RÉSEAUX - PROTOCOLES - SERVEURS ...[1]

Objectifs : Etre informé sur les réseaux, protocoles et serveurs

1 Réalisé à partir du cours de réseau de Christian Caleca

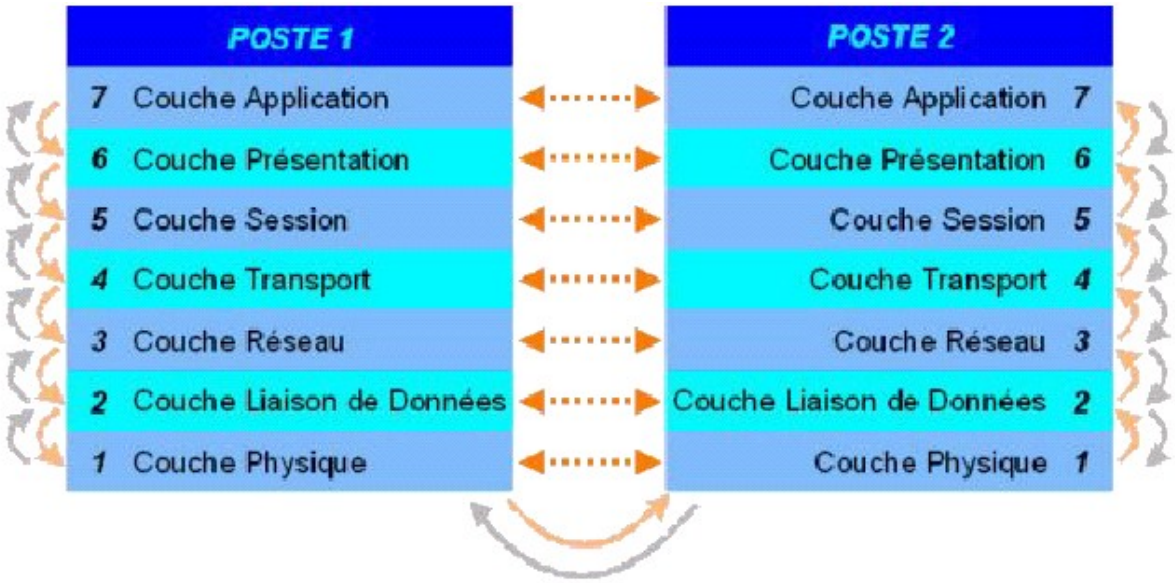
1. LE MODÈLE OSI

1.1. La communication sur un réseau

Le fondement d'un bon réseau, c'est que le système d'exploitation soit capable :

- De gérer la transmission de données.
- De fournir aux applications des interfaces standard pour leur permettre d'exploiter les ressources du réseau.

C'est le cas de tous les systèmes d'exploitation à jour.
A priori, rien ne devrait obliger les plates formes client et serveur à fonctionner avec le même système d'exploitation. C'était le cas pour les solutions propriétaires, c'est impensable aujourd'hui.
Même si un réseau Microsoft dispose d'outils qui lui sont spécifiques, les hôtes de ce réseau peuvent tout de même dialoguer avec ceux d'un réseau Unix.
Pour arriver à cette interopérabilité, il faut que les divers protagonistes se mettent d'accord sur les fonctionnalités à implanter dans leurs applications et leurs fonctions réseau. C'est le rôle des RFC (Request For Comment) et des normes que de définir ces critères.
C'est l'objectif du modèle théorique O.S.I. qui décrit comment l'O.S. réseau, encore appelé N.O.S. doit être construit. Il décrit l'architecture en 7 couches logicielles présentant chacune des interfaces standard pour communiquer entre elles.



Il y a deux points qu'il convient de bien comprendre avant tout :

- Chaque couche est conçue de manière à dialoguer avec son homologue, comme si une liaison virtuelle était établie directement entre elles.

- Chaque couche fournit des services clairement définis à la couche immédiatement supérieure, en s'appuyant sur ceux, plus rudimentaires, de la couche inférieure, lorsque celle-ci existe.

1.2. La couche physique : 1

C'est la couche spécifique à la "tuyauterie" du réseau. Elle permet de transformer un signal binaire en un signal compatible avec le support choisi (cuivre, fibre optique, HF etc.) et réciproquement.
C'est à ce niveau que se situe l'adresse MAC (Medium Access Control).
Cette couche fournit des outils de transmission de bits à la couche supérieure, qui les utilisera sans se préoccuper de la nature du médium utilisé.

1.3. La couche liaison : 2

Cette couche assure le contrôle de la transmission des données. Une trame doit être envoyée ou reçue en s'affranchissant d'éventuels parasites sur la ligne. Le contrôle est effectué au niveau du paquet de bits (trame), au moyen d'un "checksum".
Elle est elle-même divisée en deux sous-couches.

- La sous-couche MAC (Medium Access Control). C'est à ce niveau que l'on trouve le protocole de diffusion de l'information: Ethernet, Token Ring, ATM, etc. Pour un réseau domestique, c'est Ethernet qui est utilisé.
- Par ailleurs, cette couche fournit des services de base pour la transmission de données, via LLC (Logical Link Control) ou HDLC (High level Data Link Control). Ces services peuvent être classés en trois groupes :
  - Les services sans connexion et sans acquittement.
  - Les services sans connexion, mais avec acquittement.
  - Les services orientés connexion.

Nous aurons l'occasion de reparler de ces notions plus loin, dans les protocoles UDP et TCP.
Cette couche fournit des outils de transmission de paquets de bits (trames) à la couche supérieure.
Les transmissions sont "garanties" par des mécanismes de contrôle de validité.

1.4. La couche Réseau : 3

Cette couche assure la transmission des données sur les réseaux. C'est ici que la notion de routage intervient, permettant l'interconnexion de réseaux différents. C'est dans le cas de TCP/IP20 la couche Internet Protocol. En plus du routage, cette couche assure la gestion des congestions. Il faudrait beaucoup développer ce chapitre pour être clair. Disons simplement que lorsque les données arrivent sur un routeur, il ne faudrait pas que le flot entrant soit plus gros que le flot sortant maximum possible, sinon il y aurait congestion. Une solution consiste à

contourner les points de congestion en empruntant d'autres routes (phénomène bien connu des vacanciers sur les routes).
Le problème de la congestion est un problème épineux, auquel il nous arrive assez souvent hélas d'être confrontés.
Cette couche est la plus haute dans la partie purement "réseau".
Cette couche fournit des outils de transmission de paquets de bits (trames) à la couche supérieure. Les transmissions sont routées et la congestion est contrôlée.

1.5. La couche Transport : 4

Cette couche apparaît comme un superviseur de la couche Réseau. Qu'est-ce à dire? Il n'est par exemple pas du ressort de la couche réseau de prendre des initiatives si une connexion est interrompue.
C'est la couche Transport qui va décider de réinitialiser la connexion et de reprendre le transfert des données.
Son rôle principal est donc de fournir à la couche supérieure des outils de transport de données efficaces et fiables.

1.6. La couche Session : 5

La notion de session est assez proche de celle de connexion. Il existe cependant quelques détails qui peuvent justifier la présence de ces deux concepts.
Une seule session peut ouvrir et fermer plusieurs connexions, de même que plusieurs sessions peuvent se succéder sur la même connexion. Comme cette explication n'est pas forcément claire pour tout le monde, essayons de prendre quelques exemples :

- Vous avez un message à transmettre par téléphone à un de vos amis, votre épouse doit faire de même avec celle de ce même ami.

Vous appelez votre ami (ouverture d'une connexion), vous discutez avec lui un certain temps (ouverture d'une session), puis vous lui dites que votre épouse voudrait parler à la sienne (fermeture de la session).
Les épouses discutent un autre certain temps (ouverture d'une seconde session), puis n'ont plus rien à se dire (fermeture de la seconde session) et raccrochent (fin de la connexion).
Dans cet exemple, deux sessions ont eu lieu sur la même connexion.

- Vous avez un travail à réaliser avec un collègue, par téléphone. Vous l'appelez (ouverture de la connexion et ouverture de la session). Il vous demande des informations qui nécessitent de votre part une recherche un peu longue, vous raccrochez après lui avoir dit que vous le rappellerez ultérieurement (fermeture de la connexion, mais pas de la session).

Votre recherche effectuée, vous rappelez votre collègue (ouverture d'une seconde connexion pour la même session), vous lui transmettez les informations demandées, vous n'avez plus rien à vous dire (fermeture de la session), vous raccrochez (fermeture de la connexion).

Dans cet exemple une session s'étend sur deux connexions.
Cette couche fournit donc à la couche supérieure des outils plus souples que ceux de la couche transport pour la communication d'informations, en introduisant la notion de session.

1.7. La couche Présentation : 6

Cette couche est un peu un "fourre tout" de la conversion entre représentation interne et externe des données. Là encore, cette explication n'est pas d'une grande clarté... Prenons donc quelques exemples.

1.7.1. Le format de codage interne des données

Les "mots" sont une suite d'octets. Un mot de 32 bits est donc une collection de 4 octets. Chez Intel, les octets sont numérotés de droite à gauche, alors que chez Motorola, ils le sont de gauche à droite. Il s'en suit que si une machine à base Intel envoie des mots à une machine à base Motorola, il vaut mieux tenir compte de ce détail pour ne pas s'y perdre...
D'autres exemples pourraient être trouvés dans le style, comme la complémentation à 1 ou à 2 pour les représentations négatives, les divers dialectes ASCII etc.

1.7.2. La compression des données

Certains transferts de données se font par le biais d'algorithmes de compression. C'est intéressant pour certains types de documents comme le son, l'image ou la vidéo. Dans le modèle OSI, ces algorithmes sont fournis par la couche présentation.

1.7.3. Le cryptage des données

Même chose pour la cryptographie.
Ce ne sont que des exemples, le modèle définissant bien d'autres fonctions.

1.8. La couche Application : 7

A priori, cette couche pourrait être la plus simple à comprendre, ce n'est pas obligatoirement le cas.
En effet, dans le modèle OSI, cette couche propose également des services: Principalement des services de transfert de fichiers, (FTP), de messagerie (SMTP) de documentation hypertexte (HTTP) etc.
Dans le modèle, les applications ayant à faire du transfert de fichiers utilisent le service FTP fourni par la couche 7.

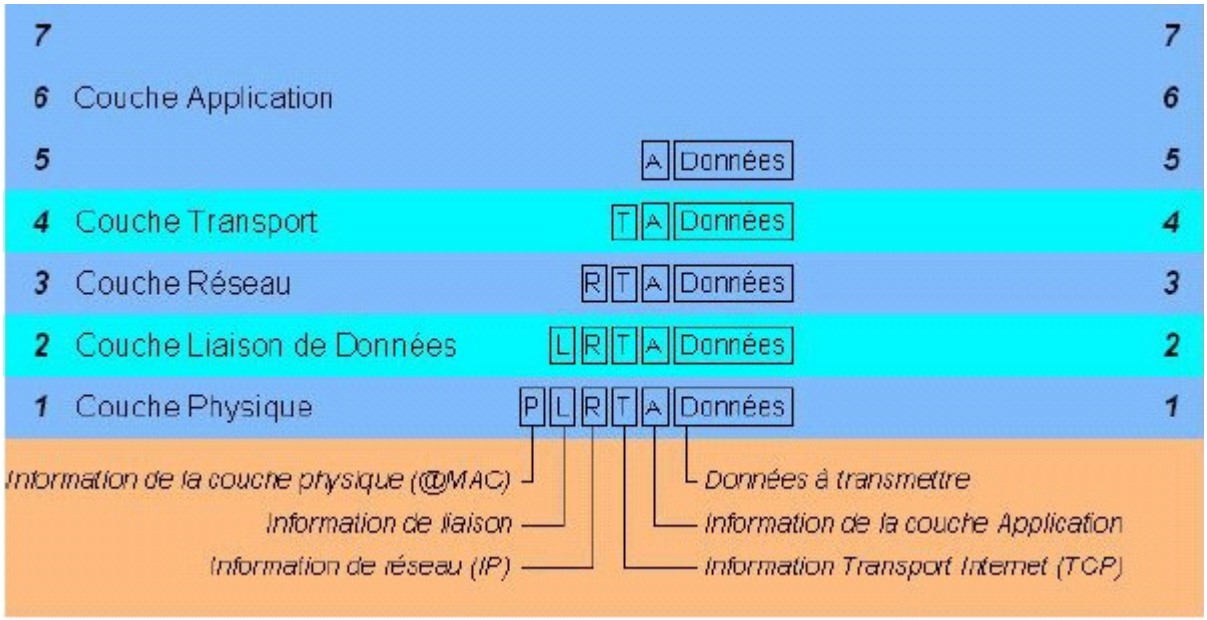
Ce modèle théorique, extrêmement détaillé est fait pour que chaque couche puisse être construite indépendamment des couches qui sont immédiatement au dessus et au dessous d'elle.

1.9. En pratique

Le modèle O.S.I. est tellement théorique qu'il va à l'encontre de l'efficacité. Il est donc souvent simplifié. Les simplifications se bornant toutefois à regrouper les fonctions de plusieurs couches O.S.I. en une seule.
Exemple: Le modèle D.O.D. utilisé dans le protocole TCP/IP21. Cependant, les couches O.S.I. Restent une référence dès lors que l'on parle de transmission de données sur un réseau.

Lorsqu'une information (Données) est émise par une application, cette donnée descend les diverses couches du réseau, en récupérant au passage des informations supplémentaires à chaque couche, comme le montre l'illustration suivante.

Modèle OSI	Modèle DOD
7 Couche Application	Couche Application
6 Couche Présentation	
5 Couche Session	
4 Couche Transport	Couche Hôte à Hôte
3 Couche Réseau	Couche Internet
2 Couche Liaison de Données	Couche Accès Réseau
1 Couche Physique	



Les trames qui circulent sur le réseau contiennent donc non seulement les données des applications, mais également tout un tas d'informations rajoutées par le N.O.S. Ces diverses informations permettront entre autres fonctions :

- Le pontage
- Le routage
- L'identification du poste émetteur
- L'identification du poste récepteur
- L'identification de l'application Emettrice
- L'identification de l'application Réceptrice

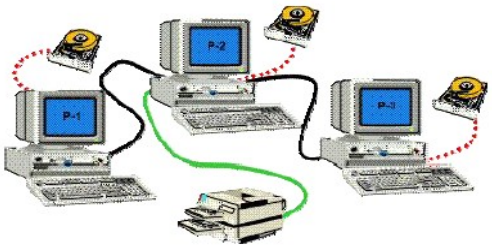
Lorsque la trame entre dans le récepteur, elle remonte les couches qui lui enlèvent au passage les informations qui les concernent, si bien que l'application reçoit ses données sans se préoccuper de la façon dont elles ont été transportées.

Remarque : l'adresse MAC est une adresse unique de votre carte éthernet. Cette adresse est fixe et définit à la fabrication.

2. LES RÉSEAUX

Un réseau permet de connecter des ordinateurs entre eux. Mais les besoins sont très divers, depuis le réseau domestique ou d'une toute petite entreprise jusqu'aux réseaux des grandes sociétés. Voyons deux approches fondamentalement différentes, encore que l'une peut facilement évoluer vers l'autre.

2.1. Le "Peer to Peer"



2.1.1. Principe

- Les postes de travail sont simplement reliés entre eux par le réseau. Aucune machine ne joue un rôle particulier. Chaque poste peut partager ses ressources avec les autres postes.
- C'est à l'utilisateur de chaque poste de définir l'accès à ses ressources. Il n'y a pas obligatoirement d'administrateur attitré.
- Dans l'exemple, chaque poste peut partager tout ou partie de sa mémoire de masse, le poste P-2 peut partager son imprimante.

2.1.2. Avantages

Il y en a quelques uns...

• Il est facile de mettre en réseau des postes qui étaient au départ isolés.

• Chaque utilisateur peut décider de partager l'une de ses ressources avec les autres postes.

• Dans un groupe de travail, l'imprimante peut être utilisée par tous.

• Cette méthode est pratique et peu coûteuse pour créer un réseau domestique.

2.1.3. Inconvénients

Il y en a beaucoup !

• Chaque utilisateur a la responsabilité du fonctionnement du réseau.

• Les outils de sécurité sont très limités.

• Si un poste est éteint ou s'il se "plante", ses ressources ne sont plus accessibles.

• Le système devient ingérable lorsque le nombre de postes augmente.

• Lorsqu'une ressource est utilisée sur une machine, l'utilisateur de cette machine peut voir ses performances diminuer.

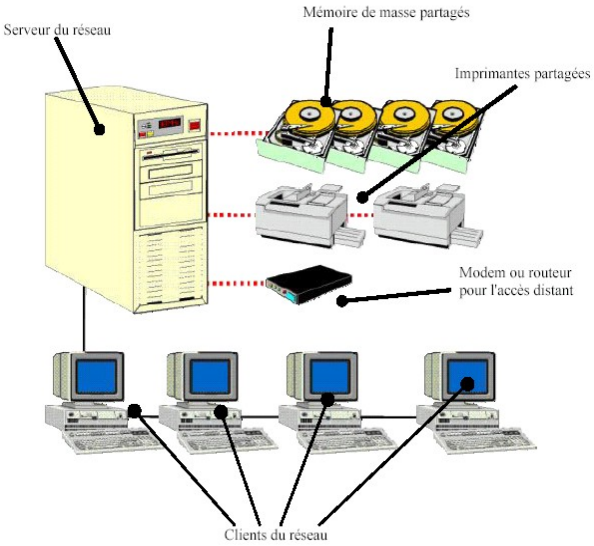
2.1.4. Conclusions

Ce type de réseau n'offre de réel intérêt que dans une configuration particulière :

• Les postes sont peu nombreux (pas plus d'une dizaine).

• Les utilisateurs restent attachés à un poste dont ils sont responsables.

2.2. Le "Client / Serveur"



2.2.1. Principe

- Les ressources réseau sont centralisées.
- Un ou plusieurs serveurs sont dédiés au partage de ces ressources et en assurent la sécurité.
- Les postes clients ne sont en principe que des clients, ils ne partagent pas de ressources, ils utilisent celles qui sont offertes par les serveurs.

2.2.2. Avantages

Il y en a beaucoup...

• Les serveurs sont conçus pour le partage de ressources et ne servent pas de station de travail. Il suffit de les dimensionner en fonction de la taille du réseau et du nombre de clients susceptibles de s'y connecter.

• Les systèmes d'exploitation de serveurs proposent des fonctions avancées de sécurité que l'on ne trouve pas sur les réseaux "peer to peer".

• Ils proposent également des fonctions avancées à l'usage des utilisateurs comme par exemple les profils itinérants qui permettent à un utilisateur (sous certaines conditions) de retrouver son environnement de travail habituel, même s'il change de poste de travail.

• Les serveurs étant toujours en service (sauf en cas de panne...), les ressources sont toujours disponibles pour les utilisateurs.

• Les sauvegardes de données sont centralisées, donc beaucoup plus faciles à mettre en oeuvre.

• Un administrateur gère le fonctionnement du réseau et les utilisateurs n'ont pas à s'en préoccuper.

2.2.3. Inconvénients

Il y en a quelque-uns tout de même...

• La mise en place d'un tel réseau est beaucoup plus lourde qu'un cas simple de "poste à poste".

• Elle nécessite i**mpérativement** la présence d'un administrateur possédant les compétences nécessaires pour faire fonctionner le réseau.

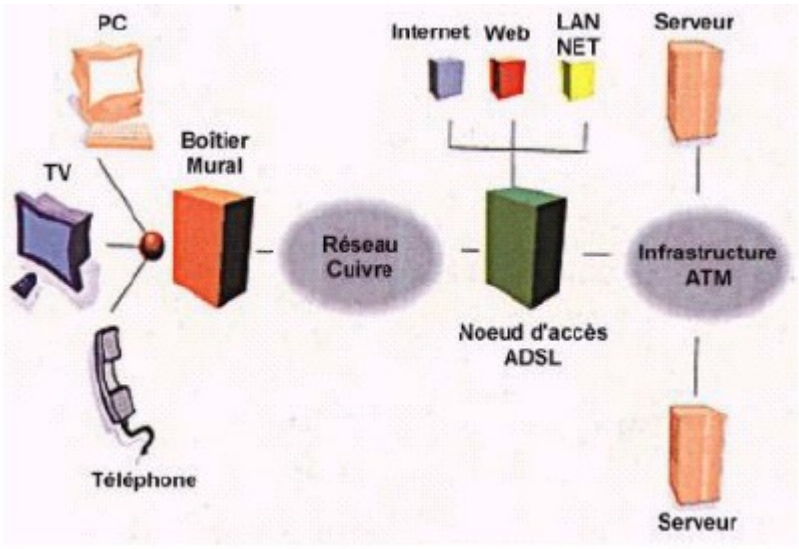
• Le coût est évidemment plus élevé puisqu'il faut la présence d'un ou de plusieurs serveurs.

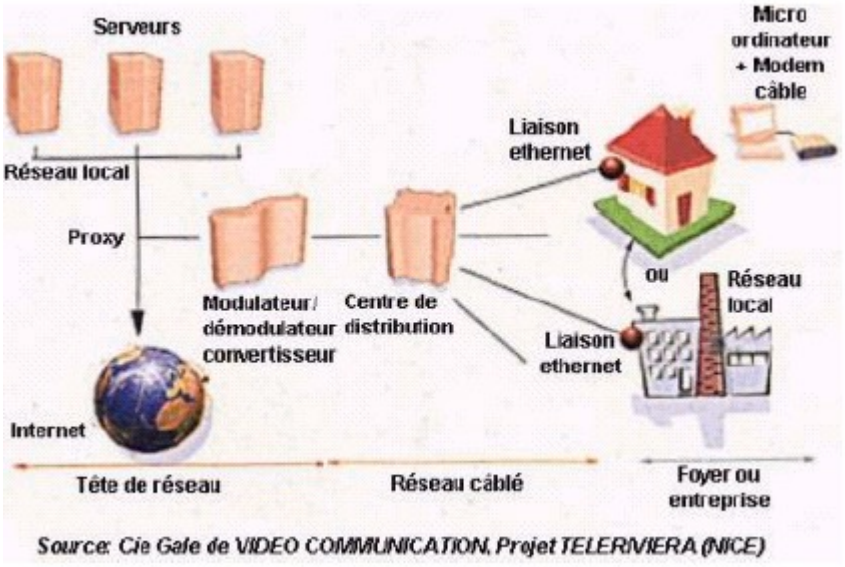
• Si un serveur tombe en panne, ses ressources ne sont plus disponibles. Il faut donc prévoir des solutions plus ou moins complexes, plus ou moins onéreuses, pour assurer un fonctionnement au moins minimum en cas de panne.

2.2.4. Conclusion

Ce type de réseau est évidemment le plus performant et le plus fiable. Vous l'aurez compris, ce n'est pas la solution la plus simple pour un réseau domestique, c'est cependant ce type d'architecture que l'on retrouve sur les réseaux d'entreprise, qui peut parfaitement supporter plusieurs centaines de clients, voire plusieurs milliers.

2.3. Les technologies actuelles

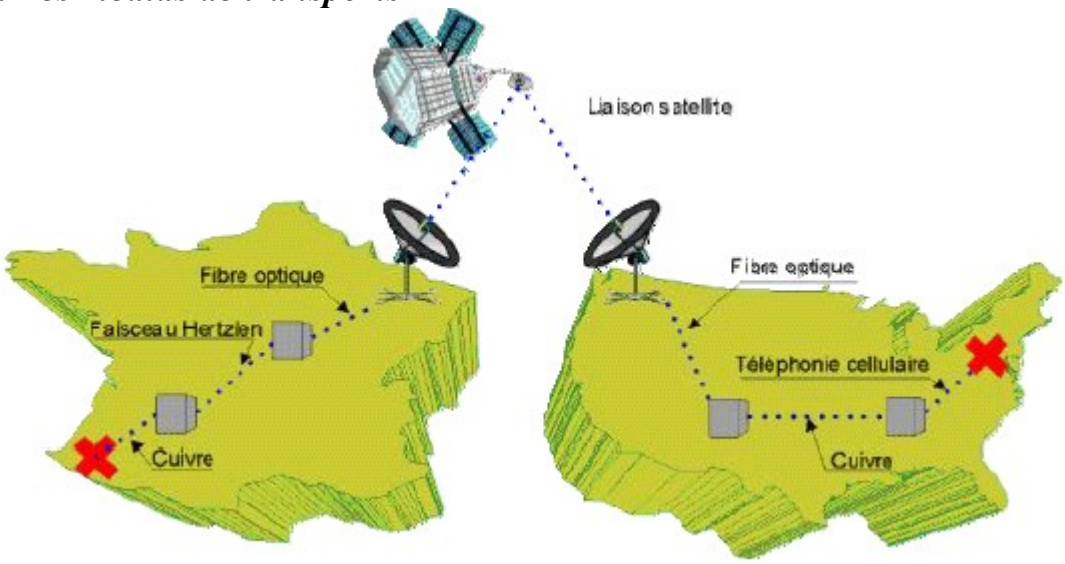




Source: Cie Gale de VIDEO COMMUNICATION, Projet TELERIVIERA (NICE)

2.4. La partie Hardware d'un réseau

2.4.1. Les médias de transports



2.4.2. Les interfaces avec l'ordinateur

Le rôle de cette interface est fondamental :

L'aspect physique

Il faut assurer la continuité du passage des données entre le média du réseau et le bus de données de l'ordinateur. Mais ce média peut être :

- Une fibre optique
- De la paire torsadée
- Du câble coaxial
- Une onde hertzienne
- Un faisceau lumineux infrarouge...

L'aspect Logique

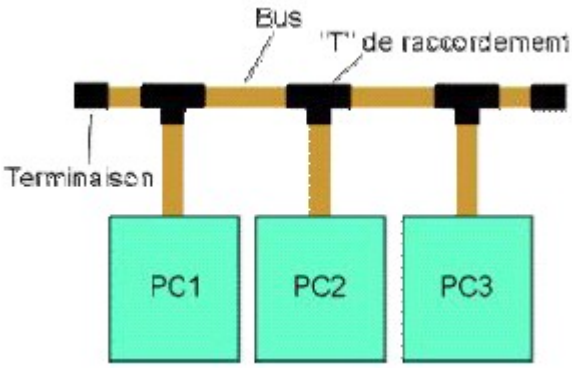
L'interface est étroitement liée au niveau 1 du modèle O.S.I.6. Son "firmware"7 doit donc tenir compte des spécifications de la norme, afin de pouvoir supporter les couches supérieures (c'est à dire les divers protocoles réseau). En d'autres termes, cette interface doit apporter une complète indépendance entre les logiciels réseau et le support matériel utilisé.

2.5. Les passerelles entre réseaux

Il existe une multitude de "passerelles" entre réseaux. **D'une manière générale, une passerelle permet la communication entre deux réseaux distincts qui peuvent être aussi différents que possible.**
Chaque passerelle sera adaptée au besoin spécifique.
Sans entrer ici trop dans le détail, on peut considérer une passerelle quelconque comme étant un ordinateur muni de plusieurs interfaces, une pour chaque réseau, avec un logiciel capable de faire transiter les informations d'un réseau vers l'autre
lorsque c'est nécessaire.
Dans notre rayon d'action, seul le cuivre va nous être accessible (l'onde Hertzienne également avec le protocole 802.11 qui permet la construction d'un réseau local sans fil, mais c'est quand même nettement plus cher).

2.6. Câblage d'un réseau

2.6.1. Le bus



Avantages

Il n'y a qu'un seul avantage à utiliser cette technologie, mais il est de taille:

• Après avoir vu les divers constituants, il devient évident que ce procédé est peu coûteux, facile et rapide à mettre en œuvre.

Inconvénients

Ils sont hélas nombreux :

• Lorsque le réseau dépasse les dimensions d'une pièce, il faut alors passer les murs, ce qui "fige" considérablement la topologie et diminue les possibilités d'extension.

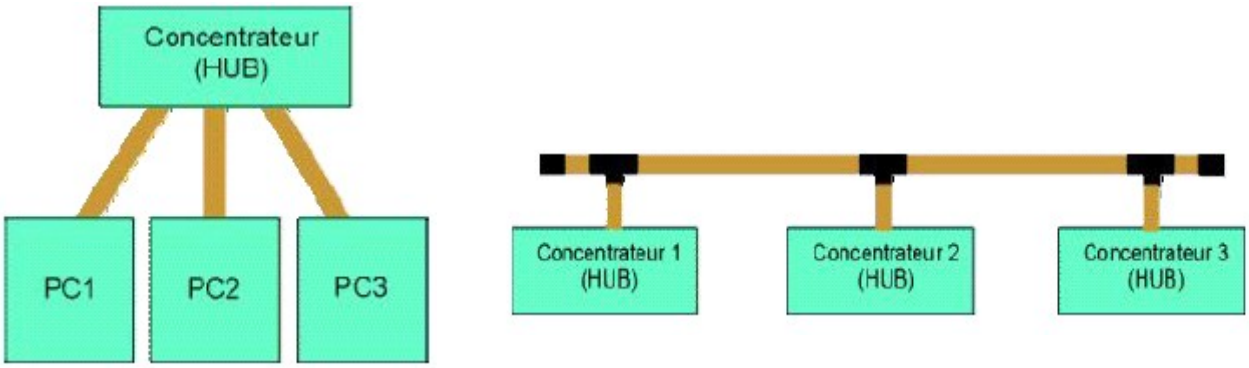
• Si un défaut de connectique apparaît, c'est tout le réseau qui devient inopérant.En effet, tout se passe alors comme si l'on avait deux réseaux, mais chacun d'eux ayant une extrémité non adaptée.Plus rien ne fonctionne et le défaut n'est pas toujours visible. Les investigations sont longues et laborieuses.

Conclusions

Malheureusement, ce type de réseau est limité à 10 Mbits/s et n'a plus d'avenir, bien qu' encore suffisant pour un réseau domestique.

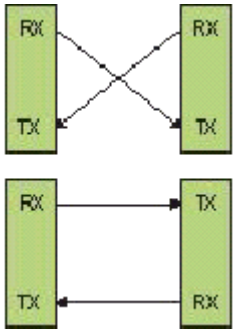
Il devient de plus en plus difficile de trouver ce genre d'adaptateur réseau. Les derniers modèles encore vendus sont souvent de type "combo", c'est à dire qu'ils permettent aussi bien un câblage coaxial en BUS qu'un câblage en étoile avec des paires torsadées, comme nous allons le voir tout de suite. Naturellement, un seul de ces deux modes est utilisable pour une interface donnée.

2.6.2. L'étoile



Chaque PC est relié par un câble constitué de 4 paires torsadées (dont deux seulement servent, normalement, l'une pour l'émission et l'autre pour la réception) à un concentrateur, encore appelé "HUB".

Sur de la paire torsadée, chaque paire est unidirectionnelle.



Le câble de type 5 est constitué de 4 paires torsadées. il peut être blindé (écranté) ou non. Le type 5 est certifié pour les réseaux 100 Mb/s. Le câble écranté offre une meilleure immunité au bruit électronique, il est à utiliser de préférence, même si son coût est plus élevé.

Ce type de câble est terminé par des connecteurs "RJ45". Suivant qu'un blindage existe ou non sur le câble, le connecteur est à choisir en conséquence. Il se place simplement si l'on dispose d'une pince spécialement conçue pour cet usage.

La prise est vue du côté des points de contact

1	TX+	Transmission de données +	Blanc/Orange
2	TX-	Transmission de données -	Orange
3	RX+	Réception de données +	Blanc/Vert
4	1+	Non utilisé en 10/100 BT	Bleu
5	1-		Blanc/Bleu
6	RX-	Réception de données -	Vert
7	2+	Non utilisé en 10/100BT	Blanc/Marron
8	2-		Marron

Avantages	Inconvénients
• D'un fonctionnement beaucoup plus sûr que le bus, si un lien vient à se rompre, seul le PC connecté par ce lien est absent du réseau. • Il est aisé d'ajouter des postes au réseau, même s'ils sont dans une pièce. • Cette technologie permet de réaliser un réseau 100 Mbits/s (à condition de disposer de HUBS qui savent le faire).	• La longueur totale de câble mise en œuvre est importante. • Au voisinage du HUB, on obtient un faisceau de câbles imposant. • Le coût est tout de même plus élevé que dans une architecture BUS.

Conclusions

Il existe des HUBS 100 Mb/s de 8, 12 ou 24 ports, que l'on peut monter en BUS sur une fibre optique, autorisant des réseaux à haut débit et de très grande taille.
De plus, des concentrateurs plus performants appelés "switches" permettant une meilleure répartition de la bande passante du réseau et augmentant encore les capacités de ce dernier.

2.7. La partie Software d'un réseau

C'est bien de disposer d'un ensemble de postes connectés entre eux, encore faut-il établir des protocoles pour transmettre les données avec quelques espoirs d'efficacité. Des protocoles, nous allons en voir quelques uns et à tous les étages. Mais commençons par le niveau le plus bas, sur le câble lui-même.

2.7.1. Trois types d'organisation réseaux

2.7.2. Ethernet

Il s'agit du système ETHERNET (à ne pas confondre avec INTERNET). Ici, un poste qui doit émettre commence par écouter le réseau. Si personne n'est en train de parler, il émet une trame de données. Comme chaque poste s'assure qu'il y a le silence avant de prendre la parole, les choses se passent en général bien.

Cependant, lorsqu'il y a beaucoup de postes, il peut se faire que deux postes décident d'émettre en même temps; il y a alors une collision entre les deux trames émises et les données deviennent inutilisables.

ETHERNET utilise donc un système de détection de collision. Dans un tel cas, chaque poste attendra un temps aléatoire et refera une tentative.

C'est le procédé le plus employé dans les réseaux actuels. celui que nous utiliserons sur un réseau local.

Avantages	**Inconvénients**
Lorsqu'il y a peu de trafic sur le réseau, il n'y a pas de perte de temps et les communications sont très rapides. Les médias mis en œuvre sont simples (paires torsadées ou coaxial) et peu onéreux, de même que la connectique.	Lorsque le taux de collision devient important, le réseau perd beaucoup de temps à transporter des informations inutilisables et le rendement diminue, la bande passante étant alors consommée par les collisions. Une autre caractéristique peut devenir un inconvénient: Il est impossible de déterminer le temps qu'il faudra pour être sûr qu'un poste a pu parler à un autre, ce temps pouvant être très court s'il y a peu de trafic ou beaucoup plus long s'il y a beaucoup de collisions.

2.7.3. L'organisation déterminée : Token ring

C'est le protocole "Token Ring" (Anneau à jeton).
Pour parler, il faut avoir le jeton. Le réseau est constitué comme un anneau sur lequel un contrôleur passe un jeton à chaque hôte connecté, à tour de rôle. Ne peut émettre que celui qui dispose du jeton.

Avantages	Inconvénients
Dans un tel système, il ne peut pas y avoir de collisions, c'est l'ordre parfait. Il est parfaitement possible, si l'on connaît le nombre de postes sur le réseau, de connaître le temps maximum qu'il faudra pour qu'un poste puisse parler à un autre. (intéressant dans la gestion d'événements "en temps réel").	Il est difficile de construire une vraie boucle ! En fait, le retour se fait dans le même câble. La connectique est donc plus complexe et onéreuse.

2.7.4. L'ATM

Le réseau ATM, mis au point par les opérateurs de télécommunications, est un procédé complexe et coûteux, mais qui garantit un fonctionnement fluide et une bande passante déterminée pour chaque poste du réseau; conditions indispensables pour effectuer de la téléphonie ou de la télévision, phénomènes en temps réel s'il en est !

Ces réseaux fonctionnent comme des réseaux commutés. Un chemin virtuel est établi entre les deux postes qui veulent échanger des données.

2.8. Les protocoles de communication

2.8.1. NetBEUI

Développé par Microsoft et IBM à l'époque des premiers réseaux de PC, ce protocole simplissime fonctionne très bien sur de petits réseaux. Malheureusement, son efficacité décroît avec le nombre de postes. De plus, il n'est pas "routable", ce qui fait que l'on ne peut interconnecter des réseaux NetBEUI autrement que par des ponts.

2.8.2. IPX/SPX

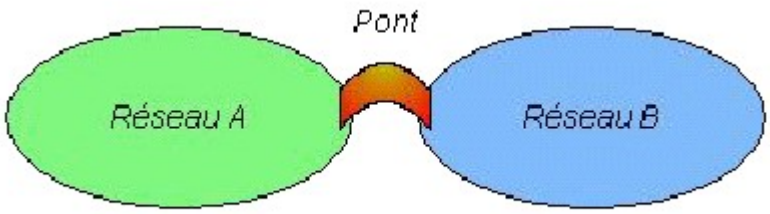
Développé par la société NOVELL, qui s'est octroyée la part du lion dans les premiers réseaux de PC avant que Microsoft ne développe Windows NT. Plus efficace que NetBEUI pour les gros réseaux, ce protocole est de plus routable ce qui augmente les possibilités d'interconnexions.

2.8.3. TCP/IP

Développé dans le monde UNIX, ce protocole est de très loin le plus compliqué. Cependant, il a été conçu au départ pour l'interconnexion de réseaux (IP=Internet Protocol !).
C'est le protocole le meilleur pour les gros réseaux et il est incontournable pour l'usage d'Internet.
C'est **LE** standard actuel.

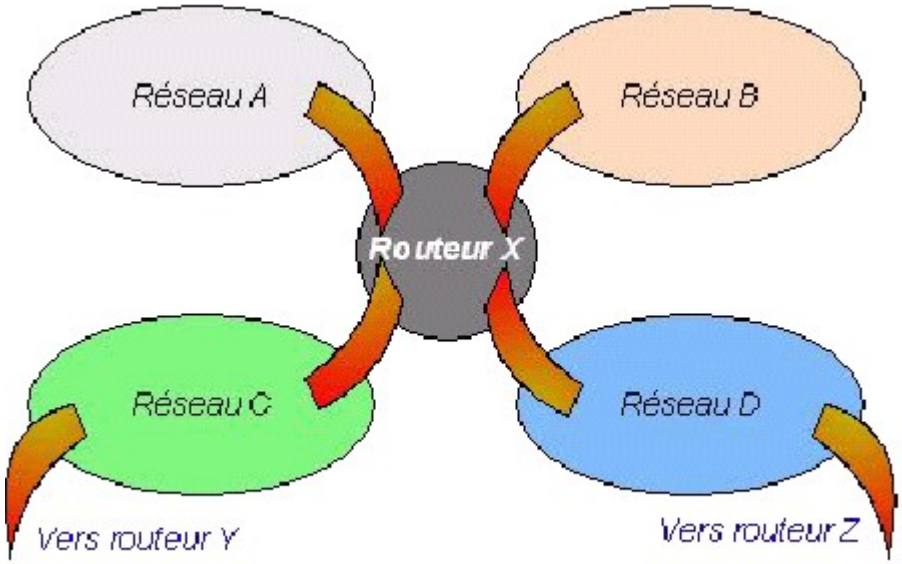
2.9. Les interconnexions

2.9.1. Les ponts



Ils sont utilisés pour interconnecter deux réseaux utilisant le même protocole, par exemple NetBEUI sur éthernet. Les ponts travaillent au niveau de la couche 2 du modèleOSI1(liaison de données).
Les ponts se basent sur l'adresse MAC (adresse en "dur" écrite dans l'interface) et le nom de la station sur le réseau pour savoir si la trame doit traverser le pont ou non. En d'autres termes, les informations ne passeront le pont que si elles doivent aller d'un réseau à l'autre.

2.9.2. Les routeurs



Les routeurs sont plus puissants: ils sont capables d'interconnecter plusieurs réseaux utilisant le même protocole entre eux. Ils travaillent au niveau de la couche 3 du modèle OSI (couche réseau) et tous les protocoles n'utilisent pas cette couche. C'est pourquoi l'on parle de protocoles "routables" ou "non routables".(NetBEUI n'est pas routable, TCP/IP et IPX/SPX le sont)

Les routeurs disposent d'une table de routage qui leur permet d'aiguiller les trames vers le bon réseau. Ils permettent une structure maillée, indispensable pour la construction de l'INTERNET.

Les passerelles

Pris au sens large, une passerelle est un outil permettant de passer d'un réseau à un autre. Dans un réseau TCP/IP14, l'adresse du routeur dans le réseau est dite "adresse de passerelle".

Au sens strict du terme, une passerelle est un outil permettant de faire communiquer entre eux deux réseaux n'utilisant pas le même protocole. La passerelle doit alors dépouiller la trame des informations spécifiques au

protocole émetteur et les remplacer par leurs équivalentes dans le protocole récepteur!

3. TCP/IP

Nous allons nous intéresser au protocole situé juste au dessus de la couche Ethernet, du moins au plus utilisé d'entre eux: TCP/IP. Ce protocole est en effet omniprésent sur le Net.

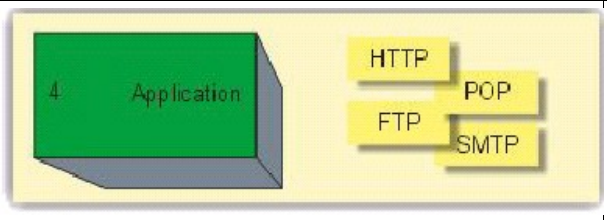
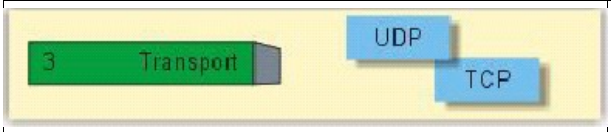
Nous traiterons

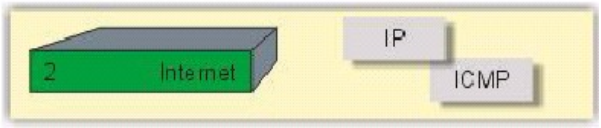
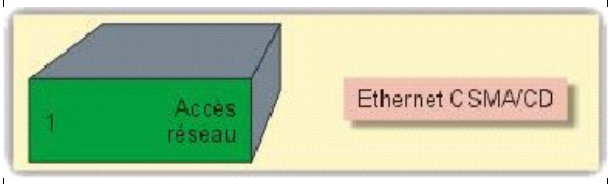
- des adresses logiques de l'Internet Protocol (couche 3 du modèle OSI)
- des modes connecté (TCP) et non connecté (UDP) (couche 4 du modèle OSI)
- des protocoles applicatifs (HTTP, FTP, SMTP, POP etc.) (couche 7 du modèle OSI)

3.1. Les protocoles

C'est un mode opératoire qui doit être commun à tous les éléments qui désirent communiquer entre eux. Il n'y a pas de communication possible sans avoir recours à un protocole. Bien entendu, le protocole doit être adapté au type de communication que l'on souhaite mettre en oeuvre.

3.1.1. Les principaux protocoles rencontrés

Association couche DOD / protocoles	*Description*
	Nous trouvons ici les protocoles applicatifs. Ce sont des protocoles de haut niveau, destinés à permettre le dialogue entre applications serveurs et clientes. HTTP, FTP, POP et SMTP sont loin d'être les seuls. Ce sont cependant ceux que les internautes utilisent le plus souvent. Parmi l'un des plus "dangereux", il y a TELNET qui permet de piloter une machine à distance.
	Ici, ce sont les protocoles orientés transport de données. UDP est dit "sans connexion" et TCP "est dit "avec connexion". Nous verrons plus loin ce que ceci veut dire. Ces protocoles permettent à ceux de la couche 4 de transporter leurs données de façon fiable.

Association couche DOD / protocoles	*Description*
	Ce sont ici des protocoles de haut niveau de la couche réseau. IP permet le routage des informations entre réseaux, c'est ici que l'adresse IP est utilisée. ICMP est un protocole de "contrôle" il met à disposition des outils de dépistage d'erreur et de signalisation. C'est un protocole important.
	Protocole de plus bas niveau sur le réseau, il assure la bonne gestion du médium (détection de collisions) et permet l'acheminement des informations entre émetteur et destinataire au niveau des adresses MAC. IP s'appuie dessus bien évidement.

3.1.2. Ethernet

Le mot "Ethernet" fait référence au support de propagation des informations utilisé. Historiquement, de trois types (mais d'autres peuvent être utilisés) :
• Coaxial épais
• Coaxial fin (RG58)
• Paire torsadée.
Pour être tout à fait précis, la norme qui décrit les réseaux de type Ethernet qui sont utilisés sur la majorité des réseaux locaux est la norme IEEE 802.3.

IP
Internet Protocol.
C'est le protocole dont on parle le plus, il est en effet directement impliqué dans la configuration réseau de l'hôte. C'est lui qui, en fonction de l'adresse IP du destinataire acheminera l'information sur la bonne route.

3.2. Les deux modes de transfert

Il existe deux modes de transfert :

3.2.1. Le mode connecté (TCP)

Dans ce mode, il se met en place un processus de "handshake" (poignée de main) entre le client et le serveur. Ce processus permet d'établir un dialogue à propos du transfert de données. Il y a des accusés réception, des demandes d'émission etc. qui permettent aux applications de savoir exactement où en est le processus de transfert de données.
Ce protocole est très robuste et permet un transfert de données dans de bonnes conditions.
Le "handshake" est un concept fondamental dans un protocole de dialogue robuste. En gros, ça veut dire :
"Chaque fois que tu envoies un message à son destinataire, assures-toi qu'il l'a reçu et compris"
La lettre recommandée avec accusé de réception est un bon exemple de mode connecté. Si l'émetteur reçoit l'accusé réception, alors il est certain que sa lettre est arrivée à destination.

3.2.2. Le mode non connecté (UDP)

C'est un mode simple, de type "on envoie les données et on espère qu'elles arriveront". Il n'y a pas de "connexion", au sens où on l'a vu pour le mode connecté. En revanche, il est possible de mettre en place un processus d'acquittement.
Ce mode est utilisé, par exemple, pour les requêtes DNS. Il offre l'avantage d'être moins gourmand en ressources, mais ne peut être efficace pour un transfert de fichiers et en général, pour les transferts de données volumineuses.
Dans ce mode, il n'y a pas de "handshake".
Une lettre simple et ici un bon exemple. L'émetteur ne reçoit à priori aucune confirmation de réception.

3.2.3. Les protocoles d'application utilisant TCP ou UDP

HTTP : ***Hyper Text Transfert Protocol :*** Ce protocole est utilisé pour la navigation web entre un serveur HTTP et un butineur.

FTP : ***File Transfert Protocol :*** Protocole qui permet d'assurer le transfert de fichiers de façon indépendante des spécificités des NOS (Network Operating System, pour mémoire).

SMTP : ***Simple Mail Transfert Protocol :*** Le protocole qui permet d'acheminer le courrier depuis le serveur SMTP de l'émetteur, jusqu'au serveur SMTP du destinataire, qui le classe dans les Boîtes aux lettres de ses clients.

POP3 : ***Post Office Protocol version 3 :*** Le protocole qui permet au client de relever à distance le courrier classé dans sa boîte aux lettres.

IMAP4 : ***Interactive Mail Access Protocol version 4 :*** Normalement, ce protocole devrait prendre la place de POP3. Certains fournisseurs sérieux, comme FREE l'implémentent déjà. Contrairement à POP3 qui ne permet une gestion des messages qu'une fois qu'ils sont rapatriés localement, IMAP propose des fonctionnalités plus fines.

NNTP : ***Network News Transfert Protocol :*** Très proche de SMTP, ce protocole est employé par les forums usenet.

TELNET : C'est un outil qui permet l'administration distante d'une machine, du moment que l'on est capable d'ouvrir une session et d'acquérir les droits de "super utilisateur". C'est le "couteau suisse" du travail à distance. En fait, un client TELNET est une console en mode texte, capable de se connecter sur la plupart des serveurs.

L'adresse IP

Avant de commencer

Il est bon de savoir qu'il existe une adresse "MAC" (Media Access Control), écrite normalement en "dur" dans la ROM de l'interface réseau et donc théoriquement ineffaçable et infalsifiable (mais ce n'est que la théorie, tous les pirates vous le diront). Cette adresse est réputée unique et décidée par le constructeur de la carte. Elle est la seule adresse exploitée au niveau 2 pour l'identification des hôtes qui dialoguent. Cette méthode ne permettant pas l'interconnexion de réseaux, il va être nécessaire d'ajouter dans la couche supérieure (niveau 3), une adresse logique qui sera attribuée par l'administrateur du réseau, en coordination avec les organismes chargés de gérer l'attribution de ces adresses. Dans le cas qui nous intéresse ici, il s'agit de la fameuse adresse IP.

Définition d'une adresse IP

Internet Protocol

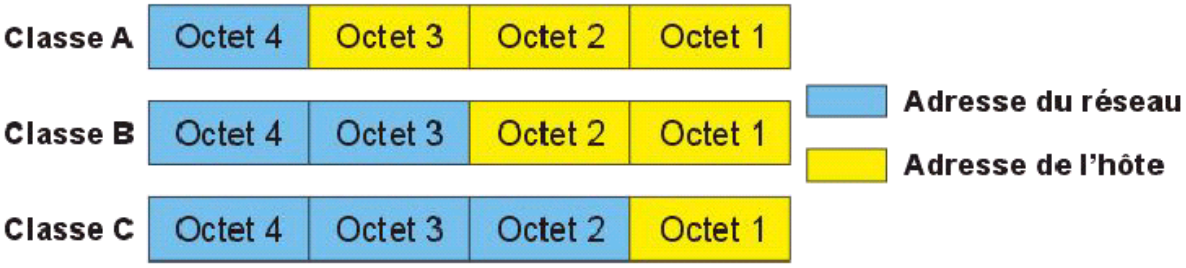
Il existe déjà sur le Net une multitude de pages qui traitent du sujet, ça ne fait rien, on va en mettre une de plus...

Dans sa version 4, IP définit une adresse sur 4 octets. Une partie définit l'adresse du réseau (NetID ou SubnetID suivant le cas), l'autre partie définit l'adresse de l'hôte dans le réseau (HostID). La taille relative de chaque partie varie suivant la classe choisie.

Les classes d'adresses

Topologie

Hormis la classe D multicast, destinée à faire de la diffusion d'information pour plusieurs hôtes simultanément, il existe trois classes d'adresses IP :



Comme vous le voyez, la classe A permet de créer peu de réseaux, mais avec beaucoup d'hôtes dans chaque réseau, La classe C faisant l'inverse.

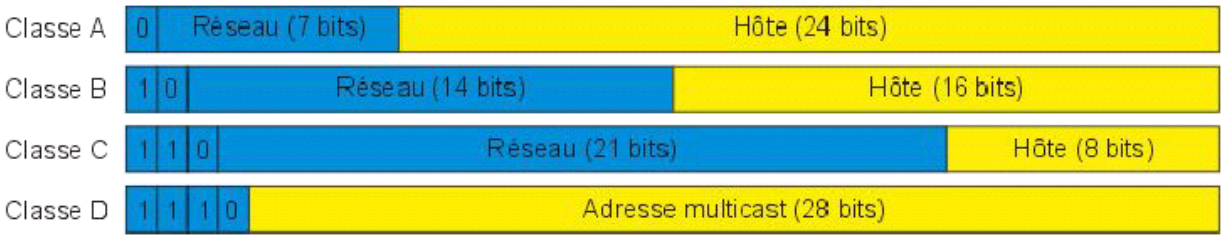
Étendue de chaque classe

Comment fait-on pour savoir à quelle classe appartient une adresse ? Il y a deux méthodes pour le savoir :

- La triviale, qui consiste à apprendre par cœur le tableau.
- La subtile, qui consiste à retenir la règle, qui est logique.

Voici donc la règle :

- La classe est définie par les bits les plus lourds (les plus à gauche)
- Le bit le moins signifiant pour la classe est toujours un 0
- Les autres sont tous à 1
- La classe A est signalée par un seul bit, donc obligatoirement un 0
- La classe B par deux bits, donc 1 0
- La classe C par trois bits, donc 1 1 0
- La classe D (multicast) par 4 bits donc 1 1 1 0



Il existe même une classe E, dont les bits les plus lourds sont 11110, qui est "réservée à un usage ultérieur".

Si l'on arrive à retenir la définition ou son image, ça devient facile de retrouver l'étendue de chaque classe :

Classe	*Première adresse*	*Dernière adresse*
A	0.0.0.1	127.255.255.254
B	128.0.0.1	191.255.255.254
C	192.0.0.1	223.255.255.254
D	224.0.0.1	239.255.255.254

A ce stade, nous pourrions penser qu'il peut y avoir, par exemple, 128 réseaux de classe A, avec la possibilité d'avoir 16 777 216 hôtes dans chaque réseau. C'est bien entendu, un peu plus compliqué que çà.

Il y a déjà quelques adresses que l'on ne peut pas attribuer à un hôte :

- L'adresse d'hôte =0 (exemple: 192.168.1.0 dans une classe C)
  Par convention, l'adresse IP dont la partie hôte est nulle est réservée à l'identification du réseau.
- L'adresse d'hôte avec tous ses bits à 1 (exemple: 192.168.1.255)
  Par convention, cette adresse signifie que tous les hôtes du réseau 192.168.1.0 sont concernés (Adresse de broadcast).

Les réseaux privés

Et ce n'est pas tout. Nous savons qu'une adresse Internet doit être unique dans un inter réseau. Cette considération, qui ne posait pas trop de problèmes pour des réseaux d'entreprise coupés du reste du monde, devient très restrictive à l'échelle de l'Internet où chaque adresse IP doit être unique à l'échelle planétaire. Ceci représente une contrainte énorme, et qui fait que la pénurie d'adresses IP est une catastrophe annoncée bien plus certaine que celle du bug de l'an 2000. (Rassurez-vous, le prochain protocole IP v6 prévoit de la marge, il faudra juste tout ré apprendre).

Pour permettre aux entreprises de construire leur réseau privé, il a donc été réservé dans chaque classe A, B et C des adresses de réseaux qui ne sont jamais attribuées sur l'Internet (RFC 1918)[8]. Tout paquet de données contenant une adresse appartenant à ces réseaux doit être éliminé par le premier routeur établissant une connexion avec l'Internet.

Ces réseaux privés sont:

Classe	*Réseaux privés*	*Identification*
A	10.0.0.0	Pour les réseaux privés
	127.0.0.0	Pour l'interface de boucle locale (*)
B	172.16.0.0 à 172.31.0.0	Pour les réseaux privés
C	192.168.0.0 à 192.168.255.0	Pour les réseaux privés
(*) L'adresse qui correspond à "localhost". Cette adresse locale est nécessaire au fonctionnement de la pile IP.		

Le masque de sous réseau

Le masque de sous-réseau a une importance que peu d'utilisateurs connaissent, elle est pourtant fondamentale. C'est un ensemble de 4 octets destiné à isoler :

- Soit l'adresse de réseau (NetID ou SubnetID) en effectuant un ET logique bit à bit entre l'adresse IP et le masque.
- Soit l'adresse de l'hôte (HostID) en effectuant un ET logique bit à bit entre l'adresse IP et le complément du masque (!masque).

Les masques de sous-réseau par défaut sont, suivant les classes d'adresses:

Classe	*Masque par défaut*	*Nbe d'octets pour l'hôte*
A	255.0.0.0	3
B	255.255.0.0	2
C	255.255.255.0	1

Par défaut, un masque de sous réseau englobe donc la totalité de la classe.

8 http://abcdrfc.free.fr/rfc-vf/rfc1918.html

Mais pourquoi "sous réseau"?

Le principe en est simple: Imaginons que nous disposions d'une classe B. Nous disposons donc de deux octets pour les adresses d'hôtes, soit 65 534 hôtes possibles (les adresses x.x.0.0 et x.x.255.255 sont réservées). Ca ferait tout de même beaucoup de machines sur le même réseau. En pareil cas, il est bien préférable d'organiser son réseau logique en plusieurs sous réseaux, connectés entre eux par des routeurs.

Si par exemple, bien qu'étant en classe B, on choisit comme masque de sous réseau 255.255.255.0, nous obtiendrons 256 sous réseaux de 254 hôtes chacun dans le même réseau. Mais il est possible de définir des masques plus subtils.

Deux hôtes, bien qu'appartenant au même réseau logique, s'ils sont placés dans des sous réseaux logiques différents, ne pourront communiquer entre eux que par l'intermédiaire d'un routeur. Cette solution est très commode pour des réseaux d'entreprise constitués de réseaux locaux distants et même pour des réseaux locaux comportant plusieurs centaines d'hôtes.

Les sur-réseaux

IPv4 est au bout du rouleau... Les adresses sont rares, les classes A ne sont plus disponibles, en classe B, pas grand chose de libre et les classes C sont exiguës. Que faire alors ? Par exemple créer un seul réseau logique avec plusieurs classes C contiguës. Dans ce cas, le masque de "sous réseau" sera un masque de "sur réseau" et définira un réseau avec plus d'hôtes qu'une classe C ne le permet.

Sur un réseau privé par exemple, nous pourrions prendre les deux classes C 192.168.0.0 et 192.168.1.0. En utilisant un masque de type 255.255.254.0, ceci nous permettra de réunir les deux classes C au sein d'un même réseau logique.

Bidouillage ? Probablement, mais ça fonctionne... Avec quelques restrictions cependant. Certaines piles IP n'accepteront pas les adresses 192.168.0.255 et 192.168.1.0 comme adresses d'hôtes valides (elles devraient être réservées dans un réseau "normal", nous l'avons vu, mais dans le cas d'un "sur réseau" constitué comme celui de l'exemple, il est logiquement possible de les utiliser).

Exercice:

- *A quel sous réseau appartient l'adresse 62.161.99.115 (SubnetID)?*

  Adresse IP `0011 1110 . 1010 0001 . 0110 0011 . 0111 0011`

  Masque de sous réseau : `1111 1111 . 1111 1111 . 1111 1000 . 0000 0000`

  Adresse du sous-réseau: (ET logique) `0011 1110 . 1010 0001 . 0110 0000 . 0000 0000`

  donc en décimal : `62.161.96.0`

  - L'opération consiste simplement en un ET logique bit à bit entre l'adresse et le masque. Mais on avait déjà la réponse en consultant les informations du client DHCP

- *Quelle est la partie de l'adresse qui concerne l'hôte (HostID)?*

  Adresse IP `0011 1110 . 1010 0001 . 0110 0011 . 0111 0011`

  Masque de sous réseau: (complément logique) `0000 0000 . 0000 0000 . 0000 0111 . 1111 1111`

  HostID: (ET logique) `0000 0000 . 0000 0000 . 0000 0011 . 0111 0011`

  donc en décimal : `0.0.3.115`

  - L'opération consiste ici en un ET logique entre l'adresse et le complément du masque

  Bien entendu, HostID + SubnetID doit reconstituer l'adresse IP, ce qui est bien le cas :
  (62.161.96.0) + (0.0.3.115) = 62.161.99.115

- *Quelle est la plus petite adresse possible dans ce sous réseau?*
  - SubnetID+1=62.161.96.1
    Qui est d'ailleurs l'adresse de la passerelle (c'est un choix de FTCI, pas une obligation. Toute adresse dans le même sous réseau aurait aussi bien fait l'affaire).
- *Quelle est la plus grande adresse possible dans ce sous réseau?*
  - C'est SubnetID+!SubnetMask-1
    Pourquoi? !SubnetMask-1 correspond à la plus grande HostID possible dans ce sous réseau, !SubnetMask correspondant à l'adresse de "l'hôte de broadcast"

  SubnetID : `0011 1110 . 1010 0001 . 0110 0000 . 0000 0000`

  !Masque de sous réseau-1 : `0000 0000 . 0000 0000 . 0000 0111 . 1111 1110`

  Plus grande adresse possible: (+) `0011 1110 . 1010 0001 . 0110 0111 . 1111 1110`

  donc en décimal : `62.161.103.254`

  L'opération est une somme binaire. Le résultat était prévisible, une fois encore, en regardant les informations du client DHCP. En effet; l'adresse de broadcast pour le sous réseau étudié est 62.161.103.255 (HostID avec tous les bits à 1).

C'est bien, n'est-ce pas, de pouvoir donner une explication rationnelle à tous ces paramètres IP plus ou moins obscurs à première vue...

Les sockets

Une oreille dans chaque port

Adresse, port et socket

Imaginons la situation suivante (fréquente sur des petits réseaux) :

Un seul "serveur" (entendez par là une machine) héberge plusieurs services bien connus des internautes :

- Un serveur web (HTTP)
- Un serveur de fichiers (FTP)
- Un serveur de messagerie (SMTP et POP3)

Tous ces services cohabitent donc sur un hôte disposant d'une seule adresse IP, disons 62.161.120.45 (pour fixer les idées) et fonctionnent sans problèmes. Vous êtes-vous posé la question de savoir par quel prodige tout ne se mélange pas? Comment se fait-il que le navigateur du client qui invoque l'URL http://62.161.120.45/default.html voit bien arriver la page demandée, alors que le client qui se connecte sur le serveur POP 62.161.120.45 va pouvoir y récupérer son courrier?

Plus fort encore, pendant qu'un client consulte la page http://62.161.120.45/default.html, un autre consulte http://62.161.120.45/sommaire.html. Et chaque client reçoit bien la page qu'il demande...

Grâce aux ports ! Les ports sont des numéros d'identification qui permettent de spécifier le service concerné. Ce numéro de port est écrit sur 2 octets, ce qui donne 65535 ports possibles (parce que le port 0 n'est, à ma connaissance, pas utilisé).

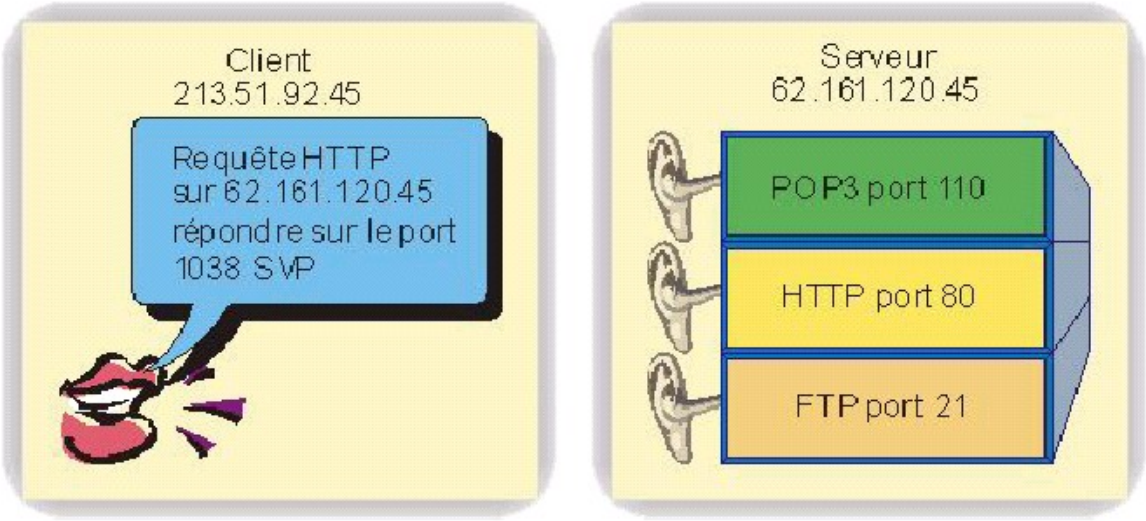
La combinaison "adresse IP:numéro de port " constitue ce que l'on appelle une "socket" (qui veut dire à peu près "connecteur" en anglais).

Une socket identifie pleinement le service qui est concerné sur une machine donnée.

Le serveur et le client

Les serveurs ont une fonction particulière : Ils doivent envoyer des informations pertinentes aux clients qui en réclament. Comme un serveur ne convient pas d'un rendez-vous avec le client, il doit rester attentif en permanence pour ne pas risquer de rater une question. Pour ce faire, on y installe des "daemons", petits programmes qui tournent en tâche de fond et qui écoutent continuellement sur un numéro de port donné. Il y a des conventions pour attribuer ces ports sur des services connus, par exemple le port 80 pour HTTP, le port 110 pour POP3, le port 21 pour FTP. Il faut qu'il y ait des conventions de ce genre pour que les clients puissent atteindre ces services.

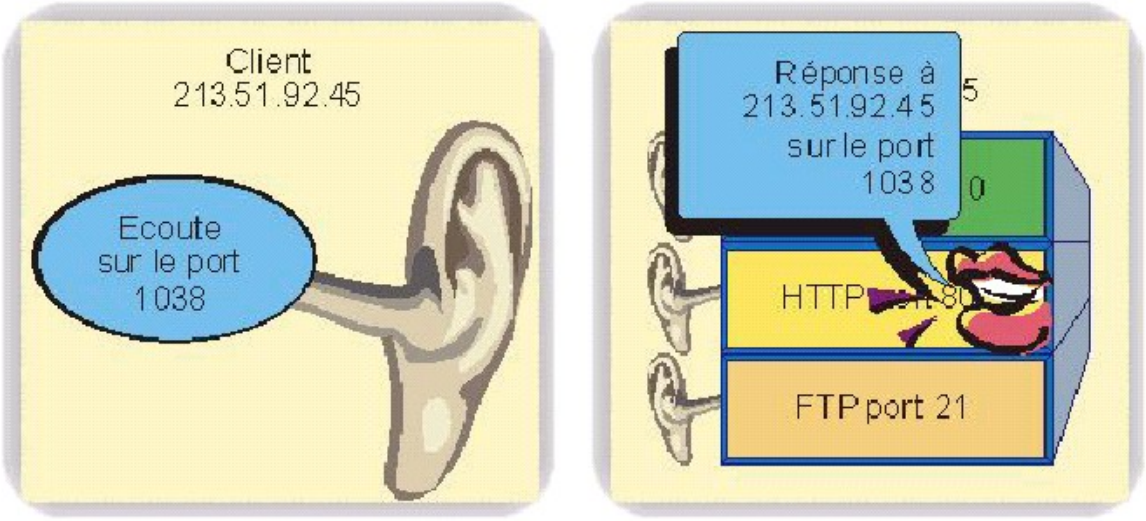
Lorsque l'on écrit http://62.161.120.45, on ne spécifie pas de port; sous-entendu, il s'agit du port 80 parce que l'on invoque un service HTTP. Il serait possible d'écrire: http://62.161.120.45:80 Ici, on spécifie le port. Certaines protections triviales consistent justement à forcer un service à ne pas employer le port standard. Un administrateur pourrait décider de mettre son serveur HTTP à l'écoute du port 88. Dans ce cas, si l'utilisateur n'est pas au courant de cette particularité, il ne pourra pas accéder à ce serveur (sauf s'il dispose d'un scanner de ports et qu'il découvre la supercherie).



En revanche, le client qui émet la requête ne dispose pas de port d'écoute attitré. Ce n'est pas un serveur, c'est un client; il n'a donc rien à écouter d'autre que les réponses à ses questions. Il faut donc, lorsqu'il envoie sa requête, qu'il spécifie sur quel port il va écouter la réponse, de manière à ce que le serveur puisse construire un socket efficace pour ladite réponse.

Vous êtes-vous demandé par quel miracle, si vous ouvrez deux fois votre navigateur pour afficher deux pages différentes sur le même serveur, les informations ne se mélangent pas ?

C'est parce que les deux sessions du navigateur indiquent des ports de réponse différents ! C'est le NOS du client qui choisit les ports de réponse en fonction de ceux qui sont disponibles sur la machine.



Un port d'écoute est une porte ouverte

Lorsqu'un port est ouvert à l'écoute sur un service serveur, c'est une porte ouverte par laquelle un intrus peut entrer.

Quelques infos supplémentaires

NAT, PAT et autres mascarades

Nous y reviendrons plus loin dans le chapitre consacré au routage[10], mais tant qu'on est dans les ports, autant dire quelques mots de ces techniques.

- NAT (Network Address Translation) est une faculté dont dispose un routeur, de modifier les adresses IP des émetteurs lors du passage des datagrammes entre deux réseaux. Ça ne nous intéresse pas directement ici.
- PAT (Port Access Translation) est une fonction qui permet de changer au passage le numéro de port dans le datagramme. Ca peut paraître tordu, mais il existe une foule d'applications possibles pour cette propriété.
- MASQUERADE, qui est un mélange des deux (NAT, PAT) est une fonction très intéressante pour connecter tout un réseau local construit sur une classe IP privée à l'Internet. La passerelle utilisera son IP publique (côté Internet) pour faire du NAT sur les adresses privées du réseau local et fera également du PAT pour savoir à qui il faudra transmettre les réponses.
    - Le principe de fonctionnement et la façon de construire une telle passerelle sont décrits dans la chapitre MASQUERADE[11], ailleurs sur ce site.

3.3. Outils logiciel

Des lignes de commande permettent d'obtenir des informations sur la carte réseau et votre adresse IP :

A partir d'une ligne de commande DOS :

Ping :

Un ping vers une adresse connue permet de vérifier que la connexion réseau fonctionne :

```
C:\WINDOWS>ping www.free.fr

Envoi d'une requête 'ping' sur www.free.fr [212.27.48.10] avec 32 octets de donn
ées :

Réponse de 212.27.48.10 : octets=32 temps=70 ms TTL=112
Réponse de 212.27.48.10 : octets=32 temps=65 ms TTL=112
Réponse de 212.27.48.10 : octets=32 temps=67 ms TTL=112
Réponse de 212.27.48.10 : octets=32 temps=68 ms TTL=112

Statistiques Ping pour 212.27.48.10:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en milli-secondes :
    minimum = 65ms, maximum =  70ms, moyenne =  67ms
```

Tracert :

Cette commande permet de connaître la route empruntée lors d'une connexion.
Exemple avec tracert www.google.fr

```
  1     1 ms     2 ms     1 ms  172.16.0.1
  2     3 ms     2 ms     3 ms  10.169.127.254
  3    59 ms    59 ms    60 ms  172.18.10.62
  4    59 ms    59 ms    60 ms  172.18.10.61
  5    61 ms    61 ms    59 ms  172.19.8.153
  6    64 ms    64 ms    59 ms  lyon-g2-0-0-700.cssi.renater.fr [193.51.184.122]

  7    71 ms    69 ms    66 ms  nri-b-pos9-0.cssi.renater.fr [193.51.179.13]
  8    68 ms    67 ms    66 ms  TELEGLOBE-FRANCE-INTERNATIONAL.sfinx.tm.fr [194.
68.129.242]
  9    66 ms    64 ms    65 ms  80.231.79.14
 10    75 ms    75 ms    74 ms  if-2-0.core2.FR1-Frankfurt.teleglobe.net [80.231
.65.65]
 11    78 ms    73 ms    76 ms  Vlan2.msfc1.FR1-Frankfurt.teleglobe.net [80.231.
66.4]
 12   137 ms    76 ms    79 ms  Vlan106.msfc1.FR1-Frankfurt.teleglobe.net [80.23
1.66.38]
 13    74 ms    75 ms    75 ms  72.14.238.124
 14    76 ms    78 ms    74 ms  72.14.232.207
 15    86 ms    83 ms    79 ms  72.14.232.190
 16    77 ms    75 ms    75 ms  www.l.google.com [72.14.221.99]

Itinéraire déterminé.
```

Ipconfig ou winipcfg :

L'utilisation de cette commande permet de connaître l'adresse IP de votre carte.

```
C:\WINDOWS>ipconfig

Configuration IP de Windows 98

0 - Carte Ethernet   :

        Adresse IP. . . . . . . . . . : 0.0.0.0
        Masque de sous-réseau . . . : 0.0.0.0
        Passerelle par défaut . . . :

1 - Carte Ethernet   :

        Adresse IP. . . . . . . . . . : 172.16.150.227
        Masque de sous-réseau . . . : 255.255.0.0
        Passerelle par défaut . . . : 172.16.0.1
```

4. DNS (DOMAIN NAME SYSTEM)

4.1. Introduction

Tous les internautes vous le diront, l'URL (ou URI) est le gouvernail de la navigation sur le Net.
Ça fait déjà au moins trois sigles à expliquer :
FQDN : Full Qualified Domain Name
Le nom complet d'un hôte, sur l'Internet, c'est-à-dire de la machine jusqu'au domaine, en passant par les sous-domaines.
URL : Uniform Resource Locator

C'est la méthode d'accès à un document distant. Un lien hypertexte avec une syntaxe de la forme:

<Type de connexion>://<FQDN>/[<sous-répertoire>]/.../<nom du document>

Exemple: http://www.ac-aix-marseille.fr/bleue/francais/nouveau.htm

- http: Hyper Text Transfert Protocol
- www.ac-aix-marseille.fr : FQDN du serveur de pages personnelles
- /bleue/francais/ : arborescence de répertoires
- nouveau.htm : nom du document.

URI : Universal Resource Identifier.
On ne va pas chipoter, c'est la même chose que l'URL. Le W3C (World Wide Web Consortium), garant de l'universalité de l'Internet, voudrait voir abandonner URL au profit d'URI. Notez la très subtile divergence de sens, qui vaut bien, le changement.

Donc, ne confondons pas tout, un FQDN est significatif d'un hôte sur l'Internet (un serveur la plupart du temps), alors qu'un URI définit l'accès à un document sur un serveur. L'URI contient donc un FQDN, mais pas seulement.

4.2. Les DNS

Les serveurs DNS sont là pour réaliser cette opération et l'inverse également (retrouver un nom à partir d'une adresse IP). Votre fournisseur d'accès met à votre disposition un serveur de ce type, dont l'adresse IP vous est habituellement donnée de façon automatique lors de la transaction DHCP (voir le chapitre à ce sujet1), ou via le protocole PPP.
Nous allons ici décortiquer le fonctionnement d'un tel serveur et même voir comment l'on peut s'en construire un personnel, aussi efficace (sinon plus) que celui de votre FAI.

4.3. Outils

Afin de visualiser l'IP d'un FQDN on peut utiliser la ligne de commande «nslookup» : exemple :
Lancer l'invite de commande dos et tapez :
nslookup www.google.fr
Il apparaît l'IP de www.google.fr : 64.233.183.147 au moment ou j'ai fais la demande !.
Et inversement : tapez nslookup 64.233.183.147 et vous verrez apparaître le FQDN correspondant !

5. DHCP (DYNAMIC HOST CONTROL PROTOCOL)

Ce protocole permet aux administrateurs de réseaux TCP/IP de configurer les postes clients de façon automatique. Il a été utilisé par les fournisseurs d'accès à l'Internet par le câble, mais a été abandonné au profit d'une connexion point à point type PPP, comme pour l'ADSL.
DHCP reste cependant un protocole de configuration de clients extrêmement pratique sur un réseau local Ethernet.
Bien que dans la plupart des cas, DHCP soit un luxe sur un réseau domestique, il peut tout de même y avoir plusieurs raisons pour vous pousser à l'utiliser :

- Vous avez des portables que vous connectez sur divers réseaux, typiquement chez vous et sur votre lieu de travail (si votre administrateur vous laisse faire, c'est qu'il est bien confiant :-)),
- vous organisez chez vous des "Lan parties" avec les machines de vos collègues,
- votre réseau local contient plusieurs dizaines de machines (vous avez une famille nombreuse, peut-être),
- vous aimez bien vous compliquer la vie à bricoler avec votre Linux,
- vous aimez le luxe, tout simplement.

6. GLOSSAIRE

ADSL Technologie en cours d'élaboration permettant, sur une ligne téléphonique analogique, de passer des données numériques à grande vitesse. Le résultat devrait être meilleur et moins onéreux que l'ISDN (ou RNIS ou NUMERIS)

API "Application Programming Interface", Ou encore "Interface de programmation pour les applications" Ce sont les points d'entrée qu'un O.S.(ou un N.O.S.) proposent aux programmeur pour gérer les ressources des machines depuis leurs applications, comme par exemple: créer un fichier, l'enregistrer, le lire, l'imprimer sur une imprimante du réseau...

DHCP Système serveur d'adresses IP.

Un serveur DHCP permet d'attribuer dynamiquement une adresse IP à une machine lorsque cette dernière se connecte au réseau. C'est un outil très pratique pour l'administrateur qui n'a plus à se préoccuper des attributions d'adresses IP à chaque machine du réseau.

DNS "Domain Name System", C'est un serveur capable de trouver une adresse IP en connaissant le nom d'une machine dans un domaine internet. Par exemple il sait chez nous que www.eme.org correspond à l'adresse 192,168,0,1.

Les serveurs DNS sont un maillon fondamental dans l'Internet, c'est grâce à eux que les URL sont exploitables.

FAT "File Allocation Table".

Système de gestion des mémoires de masse, mis en place avec MS DOS. D'abord FAT12 (pour les disquettes) puis FAT16 pour les disques durs; enfin FAT32 avec Windows 95 OSR2 et Windows 98.

Le système FAT souffre de vieillesse et de replâtrages successifs. FAT16 ne sait pas gérer de partitions supérieures à 2Go, alors que les plus petits disques actuels font 3Go!

La FAT32 sait aller plus loin, mais ce n'est qu'un bricolage sur un concept dépassé. Microsoft s'en débarrassera sans doute très prochainement.

A noter que Windows NT 4,0 sait manipuler des volumes en FAT16 mais pas en FAT32, que Windows 95 ne sait pas utiliser la FAT32 (sauf la version OSR2), que windows 98 sait utiliser les FAT16 et FAT32 (mais pas NTFS) et que Windows 2000 sait tout faire...

FIRMWARE Logiciel spécifique très proche du matériel et généralement embarqué dans une mémoire morte (ROM, EPROM, EEPROM).

ICMP

Internet Control Message Protocol.

En termes de sécurité, ce protocole fait peur à beaucoup de monde (parfois à juste titre d'ailleurs), il est cependant fondamental pour le bon fonctionnement de l'Internet. C'est grâce à ce protocole que les anomalies de fonctionnement peuvent être signalées à l'émetteur, afin qu'il puisse essayer d'y remédier.

IPX/SPX Protocole de communication réseau développé par la société NOVELL pour son N.O.S. "Novell Netware". Quelque peu inspiré de TCP/IP mais incompatible avec lui, Il est plus efficace que NetBEUI, mais n'a actuellement plus aucune chance de survie visàvis de TCP/IP qui est devenu incontournable avec l'INTERNET.
ISDN "Integrated Services Digital Network" ou RNIS : Réseau Numérique à Intégration de Services. En France "NUMERIS". Cest un réseau initialement à vocation téléphonique, transportant les informations de manière numérique, avec des services ajoutés comme la possibilité de sous adresser plusieurs terminaux sur la même ligne.
FRANCE TELECOM a vendu ce service tellement cher que presque personne n'en a voulu. Actuellement, F.T. a baissé ses prix, mais la technologie est dépassée... Il reste que c'est un bon moyen de réaliser une connexion informatique à 64Kb/s (ou plus en utilisant plusieurs canaux).
MMDS Système de distribution d'informations numériques dans un seul sens par voie hertzienne. L'interactivité ne pouvant se faire que par un retour sur ligne téléphonique. Intéressant pour les agglomérations ne pouvant pas encore se payer le "câble"
N.O.S. "Network Operating System", autrement dit "système d'exploitation réseau. Autrefois, N.O.S. et O.S. étaient séparés, ce qui rendait le tout très incertain. Actuellement la tendance est à fondre les deux en un seul produit.
La famille Windows depuis la version 3,11 est en fait un N.O.S. UNIX également.
NTFS "New Technology File System".
Système de gestion des disques durs utilisé par Windows NT. A comparer au système FAT du bon vieux DOS (pour les fonctionnalités mais pas pour les performances!)
O.S. "Operating System". Autrement dit "Système d'Exploitation", c'est le logiciel qui permet de faire fonctionner les ressources matérielles d'une machine. Les applications s'appuyant dessus pour fonctionner.
Tout le monde connaît MS DOS, Windows 95, Windows 98 et Windows NT qui sont les systèmes d'exploitation Microsoft. Mais il y a aussi LINUX, qui est un système UNIX gratuit. (et beaucoup d'autres encore, suivant les plateformes matérielles).
Bien entendu, les applications sont écrites non seulement pour une plateforme matérielle donnée, mais encore pour un OS donné.
OSI "Open System Interconnection". C'est un modèle théorique de l'architecture d'un N.O.S. destiné à permettre la communications entre deux systèmes diférents (Normalement, UNIX doit pouvoir communiquer avec Windows NT, si les deux systèmes suivent les recommandationsOSI.

PCL Langage de description de page développé par Hewlett Packard pour ses imprimantes. C'est actuellement devenu un standard. Il est moins puissant que POSTSCRIPT, mais beaucoup plus "léger". La puissance actuelle
des machines et des imprimantes à permis une deuxième génération de ce langage
dont les performances en termes de qualité avoisinent celles de POSTSCRIPT.
POSTSCRIPT Langage de description de document développé par la société ADOBE. C'est un langage très puissant qui est utilisé pour décrire l'apparence d'un document sans tenir compte des caractéristiques de l'imprimante.
Cette dernière dispose d'un interpréteur spécifique pour composer et imprimer le document. C'est un procédé très précis mais qui coute cher en ressources.
RNIS Voir ISDN
TCP/IP "Transfer Control Protocol/Internet Protocol" ou protocole de contrôle de transfert/protocole internet.
Définit deux couches du N.O.S. connu sous le même nom.
WINS C'est un système propre aux réseaux Microsoft qui fait à peu près la même chose que les DNS, à part qu'ici, on est entre nous (tout Microsoft), alors les choses sont plus faciles. WINS est un service Windows NT qui tient automatiquement à jour une table d'équivalence entre les noms des machines et leur adresse sur le réseau.Le nommage se fait "à plat" Il n'y a pas de hiérarchie comme dans DNS. WINS est abandonné dans Windows 2000 et suivants au profit d'une structure compatible DNS