IUP GMI D'AVIGNON

www.centraliup.fr.st

Réseaux & Télécoms

Patrick Isoardi

Réseaux & Télécoms

Table des matières

Version 1.1 ◆ Publiée le 24/11/2002

Eléments fondateurs

Pour aboutir à la création des premiers réseaux informatiques, à l'approche des années 80, il fallait réunir tous les éléments fondateurs de la communication moderne :

- Le **support physique** : le câble téléphonique, le réseau hertzien, la fibre optique, etc.
- Le **codage de l'information** : le morse, etc.
- Le **protocole de communication** (spécifique aux réseaux par rapport aux autres moyens de communication) : il permet l'échange de données, en utilisant le codage de l'information, mais aussi gère les interruptions sur la ligne, prépare la connexion, etc.

Voici quelques services que peuvent nous offrir les réseaux de nos jours :

- Téléphonie et fax ;
- Transfert de fichiers ;
- Partage de périphériques ;
- Emulation de terminal à distance ;
- Exécution de commandes à distance ;
- Courrier électronique ;
- Le Web : Internet ;
- Vidéo à la demande et visioconférence ;
- Accès aux données et aux traitements répartis ;
- Architecture client/serveur.

Globalement, les réseaux permettent la communication entre machines, locales ou non.

Classification

On peut classifier les différents types de réseaux selon leur taille et leur ampleur :

- Les **réseaux locaux** : LAN (Local Area Network) : par exemple 250 machines sous le même bâtiment.
- Les **réseaux de 'campus'** : plusieurs bâtiments reliés entre eux.
- Les **réseaux de grande amplitude** : MAN (Metropolitan Area Network) ou WAN (Wild Area Network) dont les liaisons distantes peuvent être de plusieurs types : le RTC (Réseau Téléphonique Commuté) et ses évolutions avec les mobiles, ou les liaisons spécialisées (par exemple les fibres optiques aux Etats-Unis).
- Les **réseaux fédérateurs** (le « backbone ») : Internet, Renater (met en relation toutes les Universités) ou R3T2.

Les liaisons

Les liaisons sont soit **directes**, c'est-à-dire qu'elles correspondent aux premiers besoins (imprimantes déportées, terminaux déportés, transferts de fichiers, sauvegardes), soit **distantes**, utilisant de la technologie standard (modem), mais dont la bande passante est limitée par rapport aux liaisons directes.

Pour ce qui est du support de ces liaisons, elles peuvent utiliser :

- Le **cuivre** : coaxial (télévision) ou paire torsadée (téléphonie) : LAN ou boucle locale.
- L'**optique** : infrastructure des opérateurs, câbles océaniques, liens « haut-débit », situations particulières.
- Le « **sans-fil** » : hertzien (mobile courte et longue distance, satellites) ou laser et dérivées.

Rôle

Le réseau peut être organisé physiquement de différentes manières :

- **Bus** : réservé aux LAN (les machines sont les unes derrière les autres)
- **Etoile** : pour les LAN et MAM (nécessite un point central : le hub)
- **Anneau** : token ring dans les LAN
- **Arbre** : SNA de IBM
- **Maillé** : désordre… pour les réseaux qui n'ont pas d'architecture propre, comme Internet.
- **Topologies logiques** liées aux protocoles.

Un modèle commun

La **norme OSI** (Open System Interconnexion) est due à l'organisme ISO (International Standardisation Organisation). Mise en place depuis plus de 10 ans, elle est aujourd'hui très performante. Son rôle est de mettre en communication des machines de façon totalement transparente pour l'utilisateur.

Elle est basée sur des couches spécifiques successives, libérant ainsi l'opérateur des problèmes physiques ou protocolaires, liées à la technique, correspondant aux couches les plus basses.

Pour quelles raisons a-t-elle été mise au point ?

- Formalisme complet ;
- Besoin d'abstraction pour les utilisateurs ;
- Répondre aux problèmes posés par l'évolution des systèmes d'information, vers toujours plus d'hétérogénéité ;
- Définition d'une gamme de services permettant de travailler en communication.

Structure en couches

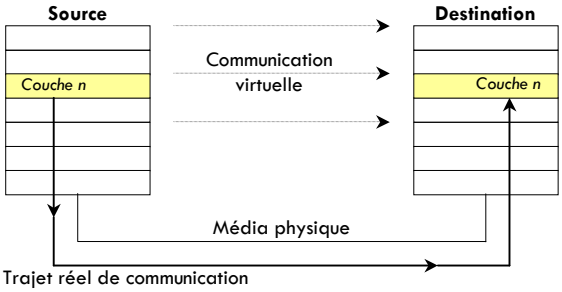
La norme OSI propose une analyse du réseau par **couches,** partant de besoins proches de l'utilisateur, et allant de plus en plus vers le matériel :

- Qu'est ce qui circule ?
- Sous quelles formes différentes ?
- Quelles règles régissent le flux ?
- Où cette circulation se produit-elle ?

Ces couches sont régies par quelques règles et principes de base :

- L'accès au modèle se fait par la partie supérieure de l'empilement ;
- Deux couches sont indépendantes, c'est-à-dire que chaque couche s'occupe d'un problème particulier ;
- La coopération entre deux couches de niveaux différents de fait par offre de service sans utilisation de protocole (passage d'informations très simples)
- La coopération entre deux couches de même niveau n (donc pour des machines différentes) utiliser le protocole de communication associé au niveau n.

Le **protocole** est l'ensemble des règles qui définissent les communications.



Deux couches de même niveau communiquent à travers des couches successives, mais utilisent le même protocole de communication. Pour l'utilisateur, tout se passe comme si la communication suivait le trajet en pointillé (communication virtuelle).

Intérêt

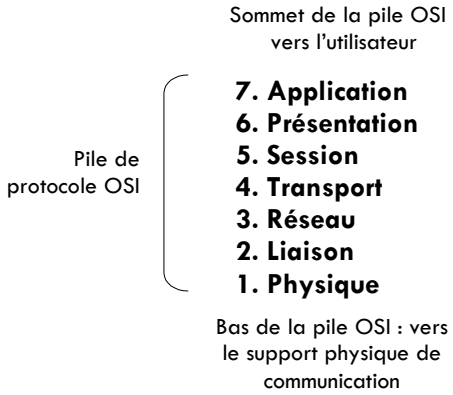
Les intérêts de cette structure en couches sont nombreux :

- **Simplification** : les fonctions homogènes sont regroupées ;
- **Indépendance** : elle permet l'évolution ;
- **Coopération** entre deux couches de même niveau *n* par le protocole de communication, vu comme une communication directe de la couche *n* de A vers la couche *n* de B.

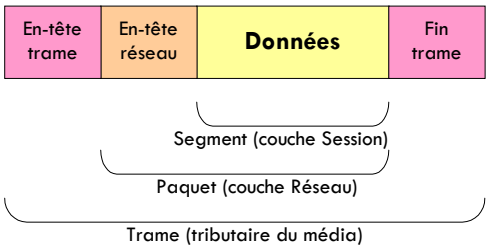
Chaque couche possède les fonctionnalités suivantes :

- Une interface avec la couche supérieure,
- L'implémentation des fonctionnalités propres à la couche,
- Une interface avec la couche inférieure.

Les 7 couches



Encapsulage des données



L'**encapsulation** consiste à ajouter des informations nécessaires à l'acheminement des données. La décapsulation est l'opération inverse, qui consiste à récupérer simplement les données.

L'encapsulation permet d'indiquer dans quel format sont les données (segment) puis la destination des données et éventuellement la route (en-tête de réseau) pour former un paquet). Si l'information ne peut être envoyée en une seule fois, elle est découpée en morceaux appelés trames, trames qui sont identifiées et ordonnées.

Les couches basses

Les **couches basses** gèrent l'hétérogénéité des câbles, des techniques d'accès au support, des routages à effectuer, etc. Leurs services sont essentiels : elles assurent la **gestion de la connexion** ainsi que le **transfert** de l'information. C'est à ce niveau que sont situés les répétitions du signal, les concentrateurs (*hubs*) et les commutateurs (*switches*).

> Couche n°1 : la couche Physique

Cette couche est à distinguer du support, qui ne fait pas partie de la pile OSI. Cette couche gère la façon d'envoyer les données. En fait, elle se contente de traduire le signal logique émis par les couches supérieures en signal numérique (électrique ou optique). Les données sont ainsi transmises sous forme de trains de bits de plusieurs façons. Par exemple, la transmission à distance nécessite une modulation du signal numérique.

> Couche n°2 : la couche Liaison

La couche Liaison découpe en « trames » l'information des couches supérieures puis la transmet à la couche Physique. Elle ajoute à chaque trame une détection d'erreurs, avec des parités, des codes détecteurs d'erreurs, des codes correcteurs, ainsi que des numéros de séquence (numéros de trame). La couche Liaison est découpée en deux parties : le contrôle de l'accès au média (MAC) qui assure l'indépendance au média, et le contrôle du lien logique (LLC) qui offre des services complémentaires tels que la fiabilité, la détection et la correction d'erreurs. C'est cette couche qui est la plus haute des deux. Cette sous-couche offre de plus quelques services aux deux communicants tels que : l'établissement et la libération de la connexion, ainsi que le transfert de données avec ou sans accusé de réception.

> Couche n°3 : la couche Réseau

Elle assure la constitution de sous-réseaux et l'interconnexion de ces sous-réseaux, en gérant notamment les problèmes de blocage et de routage. Ses fonctionnalités sont l'adressage logique et le routage à travers une série de relais dans les couches basses. La couche Réseau commande l'établissement des connexions, mais laisse le soin à la couche LLC d'établir physiquement le contact. Attention à ne pas confondre adresse physique et adresse logique : l'adresse physique est unique pour une machine donnée, alors que l'adresse logique dépend du lieu, de la situation, et de nombreux autres paramètres. Elle peut être amenée à changer à chaque connexion.

Les couches hautes

Le rôle des **couches hautes** est de fournir des **services** à l'utilisateur, de rendre l'utilisateur indépendant des échanges de contrôle, et de masquer l'hétérogénéité : par exemple, transférer un fichier quels que soient les machines et les réseaux employés.

> Couche n°4 : la couche Transport

Cette couche fait en sorte que le transfert de l'information soit fiable. Elle assure la communication de bout à bout ainsi que le multiplexage.

> Couche n°5 : la couche Session

Elle gère avant tout la synchronisation des machines. Plus précisément, la couche Session gère l'établissement de la communication, le dialogue, qu'il soit bidirectionnel alterné ou simultané, mais aussi la reprise après une interruption éventuelle du transfert.

> Couche n°6 : la couche Présentation

Cette couche permet la représentation et la compréhension des données. Elle identifie les différents types de données et les traite différemment de manière adaptée, elle convertit les alphabets, et c'est également cette couche qui offre des possibilités de cryptage, de compression et d'authentification des données.

> Couche n°7 : la couche Application

C'est la seule couche qui est en contact avec l'utilisateur. Elle est composée de 'briques applicatives'. Chacune de ces briques réunit un ensemble indissociable de fonctionnalités : terminal virtuel, messagerie, électronique, processus de communication. La couche Application contient toute la richesse applicative du modèle.

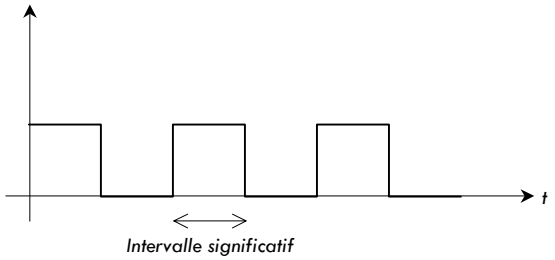
L'information et son codage

> La couche Physique

Définitions

L'unité de base des données informatiques est le bit. Un bit peut prendre deux états : 0 ou 1. Le but de la **couche physique** est de transformer cette information logique en information électrique bien physique.

Voici un exemple de signal numérique :



Cette information numérique possède deux niveaux de quantification du signal : 0V et 5V.

On pourrait aussi imaginer avoir un signal avec 4 niveaux de quantification, par exemple : -12V, -5V, 5V et 12V.

L'**intervalle significatif** est le plus petit intervalle au cours duquel le signal reste constant.

La **rapidité de modulation** (***R***) est le nombre d'intervalles significatifs par seconde. Elle mesure l'échantillonnage du signal, et est exprimée en bauds.

La **valence d'un signal** (***V***) est le nombre de niveaux de quantification (de valeurs) transportés dans un intervalle significatif.

Enfin, le **débit** (***D***) est la quantité d'informations binaires par secondes transportée par le signal. Elle se mesure en bits/s.

Relations liant *D*, *V* et *R*

Les niveaux de quantification ou valence et la quantité d'informations binaires transportées par intervalle significatif (notée *n*) sont liés par la relation suivante :

$$n = \log_2 V \Rightarrow V = 2^n$$

Débit, rapidité de modulation et valence sont liés par les relations :

$$D = R.n \quad \text{d'où :} \quad D = R.\log_2 V$$

Il découle de ces relations que le débit dépend de la rapidité de modulation (nombre d'intervalles de quantification par unité de temps) et de la valence (nombre de niveaux caractéristiques identifiables sur le signal). De même, la rapidité de modulation dépend de l'étendue de la bande de fréquence exploitable ; et la valence dépend de la qualité de la liaison S/B.

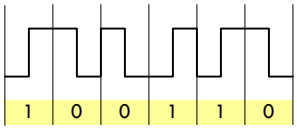
Dès lors, on distingue deux types de codage :

- Le codage par bande de base : toute la bande passante est disponible, et à chaque état correspond un niveau de tension.
- Le codage par modulation : l'information est transmise par la modulation d'une tension sinusoïdale de référence appelée la porteuse. Dans ce contexte, un état est codé par une modification de cette porteuse.

Codage par Bande de Base

Le **codage par Bande de Base** consiste en la transmission d'un potentiel et de son opposé. Les bits sont codés par les transitions plutôt que par niveaux, et ce pour éviter les déperditions dues à la baisse de potentiel. Malheureusement, cette transmission est limitée à de courtes distances : de quelques centaines de mètres à quelques kilomètres. De plus les signaux ne peuvent être superposés : il y a un signal à la fois sur le média considéré.

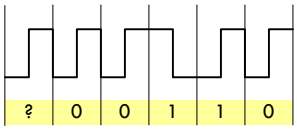
> Codage Manchester



Ce codage impose toujours une transition par état. Le sens de la transition donne la valeur de l'état (ici, par exemple : vers le haut : 1, vers le bas : 0).

On remarque que le débit est de moitié la rapidité de modulation, ce qui interdit des fréquences très élevées. Par contre le transfert est d'une fiabilité vraiment remarquable.

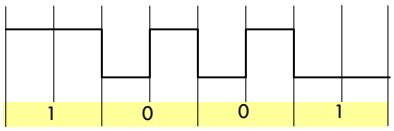
> Codage Manchester Différentiel (Réseaux Ethernet)



Ce codage impose également une transition par état, mais il est basé sur le changement de polarité (changement en début d'intervalle : 0, pas de changement : 1).

Ici aussi, le débit est moitié moindre. La fiabilité est bonne, mais pas autant que pour le codage Manchester. Par contre, on peut avoir des vitesses beaucoup plus élevées.

> Codage nB/mB



On a représenté ici un codage 1B/2B, c'est-à-dire qu'un 1 est représenté par deux intervalles de même niveau, et le 0 est représenté par un motif figé qui se répète.

De manière générale, un mot de *n* bits est codé par blocs de *m* intervalles (plus d'explications en TP, certainement).

Codage par Modulation

Dans le **codage par modulation**, l'information est constituée de modifications d'une porteuse sinusoïdale. Ce codage permet la transmission sur de longues distances. Les modulations de la porteuse peuvent être de trois sortes : modulation d'amplitude (mauvaise qualité), modulation de fréquence, et modulation de phase (beaucoup plus sûre).

Ces modulations peuvent être combinées entre elles pour aboutir à des codages plus complexes. Ainsi, si l'on combine amplitude et fréquence, ou phase et amplitude, on obtiendra un débit double et une valence de 4.

L'accès au média

> La couche Liaison

Identification

Un des problèmes que doit relever la **couche Liaison** est la question de l'identification des ordinateurs reliés au média. Dans un réseau, tous les ordinateurs doivent avoir un nom pour pouvoir discuter. Ce nom doit respecter des normes bien précises.

L'**adresse physique**, aussi appelée **adresse MAC**, est l'identifiant unique de l'ordinateur. Elle est en quelque sorte gravée sur la carte réseau. L'adresse physique occupe 6 octets, décomposés comme suit :

Identifiant unique d'organisation (O.U.I) : nom du constructeur	ID attribué par le fournisseur (carte réseau, interfaces) : numéro de série
24 bits (6 chiffres hex.)	**24 bits (6 chiffres hex.)**
Ex : *00 60 2F* *Cisco*	Ex : *3A 07 BC* *Unité particulière*

La dynamique de l'échange dans un réseau se résume par les étapes suivantes :

- Lorsqu'une source envoie des données dans un réseau, ces données transportent l'adresse MAC de leur destination.
- La carte réseau de chaque unité du réseau vérifie si son adresse MAC correspond à l'adresse physique de destination transportée par le paquet.
- S'il n'y a pas de correspondance, la carte réseau ignore le paquet, qui poursuit son chemin. S'il y a correspondance, cependant, la carte réseau effectue une copie du paquet de données, qu'elle place dans l'ordinateur, au niveau de la couche Liaison. Le paquet de données original poursuit son chemin dans le réseau.

La trame

Le **verrouillage de trame** aide à obtenir de l'information essentielle qu'il n'était pas possible d'obtenir uniquement avec les trains binaires :

- Quel ordinateur communique avec quel autre ;
- Quand la communication entre les ordinateurs individuels commence et quand elle se termine ;
- Quelles erreurs se sont produites pendent la communication ;
- A qui le tour de 'parler' dans une 'conversation'.

Le verrouillage de trame est le processus d'encapsulage de niveau 2.

Une **trame** possède une structure générique :

A	B	C	D	E	F
Champ de début de trame	Champ d'adresse	Champ de type et de longueur	Champ de données	Champ de FCS (Frame Check Sequence)	Champ de fin de trame

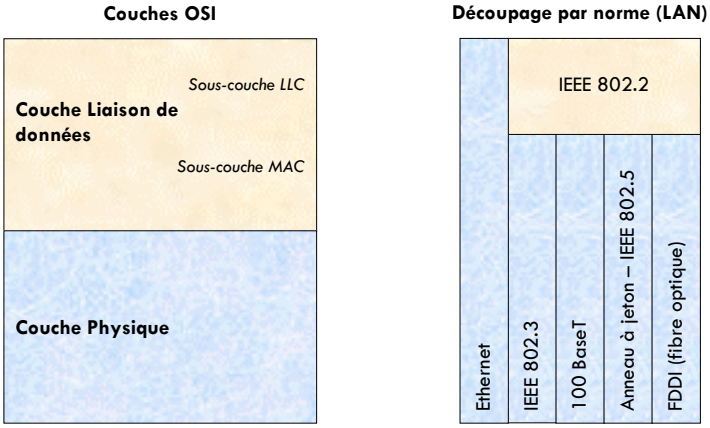
Dès la couche 2, on se préoccupe du type de réseau :

- A connexion **directe** : média partagé (accès multiple) ou média étendu (accès multiple avec unité de réseautage de couche 1), ou encore point à point ;
- A connexion **indirecte** : commutation de circuits ou commutation de paquets.

Les réseaux locaux ont ceci de particulier que tout le monde est relié par le même média, et que **tout le monde reçoit les informations**. Chacun trie ensuite et conserve ce qui lui est dédié.

Normes de l'IEEE

Les normes de l'IEEE (*Institute of Electric and Electronic Engineering*) ne suivent pas totalement le découpage des deux premières couches, comme le montre le schéma suivant :



2 méthodes d'accès

Deux méthodes d'accès au support de transmission sont utilisées :

- Le **CSMA/CD**, qui est un accès aléatoire (tout le monde 'parle' en même temps),
- L'**anneau à jeton**, ou **token ring**, qui est déterministe et supervisé (la 'parole' est donnée à celui qui a le jeton).

Ces méthodes se placent dans la sous-couche MAC de la couche liaison.

L'anneau à jeton

Un réseau fonctionnant par la méthode de l'**anneau à jeton** est en circuit fermé, c'est-à-dire que les données empruntent une voie circulaire, en sens unique. On fait circuler un 'jeton', une trame particulière qui indique que la voie est libre. Une station qui veut émettre accroche ses données au jeton, s'il est libre.

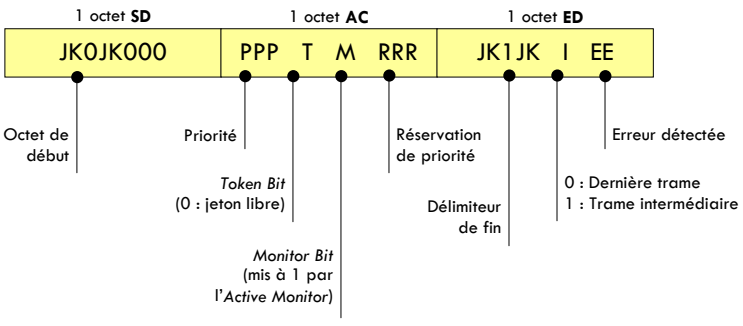
Pour qu'une trame d'information arrive à destination, elle doit être recopiée de station en station, on peut ainsi faire du multicast. Le destinataire garde une copie, mais n'arrête pas la retransmission. Quand la trame a fait un tour complet, l'émetteur la retire lui-même de l'anneau, et réemet le jeton libre.

Il découle de ces principes généraux un certain nombre de contraintes. Chaque station est responsable de ses trames, elle doit les retirer ! Une station doit renvoyer le jeton libre après avoir retirer sa dernière trame. Techniquement, l'option *Early Token Release* permet de réemettre le jeton après avoir fini l'émission (donc il y a deux jetons pendant un temps). Il doit y avoir une horloge qui synchronise les liens. Enfin, il faut un temps maximal pour le jeton de parcours du réseau, et un temps maximum de possession du jeton.

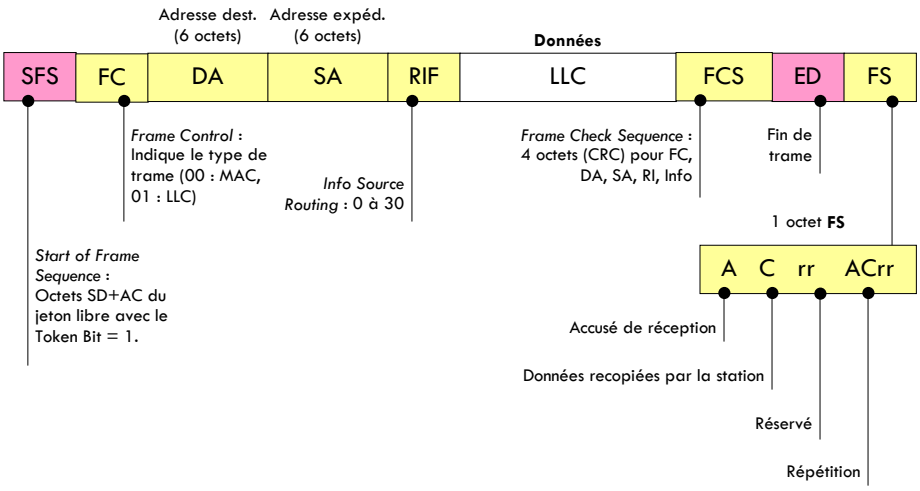
Une station a un statut particulier, celui de maître. On l'appelle l'*Active Monitor* (AM). C'est elle qui gère l'horloge, qui vérifie la présence continue d'un et d'un seul jeton, qui gère l'insertion de stations, et qui prévient régulièrement les autres de sa présence, par l'émission de trams *Active Monitor Present* toutes les 7 secondes.

Si une station considère qu'il n'y a plus d'*Active Monitor*, elle émet des trames *Claim Monitor*. Si elle reçoit une trame provenant d'une adresse de priorité supérieure, elle recopie celle-ci, sinon elle la remplace par la sienne. Si une station reçoit ses propres trames, elle se proclame Active Monitor et nettoie alors le réseau par un *Ring Purge* (*Reset*).

Jeton libre :



Jeton avec trame :



L'acquittement se fait au retour du message à l'émetteur :

- Si A=0 et C=0, il n'y a aucun destinataire.
- Si A=1 et C=0, il y a au moins un destinataire, mais il n'a pas recopié la donnée.
- Si A=1 et C=1, tout s'est bien passé !

Une station qui voit passer un jeton libre le capture et lui ajoute ses trames, à condition que sa priorité PPP soit supérieure ou égale à celle du jeton. Dans le cas contraire, elle indique dans RRR sa propre priorité. Si une autre station réserve à son tour, la première n'aura plus qu'à recommencer. Lorsque la station retenue a terminé son émission, elle réemet le jeton libre avec la priorité PPP à la valeur RRR. Une station qui augmente la valeur de PPP mémorise ma valeur initiale et est chargée, dès que possible, de réemettre un jeton du niveau de priorité initial.

Les trames sont surveillées : si une station qui a émis une trame disparaît, le jeton n'est plus libéré. La station moniteur force à chaque passage de jeton chargé le bit M du champ AC à 1, et si elle revoie passer une telle trame, elle la supprime. Les priorités sont également surveillées : le blocage peut être dû à une priorité trop élevée. La résolution se fait comme précédemment, grâce à l'*Active Monitor*. Enfin, lorsque le jeton se perd, la station moniteur le rétablit.

L'accès aléatoire : CSMA/CD

La méthode **CSMA/CD** (*Carrier Sense Multiple Access / Colision Detection*) fonctionne sur un principe complètement précédent. Chaque machine regarde si la voie est libre, par détection de la porteuse. Si c'est le cas, elle émet, sinon elle recommence l'étape précédente. Si jamais une collision survient, elle attend un *certain* temps, puis recommence (ou abandonne…).

Cette méthode est la méthode d'accès des produits Ethernet. C'est pourquoi on confond souvent les dénominations CSMA/CD et Ethernet.

Les normes Ethernet et IEEE 802.3 précisent des technologies semblables ; les deux décrivent des réseaux à accès CSMA/CD. Les différences qui existent entre les réseaux Ethernet et IEEE 802.3 sont subtiles. Les spécifications de réseau local Ethernet ou IEEE 802.3 sont mises en œuvre par du matériel informatique. Habituellement, la manifestation physique de ces protocoles est une carte d'interface située dans un ordinateur hôte.

Une collision se produit lorsque deux machines 'parlent' en même temps. On remarque qu'une collision a eu lieu grâce au parasitage mutuel des deux signaux. Le signal ne correspond alors ni à un 1, ni à un 0.

Les stations qui détectent une collision la renforcent en envoyant un *jam*. Si la station émettrice est encore en train d'émettre lorsqu'elle reçoit le *jam*, elle est informée de la collision. Si elle a fini d'émettre, elle ne sait pas si cette collision concerne sa trame.

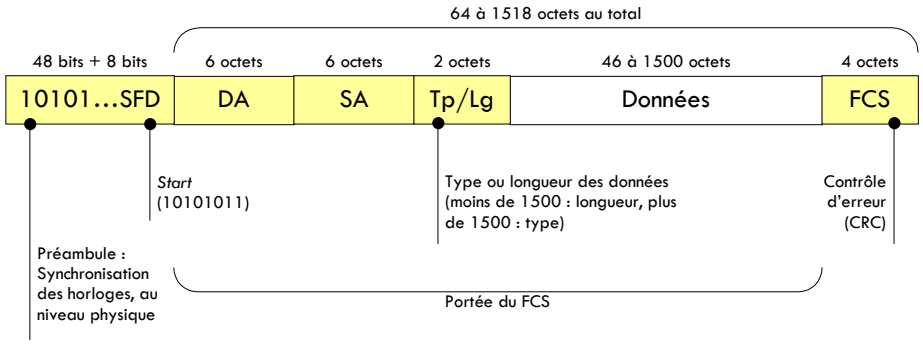
On rajoute donc un paramètre supplémentaire pour la gestion des collisions : le *Round Trip Delay*, qui est le temps de propagation aller-retour dans le réseau (on parle alors de diamètre du réseau). Ainsi, s'il la trame prend au moins ce temps pour être transmise, lors d'une collision, la station sera toujours en train d'émettre… Cela donne une taille minimum de trame de 64 octets.

Les grandeurs caractéristiques d'un réseau CSMA/CD sont :

- Le temps minimal d'émission ou Slot Time (ST) exprimé en secondes ;
- Le débit nominal du réseau ou capacité C du réseau, exprimée en bits.s-1 ;
- La longueur maximale entre deux stations, appelée diamètre D, mesurée en mètres ;
- La vitesse de propagation VP, en m.s-1 ;
- La fenêtre de vulnérabilité, temps que met une trame pour parcourir toute la longueur du réseau, exprimée en secondes.

Après une collision, les stations attendent un *certain* temps avant de recommencer à émettre, et ce *certain* temps est calculé à partir d'un algorithme d'attente aléatoire, le BEB (*Binary Exponential Backoff*). Techniquement, il assure une retransmission selon une loi exponentielle binaire. Ce temps est un multiple du *Slot Time*, et la fenêtre de tirage aléatoire augmente en fonction du nombre d'essais tentés pour émettre une trame donnée, c'est-à-dire que le temps x est tiré dans l'intervalle $[0,2^n[$, n étant le nombre d'essais pour la trame en cours. On attend alors le temps x multiplié par ST. A partir du 10e essai, la fenêtre reste de taille constante, et au bout du 16e essai, la transmission échoue. Bien entendu, cet algorithme est exécuté indépendamment sur chaque machine, et moins il y a de temps d'attente égaux sur différentes machines, mieux le réseau fonctionne.

La trame :



Bilan

Lorsque le réseau est peu chargé, on obtient un très bon rendement en CSMA/CD, par contre, le rendement est faible en anneau à jeton, car plus le nombre de communications est grand, et plus le rendement est faible.

Lorsque le réseau est chargé, on observe une limite critique en CSMA/CD, alors que le rendement approche 1 en anneau à jeton !

L'anneau à jeton fonctionne seulement en 4 ou 16 MBits. En ce qui concerne les réseaux Ethernet, il existe plusieurs vitesses :

- 10 MBits : *Ethernet*,
- 100 MBits : *Fast Ethernet*,
- 1000 MBits : *Giga Ethernet*.

Adressage et routage

> La couche Réseau

Fonctionnalités

La **couche Réseau** offre des fonctionnalités :

- d'**adressage** : une adresse logique unique et distincte de l'adresse physique, appelée aussi adresse MAC ;
- de **routage** : avec des algorithme de routage permettant de trouver le chemin le plus court, et des fonctions de routage permettant d'aiguiller les trames ;
- de gestion du trafic entre les réseaux : un **filtrage** pour la sécurité et les performances.

Adressage

Chaque ordinateur possède une adresse qui est une valeur linéaire d'identification par le constructeur : c'est l'adresse physique ou adresse MAC, car elle utilisée dans les trames de la couche 2.

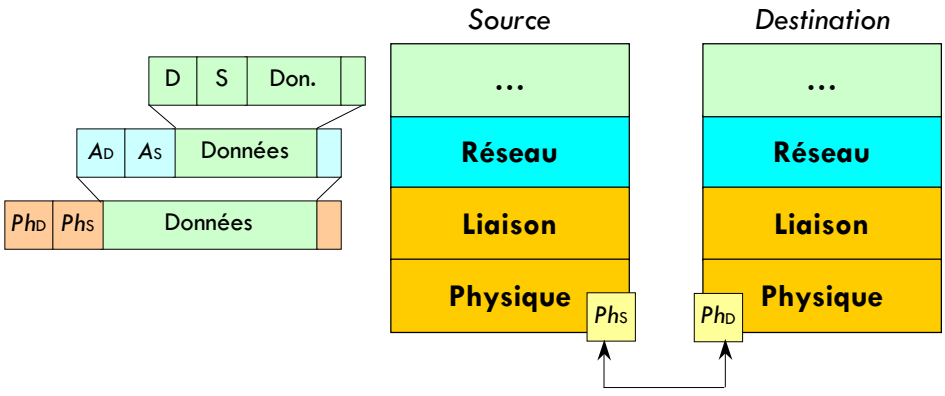
A mesure qu'augmente le nombre de réseau, un adressage hiérarchique s'impose pour permettre la localisation de l'ordinateur. Cet adressage est un peu comme un numéro de téléphone par rapport au numéro d'identification de l'appareil. On l'appelle l'adresse logique.

Ainsi, un même ordinateur qui a une seule adresse physique peut se retrouver à un moment donné à une adresse logique différente.

Rappel : L'adresse physique est composée de 3 octets de code fournisseur, et de 3 octets indiquant le numéro de série. Cette adresse est inscrite dans la mémoire morte de la carte réseau. Il est donc impossible de la modifier.

L'adresse logique est composée également de deux parties : l'adresse de réseau, qui identifie l'emplacement du réseau, et est utilisée par le routeur ; et l'adresse d'hôte, qui identifie un port ou un dispositif particulier du réseau.

Le schéma suivant décrit l'encapsulation des adresses à travers les couches successives. A_D et A_S sont les adresses logiques de la source et de la destination, et Ph_D et Ph_S sont les adresses physiques.



Le Routeur

En plus de l'adressage, la couche Réseau offre des fonctions de recherche de meilleure chemin dans l'interréseau. Elle implémente pour cela des algorithmes de calcul de route.

La fonction principale de la couche réseau est de router les « paquets » à travers les réseaux. L'algorithme de routage est le logiciel qui a la responsabilité de décider sur quelle ligne de sortie un paquet entrant doit être retransmis pour atteindre sa destination. Pour cela, on doit établir les tables de routage. Un algorithme de routage se doit de posséder les propriétés suivantes :

- Rapidité en temps, en distance ou en sauts,
- Efficacité en terme de débit,
- Convergence et stabilité,
- Qu'il soit équitable vis à vis des usagers.

Un algorithme de routage peut être **statique**, c'est-à-dire mis en place par l'administrateur, ou calculé une seule fois. Ce procédé est peu coûteux et s'adapte bien à des interconnexions stables et « lourdes ». Il peut au contraire être **adaptatif**, ce qui veut dire qu'il est remis à jour régulièrement. Cela permet de prendre en compte des événements récents, comme la congestion du réseau, un arrêt, des modifications de coûts, des encombrements… Par contre ce procédé est complexe et coûteux en bande passante, du fait des échanges inter-routeurs constants.

Ces algorithmes peuvent être centralisés ou répartis.

Il existe différents **algorithmes statiques** de « plus court chemin », citons par exemple Dijkstra. Cet algorithme considère le réseau comme un graphe. Les arcs peuvent ne pas être pondérés, on parlera alors de sauts, ou bien ils peuvent être pondérés par une métrique : distance, attente, etc.

Pour trouver une route entre deux routeurs, l'algorithme se contente de trouver le plus court chemin entre eux et le mémorise.

Dans le **routage à vecteur de distance** (RIP), chaque routeur dispose d'une table de routage précisant pour chaque destinataire la meilleure distance connue et la ligne par où l'atteindre – le « vecteur de distance ».

Chaque routeur échange périodiquement ses informations avec ses voisins et met à jour ses tables. La métrique peut être en sauts, en nombre de paquets en file d'attente ou en temps d'acheminement, qui est alors calculée à l 'aide d 'un paquet spécial « Echo ». Les algorithmes à vecteur de distance convergent mais lentement. D'autre part, les temps de modifications sont égaux au nombre de sauts.

Le routage par information d'état de lien (exemple : OSPF), se compose de plusieurs étapes :

1. Découvrir les voisins et leur adresse, en envoyant un paquet spécial « Hello » sur chaque ligne physique. Les voisins répondent en envoyant leur nom et leur adresse.
2. Mesurer le temps d'acheminement vers chacun d'eux par un paquet « Echo ».
3. Construire un paquet « d'information d'état de liens », qui est composé des informations, du numéro de séquence, et de l'âge.
4. L'envoyer à tous les routeurs (unités de couche 3) du sous-réseau.
5. Ces routeurs utilisent 'l'inondation' en retransmettant ces paquets d'information d'état de lien, à condition que le numéro de séquence ne soit pas déjà passé.
6. Calculer le plus court chemin vers tous les autres routeurs, ce qui est possible puisque on possède le graphe complet du réseau. L'âge d'un paquet est décrémenté toutes les secondes et le paquet est détruit lorsqu'il atteint 0.

Qualité de service

> La couche Transport

Objectifs

La **couche Transport** a pour objectif d'assurer la **qualité de service**, notamment en terme de fiabilité, par un contrôle d'erreur, et un contrôle de flux.

- Le **contrôle d'erreur** détecte les erreurs de transmission, qui peuvent être des pertes ou des ajouts de données erronées, ou bien un dysfonctionnement du communicant.
- Le **contrôle de flux** évite les problèmes de famine ou de congestion du réseau, améliore le rendement, et permet de réagir en cas d'erreur.

De nombreux problèmes peuvent survenir lorsqu'il n'y a pas de numérotation, et de contrôle des pertes : trames manquantes, mauvais ordre de réception, taille du tampon de réception, vitesse de lecture, etc.

Solutions

Pour résoudre ces problèmes, on pourrait commencer par numéroter les trames, ce qui permettrait de retrouver l'ordre original, mais ne permettrait pas de détecter la perte des dernières trames. Dès lors, il peut être utile d'envoyer un accusé de réception, ou encore d'envoyer plusieurs fois les trames.

On peut envisager deux protocoles, l'un très simple, avec accusé de réception après chaque trame. Ce procédé est archaïque en plus d'être lent, car il faut à chaque fois compter un aller-retour entre chaque envoi. L'autre protocole est plus complexe, utilisant des fenêtres d'anticipation et d'acquittement, permettant de ne pas attendre systématiquement les accusés de réception.

Protocole simple

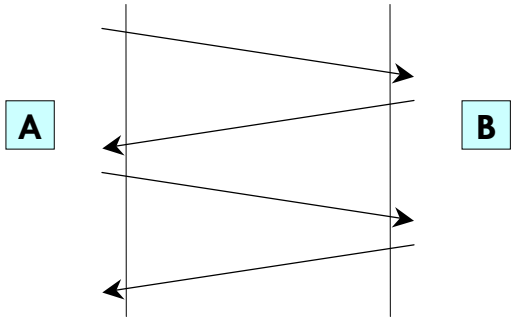
Le protocole simple met en œuvre tout d'abord une détection d'erreur basée sur le **CRC** (*Cyclic Redondancy Check*) des données envoyées. Il utilise un mécanisme d'**acquittement** de telle sorte que la transmission s'effectue de la manière suivante :

a) La machine A émet sa trame de données 0 (DATA 0),

b) La machine B reçoit la trame et la vérifie,

c) La machine B émet une trame d'acquittement (ACK 0),

d) La machine A reçoit la trame ACK 0, puis émet sa deuxième trame…



Ce protocole présente avant tout l'avantage d'être simple ! Cependant, il est peu efficace en raison des nombreuses attentes engendrées par le mécanisme d'acquittement systématique. Il existe réellement, par exemple dans les systèmes *Kermit*.

Il ne gère pas le flux, pour cela il faut ajouter des trames de contrôle Xon, Xoff ; par contre il peut gérer des erreurs par une trame de non acquittement (NACK), et les pertes par temporisation. Une autre solution serait la double émission…

Contrôle de flux par fenêtre

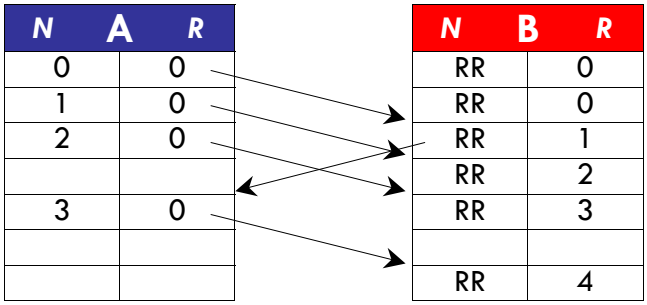
Le second protocole met en jeu la notion de **fenêtre**. Considérons une fenêtre de taille *n* qui se déplacera le long de la suite ordonnée des trames de données de la station émettrice A. A va émettre les *n* trames de la fenêtre les unes après les autres, sans attendre d'acquittement. Pour chaque trame reçue sans erreur, B émet les trames d'acquittement correspondantes. Lorsque A reçoit la trame ACK de la première trame de la fenêtre, elle décale la fenêtre d'une trame et émet donc la n+1e trame.

Dans ce contexte, les trames peuvent être de deux types : soit ce sont des trames de contrôle S, soit des trames de données I (Data). Chaque trame I comporte, en plus des données, deux compteurs : *N*, qui est le numéro de la trame ; et *R*, qui est le nombre de trames bien reçues. Dans ce cadre, le transfert est bidirectionnel.

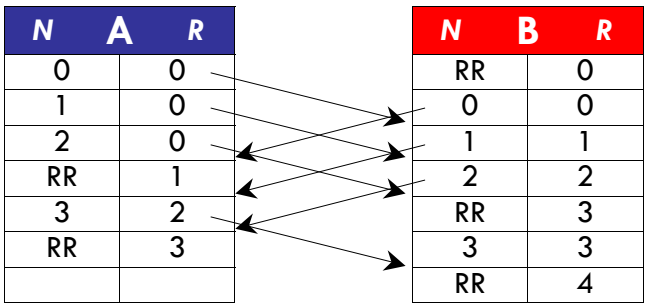
Ces deux compteurs sont modulo *i*, le plus souvent 8. De fait, *N* commence impérativement à 0. Une trame I reçue par A en provenance de B contient des données de l'échange B→A, et un compteur qui accuse réception de toutes les trames I de l'échange A→B, portant un numéro inférieur à *R*.

Les trames S contiennent, en plus du compteur *R* (comme précédemment), un indicateur de type, qui peut être : **RR** (Receive ready), **RNR** (Receive not ready), **REJ** (REJet), **SREJ** (REJet Sélectif), etc.

Echange unidirectionnel, fenêtre de taille 3



Echange bidirectionnel, fenêtre de taille 3



En cas d'erreur, deux procédures peuvent être utilisées.

- Dans le cadre de la reprise sur erreur par **rejet classique**, on envoie une trame REJ, qui porte le compteur *R*, qui accuse réception des *R* trames inférieures et qui demande la reprise à partir de la trame *R*. Ce procédé provoque la réémission de toute la fenêtre, ce qui peut être une perte de temps, et engendrer une occupation de la bande passante. Par contre, c'est intéressant si il y a plusieurs erreurs successives ou si les fenêtres sont petites.

- Lors de la reprise sur erreur par **rejet sélectif**, on envoie une trame SREJ, qui accuse réception des *R* trames inférieures, et demande la réémission uniquement de la trame *R*. Contrairement au rejet classique, on conserve les trames bien reçues (c'est-à-dire de numéro supérieur à *R*). On gagne de surcroît en efficacité s'il y a peu d'erreurs, ou si on utilise de grandes fenêtres.

Les équipements d'interconnexion

Le répéteur

Unité de couche 1

Le **répéteur** amplifie et re-synchronise les signaux présents sur la ligne. Il n'agit qu'au niveau du bit, et ne regarde aucune autre information, d'où son appartenance à la couche physique. Il permet de raccorder plusieurs segments éloignés d'un réseau local (jusqu'à plusieurs centaines de mètre, pas davantage), mais ne permet pas de dépasser certains paramètres de délai maximum.

Le concentrateur ou *hub*

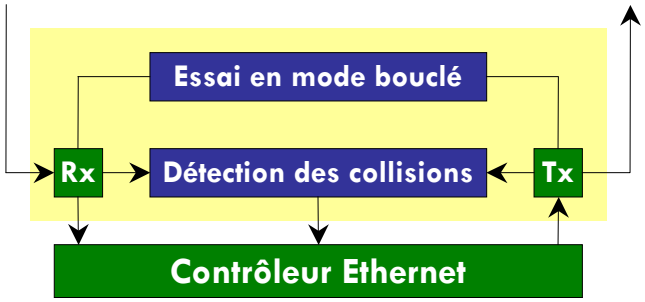
Unité de couche 1

Le **concentrateur** prend un signal entrant et le répète sur chaque port. On peut donc considérer que c'est un répéteur multiport. Il permet de connecter simplement un grand nombre d'ordinateurs du réseau, et confère au réseau une architecture matérielle en étoile, tout en correspondant à une architecture logicielle en bus, car l'information est transmise à tous, directement, sans aucun filtrage.

Le port

Unité de couche 2

Le **port**, appelé aussi carte réseau, fait le lien avec le média du réseau par ses connecteurs (RJ45, AUI, BNC, etc.). Il convertit le signal électrique du média (du câble) en bits. Pour cela, il reconnaît une adresse qui est l'adresse physique de l'ordinateur dans le réseau. C'est également lui qui organise la mise en trame et qui contrôle l'accès au média.



Le contrôleur physique Ethernet fournit plusieurs circuits : Rx (réception), Tx (transmission), et détection de collision.

Le pont

Unité de couche 2

Le **pont** filtre le trafic sur un réseau en regardant l'adresse physique de destination de la trame. Il permet ainsi de segmenter un réseau en sous-réseaux. En éliminant le trafic inutile, les ponts réduisent les congestions et les collisions. Un pont crée, pour chacun des segments qu'il contrôle, une table de toutes les adresses physiques situées sur ce segment ; il possède donc une table par segment. Quand un trame arrive d'un segment, le pont compare son adresse physique de destination à celles contenues dans la table de ce segment.

Si l'adresse de destination concerne un ordinateur du même segment de réseau que l'ordinateur source, le pont achemine cette trame sur ce segment, mais ne la propage pas aux autres segments du réseau. Sinon, il achemine la trame à tous les autres segments. Donc, si une trame n'est pas locale à un segment, le pont l'émet sur tous les autres segments du réseau, il est comme un répéteur. Ceci n'est pas très efficace dans les réseaux ayant beaucoup de segments. D'autre part, de manière générale, le pont augmente la latence du réseau.

Le commutateur ou *switch*

Unité de couche 2

Le **commutateur** est un pont multiport : chaque port du commutateur est un pont. Contrairement au pont, si l'information n'est pas locale à un segment, le commutateur cherche le bon segment grâce à ses tables. Il permet d'atténuer la congestion des réseaux. Cependant, il ne travaille que sur les adresses physiques, donc il ne peut que rester local, puisqu'il lui faut connaître toutes les adresses MAC de ses ports. Il n'est donc pas adapté à Internet.

On peut parfaitement considérer qu'un commutateur est équivalent à un concentrateur ayant un pont sur chacun de ses ports. Comme un pont, un commutateur permet de constituer des domaines de collisions, et donc d'étendre le diamètre du réseau.

En réalité, le commutateur possède une table référençant les adresses physiques de toutes les machines connectées en fonction de leur port. Cette table n'est pas statique, mais dynamique. En effet, les adresses sont découvertes au fur et à mesure.

Son intérêt par rapport au concentrateur est évident. Il permet d'éliminer l'effet des collisions grâce à la microsegmentation. Il offre également une latence peu élevée et de hauts débits d'acheminement des trames, et ce à chaque port d'interface.

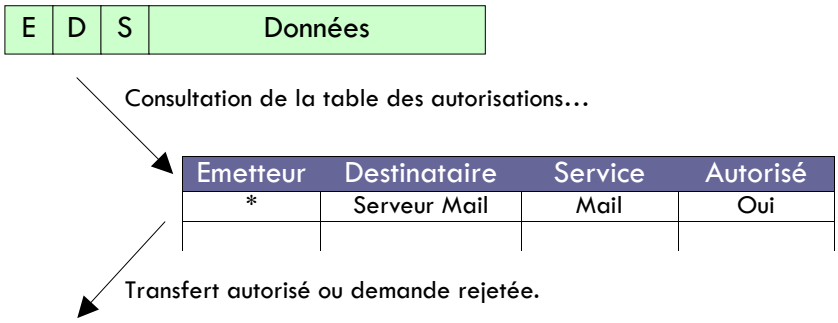
Le routeur

Unité de couche 3

Le **routeur** est un dispositif qui recherche la meilleure route pour la transmission des données. Il est utilisé pour la connexion de réseaux hétérogènes, ou à travers des liaisons à distance. Alors que le pont travaille avec les adresses physiques, donc de niveau 2, le routeur utilise les adresses logiques (de niveau 3) qui sont indépendantes de la nature du réseau. A cause de cela, il est plus lent que le pont ou le commutateur, mais il peut aussi faire du filtrage.

Le routeur est plus facile à gérer et offre des fonctionnalités accrues ainsi que de multiples voies actives. A l'instar du commutateur, il permet de constituer des domaines de collision plus petits.

Filtrage



Récapitulatif historique

Le besoin d'accroître la distance entre les ordinateurs a mené au développement du **répéteur** (un concept emprunté à d'autres technologies de télécommunications).

Le besoin d'une connectivité accrue dans un groupe de travail a mené au **concentrateur**. En tant qu'unités de couche 1, les répéteurs et les concentrateurs n'examinent pas l'information qui passe par eux. Les limites du concentrateur, soit le fait qu'il ne filtre pas du tout le trafic réseau, sont devenues apparentes avec l'augmentation des PC connectés aux concentrateurs se partageant la largeur de bande.

Le **pont** a alors été introduit comme moyen de filtrer le trafic réseau en trafic local et non local ; comme ce filtrage est accompli au moyen des adresses de couche physique, le pont est considéré comme une unité de couche 2. Les ponts ont été introduits pour segmenter les réseaux en plus petits domaines de collision.

L'idée de base des ponts a été combinée à la connectivité (densité de ports) des concentrateurs, donnant ainsi naissance au **commutateur**, un pont multiport. Le commutateur, qui est aussi une unité de couche 2 dont les décisions d'acheminement reposent sur les adresses physiques, offre une grande densité de ports (connectivité) et une largeur de bande spécialisée entre deux PC en communication.

Avec la croissance des réseaux, la diversité de plates-formes, de protocoles et de médias, la distance géographique entre les ordinateurs, le nombre d'ordinateurs désirant communiquer, tous ces éléments ont mené au développement du **routeur** – une unité de couche 3 – qui sélectionne la meilleure route et prend des décisions de commutation fondées sur les adresses logiques des réseaux.

Actuellement, les routeurs sont devenus tellement développés que les algorithmes de calcul de meilleur chemin sont devenus lourds et lents, du fait de l'augmentation des machines. Ils passent finalement plus de temps à calculer les routes qu'à transmettre l'information. La tendance est plutôt de passer à une 'commutation de niveau 3', en utilisant des commutateurs, équipements beaucoup plus rapides que les routeurs, mais agissant au niveau de l'adresse logique. Mais c'est encore le futur...

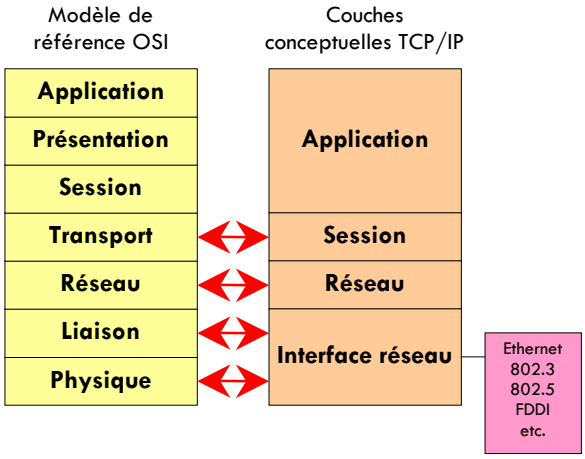
Le protocole TCP/IP

TCP/IP et OSI

TCP/IP (Transmission Control Protocol / Internet Protocol) est un ensemble de conventions pour l'interconnexion des réseaux et le routage des informations au sein de ces réseaux. Cette technique a prouvé sa viabilité à grande échelle dans Internet. En fait, TCP et IP sont des protocoles de communication pour échanger des messages. Ils décrivent :

- la **structure des messages,**
- le **comportement des ordinateurs,**
- la **gestion des erreurs.**

C'est un exemple d'application de la norme OSI. Ainsi, le TCP/IP est plus détaillé que son modèle, tout en le respectant :



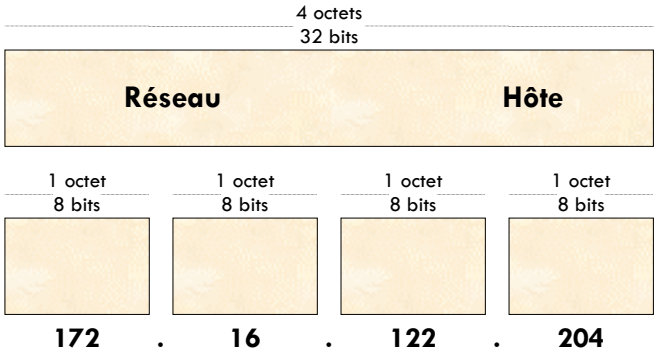
Pour chaque couche du modèle OSI, TCP/IP propose des protocoles et des services spécifiques :

- **Couche application** : RLogin, Telnet, FTP, SMTP, SNMP, http, etc.
- **Couche transport** (session) : les protocoles UDP et TCP.
- **Couche réseau** : le protocole IP assure le routage des messages, ICMP les transmet et les contrôle, ARP détermine les adresses MAC pour les adresses IP, et RARP fait l'opération inverse.

Adresses Internet

L'Internet **TCP/IP** est un réseau virtuel constitué d'un ensemble de réseaux interconnectés par des routeurs. L'adressage logique est donc l'élément essentiel pour masquer les différences physiques. L'adressage unique permet la communication entre les stations d'extrémité. La voie choisie dépend de l'emplacement, lequel est représenté par une adresse IP.

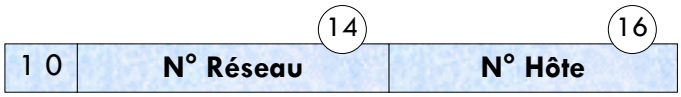
Les **adresses IP** ont le format suivant :



L'adresse IP est divisée en deux parties : une partie qui indique l'adresse réseau, et une partie qui référence l'hôte, ou la machine, dans ce réseau. La répartition de ces deux parties est variable, et conditionne le nombre de machines et de réseaux indexables :

- **Classe A** : Ce sont les adresses commençant par le bit 0, c'est-à-dire dont le premier octet est compris entre 0 et 127. La partie réseau mesure 7 bits, et la partie hôte 24 bits. Elle permettent de coder 126 réseaux de 16 millions de machines chacun.

- **Classe B** : Ce sont les adresses commençant par les bits 10, c'est-à-dire dont le premier octet est compris entre 128 et 191. La partie réseau mesure 14 bits, et la partie hôte 16 bits. Ces adresses permettent de coder environ 16000 réseaux de à peu près 65000 de machines chacun.



- **Classe C** : Ce sont les adresses commençant par les bits 110, c'est-à-dire dont le premier octet est compris entre 192 et 223. La partie réseau mesure 21 bits, et la partie hôte 8 bits. Ces adresses permettent de coder environ 2 millions réseaux de 254 machines chacun.

- **Classe D** : Ce sont les adresses restantes, c'est-à-dire dont le premier octet est supérieur ou égal à 224. Elles permettent d'utiliser le multicast, les adresses de groupe.

Certaines adresses IP sont particulières. Ainsi, une adresse IP se terminant par des 0 dans le champ 'hôte' est l'adresse du réseau complet. Une adresse IP se terminant par des 1 dans le champ 'hôte' est l'adresse de diffusion ; elle permet d'envoyer des données à toutes les unités du réseau. Enfin, une adresse IP dont le champ hôte porte l'octet 1 (*Ex* : xxx.xxx.0.1) représente très souvent la passerelle, l'entrée du réseau, qui peut être un ordinateur ou un serveur particulier dans le réseau.

Par exemple

Adresse d'hôte : 205.57.216.153

Adresse du réseau : 205.57.216.0

Adresse de diffusion : 205.57.216.255

C'est pour cette raison qu'il ne peut y avoir que 254 machines dans un réseau de classe C : parce que les hôtes 0 et 255 ont une signification particulière pour les routeurs et ne peuvent être adressés à une machine.

Masques de sous-réseaux

Dans l'état actuel des choses, le protocole IPv4 pose certains problèmes, liés d'une part à l'adressage mondial géré par un organisme centralisé, et d'autre part par la saturation imminente du nombre d'adresses disponibles.

Pour cela, on peut envisager les solutions suivantes : un sous-adressage par masque permettant d'organiser les réseaux en sous-réseaux, également la translation d'adresse (NAT) qui est pratiquée à l'IUP, et dans quelques années, l'extension de l'espace adressable avec la nouvelle norme IPv6.

En plus du gaspillage des adresses IP, la création de sous-réseaux permet la réduction de la taille des domaines de diffusion qui occasionne des congestions dans les réseaux.

Mais comment créer un champ spécifiant le sous-réseau dans l'adresse IP, sans aller à l'encontre des classes définies ci-dessus ?

Tout simplement en créant des **masques de sous-réseaux**. Tout d'abord, seul le champ 'Hôte' de l'adresse IP peut-être découpé en 'sous-réseau' + 'hôte'. Un masque de sous-réseau se rajoute à l'adresse IP et utilise le même format que celle-ci : 32 bits. La portion réseau et sous-réseau du masque contient des 1, et la portion 'hôte' des 0, ce qui permet de récupérer l'adresse du sous-réseau en par un ET logique entre l'adresse IP et son masque.

Par défaut si aucun sous-réseau n'est crée, le masque contient des 1 dans la portion 'réseau'.

Planification d'un sous-réseau de classe B

	Réseau		**Sous-rés.**	**Hôte**
Adresse hôte IP **172.16.2.120**	10101100	00010000	00000010	01111000
Masque de sous-réseau **255.255.255.0 /8**	11111111	11111111	11111111	00000000
	10101100	00010000	00000010	00000000
	172	**16**	**2**	**0**

- *Adresse de sous-réseau : 172.16.2.0 ;*
- *Adresses d'hôtes : 172.16.2.1 à 172.16.2.254 ;*
- *Adresse de diffusion : 172.16 .2.255 ;*
- *Huit bits pour la création de sous-réseaux.*

Planification d'un sous-réseau de classe C :

	Réseau			**Ss-rés.**	**Hôte**
Adresse hôte IP **192.168.5.121**	11000000	10101000	00000101	01111	001
Masque de sous-réseau **255.255.255.248 /8**	11111111	11111111	11111111	11111	000
	11001001	11011110	00000101	01111	000
	192	**168**	**5**	**120**	

- *Adresse de sous-réseau : 192.168.5.120 ;*
- *Adresses d'hôtes : 192.168.5.121 à 192.168.5.126 ;*
- *Adresse de diffusion : 192.168.5.127 ;*
- *Cinq bits pour la création de sous-réseaux.*

L'adressage de sous-réseaux permet au niveau des routeurs, de diviser le réseau en plusieurs parties, et d'élaborer de nouvelles tables de routage basées sur ces adresses, sans lister exhaustivement toutes les machines, ce qui donne un gain de temps non négligeable dans les calculs de route.

Il est important d'optimiser le choix de la répartition hôte/sous-réseau, car des blocs entiers d'adresses IP commençant par les identificateurs d'adresses de réseau ou de diffusion sont gaspillés.

En effet, pour chaque sous-réseau, il faut réserver l'adresse du sous-réseau et l'adresse de diffusion. Il faut donc trouver un compromis acceptable, comme l'illustre le tableau suivant, détaillant les différentes façons de découper en sous-réseaux un réseau de classe C :

Nombre de bits empruntés	Nombre de sous-réseaux créés	Nombre d'hôtes par sous-réseau	Nombre total d'hôtes	Pourcentage utilisé
2	2	62	124	49%
3	6	30	180	71%
4	14	14	196	77%
5	30	6	180	71%
6	62	2	124	49%

Pour chaque classe, certaines plages d'adresses IP ne sont pas attribuées. Cela permet d'augmenter le nombre de machines d'un réseau, lorsque le nombre d'adresses publiques est limité, en leur donnant une adresse IP 'virtuelle', dans le sens où, bien sûr, ces machines ne peuvent pas sortir du réseau, tout en pouvant communiquer avec les autres selon le même protocole TCP/IP.

Les plages suivantes sont disponibles pour l'adressage privé :

- 10.0.0.0 → 10.255.255.255
- 172.16.0.0 → 172.31.255.255
- 192.168.0.0 → 192.168.255.255

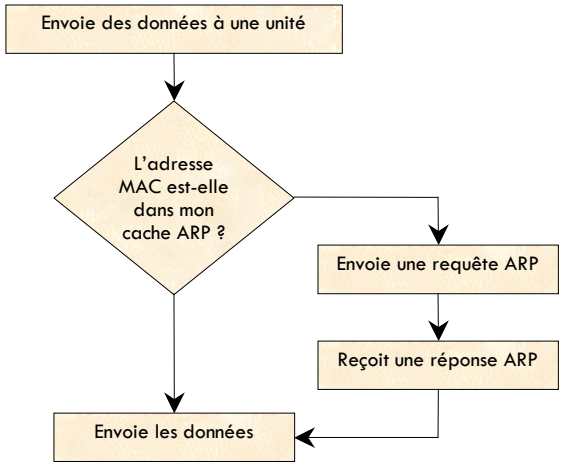
Mappage

Une adresse IP est affectée à tout ordinateur faisant partie d'un Internet TCP/IP. A terme, les données sont encapsulées dans les trames de la couche 2, qui ne connaît que les adresses physiques (MAC).

La mise en correspondance de l'adresse IP et de l'adresse MAC s'appelle le **mappage**. Cette correspondance est notée dans une table de façon statique ou dynamique. On l'appelle table ARP (protocole de résolution d'adresses).

Pour trouver l'adresse MAC d'une unité de destination, la source consulte sa table ARP. Si elle ne trouve pas cette adresse, elle déclenche une requête ARP. C'est un paquet qui est envoyé à toutes les unités du réseau, avec l'adresse IP de destination et une adresse MAC de diffusion (FF.FF.FF.FF.FF.FF).

Si l'adresse IP d'un ordinateur correspond à l'adresse IP de destination, il répond en envoyant son adresse MAC à la source, qui complète ainsi sa table ARP. Le processus ARP local de résolution d'adresse peut se résumer ainsi :



Par contre, un ordinateur ne peut pas envoyer une requête ARP à un réseau distant, car comme ce sont des adresses de diffusion, elles ne sont pas acheminées par les routeurs. Dans ce cas, on parle de Proxy ARP, et le routeur intercepte la requête, et répond avec sa propre adresse MAC, à mapper à l'adresse MAC du destinataire. Les tables ARP des routeurs s'en trouvent donc modifiées : elles présentent deux différences par rapport à d'autres tables ARP :

- Premièrement, les tables ARP des routeurs contiennent les paires d'adresses MAC/IP de plusieurs réseaux (alors qu'un hôte donné tient des tables ARP uniquement des autres hôtes de son réseau).
- Deuxièmement, la table ARP du routeur conserve la trace de l'interface par laquelle passe la voie vers une paire adresse MAC/IP donnée. Le routeur a besoin de cette information pour choisir la meilleure voie et commuter les paquets.

Protocole IP

Le protocole IP met en jeu des **datagrammes,** qui encapsulent les données dans une structure composée des éléments suivants :

- ***VER*** (4 bits) : numéro de version ;
- ***HLEN*** (4 bits) : longueur de l'en-tête, en mots de 32 bits ;
- ***Type de service*** (8 bits) : type de service de traitement du datagramme ;
- ***Longueur totale*** (16 bits) : longueur totale (en-tête et données) ;
- ***Identification*** (16 bits) : repères et fragmentation ;
- ***Drapeaux*** (3 bits) ;
- ***Décalage frag.*** (13 bits) ;
- ***TTL*** (8 bits) : durée de vie minimum ;
- ***Protocole*** (8 bits) : protocole de couche 4 ou 3 qui envoie le datagramme ;
- ***Contrôle de l'en-tête*** (16 bits) : contrôle d'erreur sur l'en-tête ;
- ***Adresse IP source*** et ***adresse IP destination*** (2x32 bits) ;
- **Options IP** (variable) : vérification de réseau, de sécurité ;
- Et enfin les ***données***.

Protocoles de routage

Les protocoles de routage les plus courants associés au protocole IP sont le RIP, l'IGRP, et l'OSPF.

Le **protocole de routage RIP** est un routage à vecteur de distance dont la métrique est en sauts, le plus souvent de maximum 15. Sa mise à jour est diffusée toutes les 30 secondes. La métrique du nombre de sauts sélectionne le chemin à emprunter.

Erreurs et supervision

Le **protocole ICMP** se charge de la supervision et de la délivrance de messages par le routeur en cas d'erreurs.

Ainsi, si un routeur reçoit un paquet qu'il est incapable de livrer à sa destination finale, le routeur envoie à la source un message ICMP « Destination inaccessible », qui l'avertit l'hôte, le port, ou le réseau sont inaccessibles.

La commande « Ping » permet de connaître la présence d'un ordinateur. Elle génère un message ICMP. Une réponse d'écho est alors attendue, mais les messages peuvent être tout autre.

Protocole RARP

L'adresse physique de l'ordinateur est l'élément unique l'identifiant. Un ordinateur peut obtenir son adresse IP ou celle d'un autre en s'adressant à un **serveur RARP**.

RARP est dérivé du protocole ARP, mais fait également l'inverse, le R signifie donc « Reverse ».

Il faut savoir que ARP et RARP sont mis en œuvre directement par-dessus la couche Liaison de données.

TCP et UDP

Les protocoles **TCP** et **UDP** se partagent la couche transport.

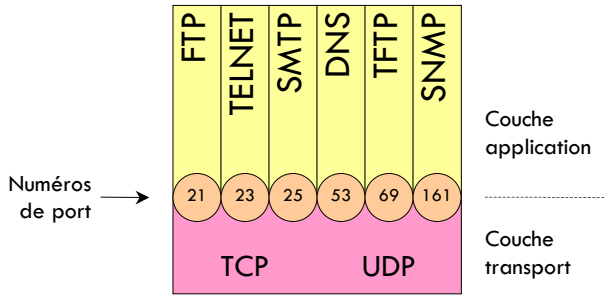
Le TCP est fiable, divise les messages sortants et assemble les messages entrants, et se charge automatiquement de renvoyer tout message non reçu.

L'UDP quant à lui brille par son absence de fiabilité et de confirmations, par contre il est nettement plus rapide que son confrère.

Les protocoles TCP et UDP utilisent des numéros de port (ou de prise) pour transmettre de l'information aux couches supérieures. Les numéros de port servent à distinguer les différentes conversations qui circulent simultanément sur le réseau

Par convention, les numéros inférieurs à 255 sont réservés aux applications publiques. Les numéros compris entre 255 et 1023 sont attribués aux entreprises pour les applications à commercialiser.

Voici les numéros de port les plus couramment utilisés :



Le **protocole TCP** fonctionne en mode connecté. Il est fiable, que ce soit à l'ouverture ou la fermeture de la connexion. Il offre un contrôle de flux/pertes/erreurs par fenêtres au niveau de l'octet, permet la négociation de la taille des fenêtres, par contre il ne gère pas les trames de supervision et de rejet.

Voici la structure d'un segment TCP :

- ***Port source*** (16 bits) : numéro du port demandeur ;
- ***Port de destination*** (16 bits) : numéro du port demandé ;
- ***Numéro de séquence*** (32 bits) : numéro utilisé pour assurer la bonne séquence des données entrantes ;
- ***Numéro d'accusé de réception*** (4 bits) : prochain octet TCP attendu ;
- ***HLEN*** (4 bits) : nombre de mots de 32 bits contenus dans l'en-tête ;
- ***Réservé*** (6 bits) : réglé à zéro ;
- ***Bits de code*** (6 bits) : ils déterminent la nature du segment :
  - URG=1 si le champ « pointeur urgent » est positionné,
  - ACK=1 si le champ « numéro d'acquittement » est significatif,
  - EOM=1 indique la fin du message,
  - RST sert à réinitialiser la connexion,
  - SYN sert à établir la connexion,
  - FIN indique que l'émetteur n'a plus de données.
- ***Fenêtre*** (16 bits) : nombre d'octets que l'émetteur est prêt à accepter ;
- ***Total de contrôle*** (16 bits) erreur calculée sur l'en-tête et les données ;
- ***Pointeur d'urgence*** (16 bits) : indique la fin des données urgentes ;
- ***Options*** (0 ou 32 bits) : taille maximale d'un segment TCP ;
- ***Données*** : contient les données du protocole de couche supérieure.

Le **protocole UDP** utilise également les datagrammes, ainsi qu'un contrôle d'erreurs sur les données reçues. Par contre, il n'y a pas de contrôle des pertes, ni de contrôle de flux. Il est simple et rapide, mais nécessite une connexion de bonne qualité.

Ce protocole n'offre pas de numérotation, ni d'accusés de réception; il n'y a pas non plus de retransmission. C'est la couche application qui doit assurer la fiabilité au besoin. Un segment UDP se structure ainsi :

- ***Port source*** (16 bits),
- ***Port de destination*** (16 bits),
- ***Longueur*** (16 bits),
- ***Total de contrôle*** (16 bits),
- ***Données***.

Bibliographie

- *Réseaux locaux et Internet,*
  L. Toutain, Hermes ;
- *Les réseaux locaux commutés et ATM,*
  A. Ferréro, InterEditions ;
- *Les Réseaux, principes fondamentaux,*
  P. Rolin, Hermes ;
- *OSI, les normes de communications entre systèmes ouverts,*
  J. Henshall et S. Shaw, Masson ;
- *Architecture des réseaux haut débit,*
  K. L. Thai, V. Vèque, et S. Znaty, Hermes ;
- *Les réseaux locaux industriels,*
  F. Lepage et C., Hermes ;
- *Réseaux : Architectures, protocoles et applications,*
  Andrew Tanenbaum, IIA ;
- *Réseaux locaux et migrations de systèmes,*
  Pierre Jacquet, Eyrolles ;
- *L'intelligence dans les réseaux,*
  D.Gaiti et G. Pujolle, Eyrolles ;
- *Gestion des réseaux informatiques,*
  J.P. et M. Claudé, Eyrolles ;
- *Réseaux Informatiques 2,*
  D. Dromard, F. Ouzzani, D. Seret et K. L. Thai, Eyrolles ;
- *Théorie de l'information : Application aux techniques de communication,*
  G. Battail, Masson ;
- *Installer et configurer un routeur Cisco,*
  Chris Lewis, Eyrolles ;
- *Cisco Installation Configuration Utilisation,*
  George C Sackett, Eyrolles;
- *IP Routing Fundamentals,*
  Mark Sportack, Cisco Press;
- *Configuration IP des routeurs Cisco,*
  Innokenty Rudenko, Eyrolles.