterminé, le module TCP d'envoi retransmet les segments pour lesquels aucun accusé de réception positif n'a été reçu.

6.5.2 UDP

Le protocole UDP permet aux applications d'accéder directement à un service de transmission de datagrammes, tel que le service de transmission qu'offre IP. UDP est caractérisé par :

- UDP possède un mécanisme permettant d'identifier les processus d'application à l'aide de numéros de port UDP.
- UDP est orienté datagrammes (sans connexion), ce qui évite les problèmes liés à l'ouverture, au maintien et à la fermeture des connexions.
- UDP est efficace pour les applications en diffusion/multidiffusion. Les applications satisfaisant à un modèle du type "interrogation-réponse" peuvent également utiliser UDP. La réponse peut être utilisée comme étant un accusé de réception positif à l'interrogation. Si une réponse n'est pas reçue dans un certain intervalle de temps, l'application envoie simplement une autre interrogation.
- UDP ne séquence pas les données. La remise conforme des données n'est pas garantie.
- UDP peut éventuellement vérifier l'intégrité des données (et des données seulement) avec un total de contrôle.
- UDP est plus rapide, plus simple et plus efficace que TCP mais il est moins robuste.

Le datagramme UDP est organisé comme suit :



- Port source (16 bits) : Il s'agit du numéro de port correspondant à l'application émettrice du paquet. Ce champ représente une adresse de réponse pour le destinataire.
- Port destination (16 bits) : Contient le port correspondant à l'application de la

machine à laquelle on s'adresse.

- Longueur (16 bits) : Précise la longueur totale du datagramme UDP, exprimée en octets.
- Total de contrôle ou checksum (16 bits). Bloc de contrôle d'erreur destiné à contrôler l'intégrité de l'entête du datagramme UDP.

Les processus applicatifs utilisent des sockets UDP. Leur manipulation est très simple puisque le protocole n'est pas en mode connecté : il n'y a pas de procédure de connexion et donc pas de fermeture non plus. Comme pour TCP, du côté du serveur, il faut d'abord créer le socket et le paramétrer par la primitive Bind, en lui associant le numéro de port correspondant. Puis il faut le placer dans un état d'attente des données du client (primitive Listen). Côté client, il faut créer le socket. Le transfert des données peut commencer directement en utilisant des primitives Read et Write.

Chapitre 7

Couches applicatives (2 cours)