Architecture OSI et Internet
Protocoles TCP/IP et DNS

Plan

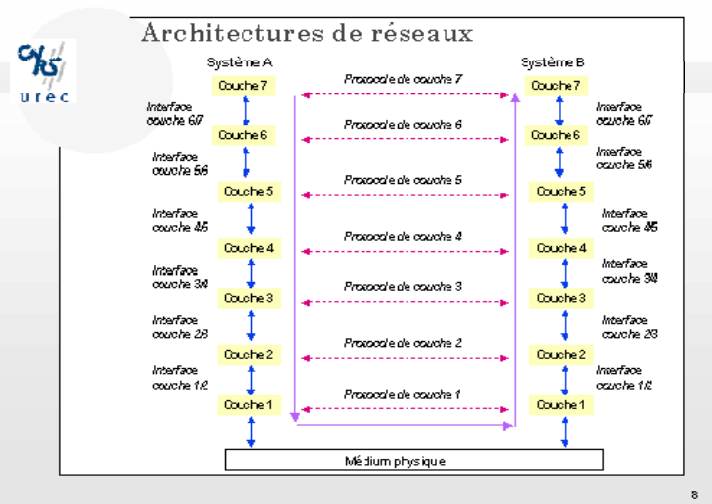
- **ARCHITECTURE OSI**
  - INTRODUCTION
  - GESTION DE RESEAUX PHYSIQUES
  - GESTION DE RESEAUX LOGIQUES
  - GESTION DES APPLICATIONS EN RESEAUX
- **PROTOCOLES TCP/IP**
  - CONCEPTS DE L'INTERCONNEXION
  - L'ADRESSAGE INTERNET
  - PROTOCOLES DE RESOLUTION D'ADRESSE
  - UDP : TRANSPORT DATAGRAM
  - TCP : TRANSPORT FIABLE
- **DOMAIN NAME SERVER**
  - INTRODUCTION
  - ESPACE NOM DE DOMAINE
  - NOM DE DOMAINE
  - DOMAINE
  - SERVEUR DE NOM
  - UTILISATION DE DNS
- **SERVICES INTERNET**
  - MODELE LIENT/SERVEUR
  - TELNET
  - FTP
  - COURRIER ELECTRONIQUE
  - HTTP
  - APACHE

ARCHITECTURE O.S.I.
INTRODUCTION

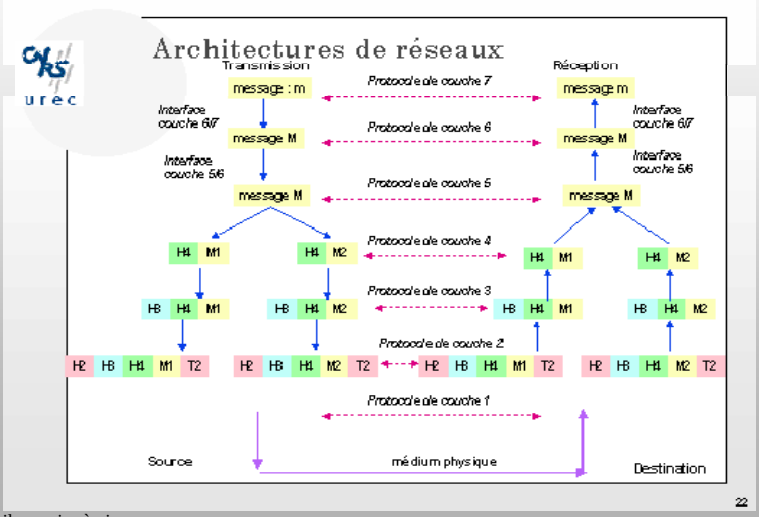
- **O.S.I. = OPEN SYSTEM INTERCONNECTION**
  - Modèle fondé sur un principe énoncé par Jules César « Diviser pour mieux régner »
  - Le principe de base est la description des réseaux sous forme d'un ensemble de couches superposées les unes aux autres
  - L'étude du tout est réduit à celle de ses parties
  - L'ensemble devient plus facile à construire et à manipuler

- **Deux organisations de normalisation pour les réseaux informatiques**
  - ISO (International Standardization Organization) dépendante de l'ONU avec les représentants nationaux : ANSI pour les USA, AFNOR pour la France, DIN pour l'Allemagne, BSI pour le Royaume Uni, HSC pour le Japon, …
  - UIT-T (Union Internationale des Télécommunication) comprend des opérateurs et des industriels des télécommunications

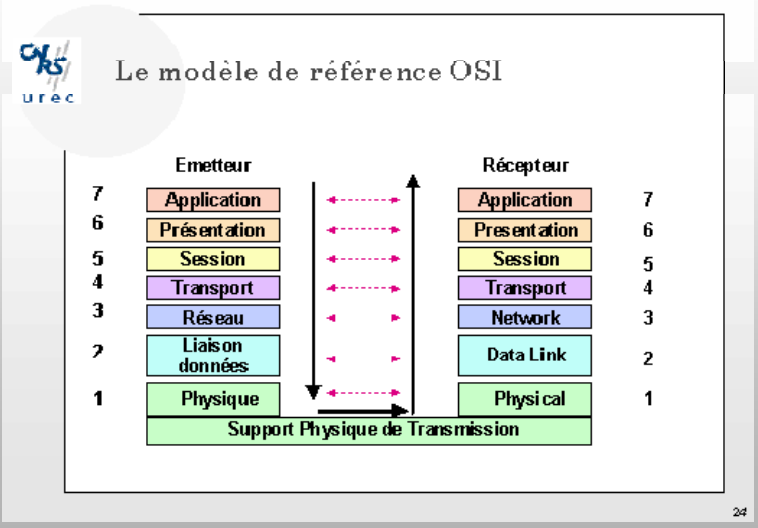
ARCHITECTURE O.S.I.
INTRODUCTION



ARCHITECTURE O.S.I.
INTRODUCTION

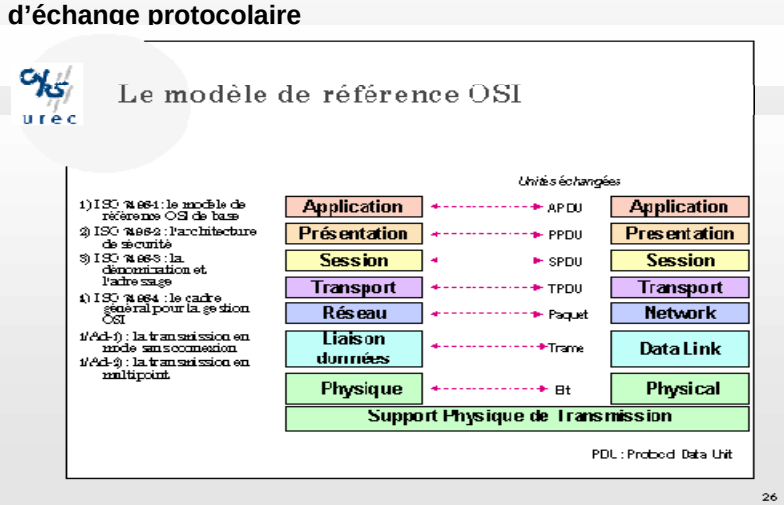


ARCHITECTURE O.S.I.
INTRODUCTION



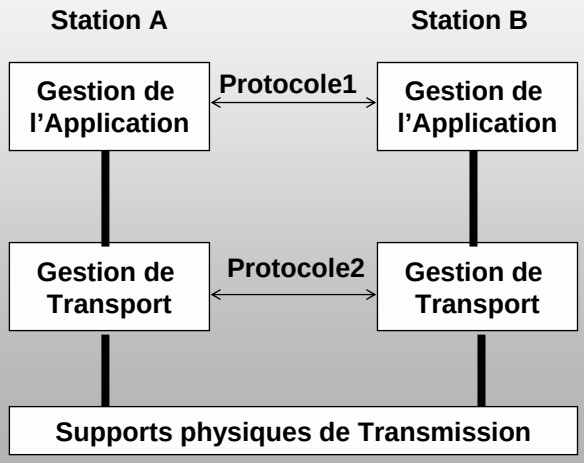
ARCHITECTURE O.S.I.
INTRODUCTION

- **Unité d'échange protocolaire**



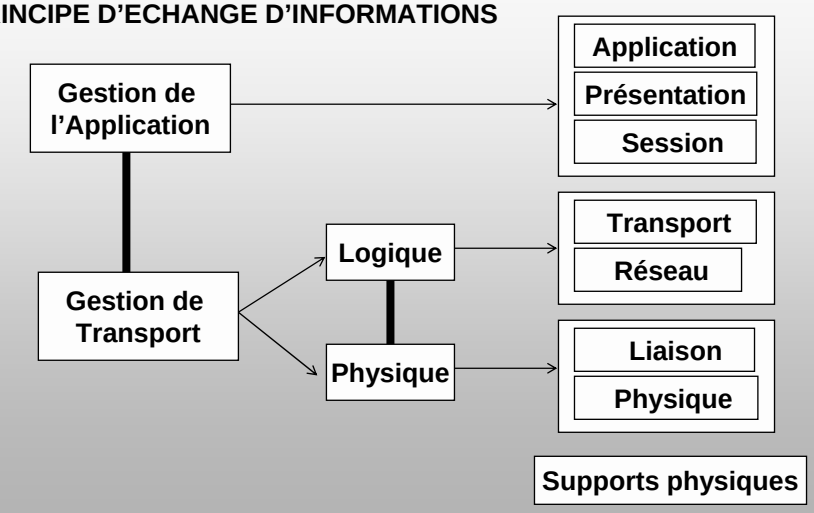
ARCHITECTURE O.S.I.
INTRODUCTION

- **PRINCIPE D'ECHANGE D'INFORMATIONS**



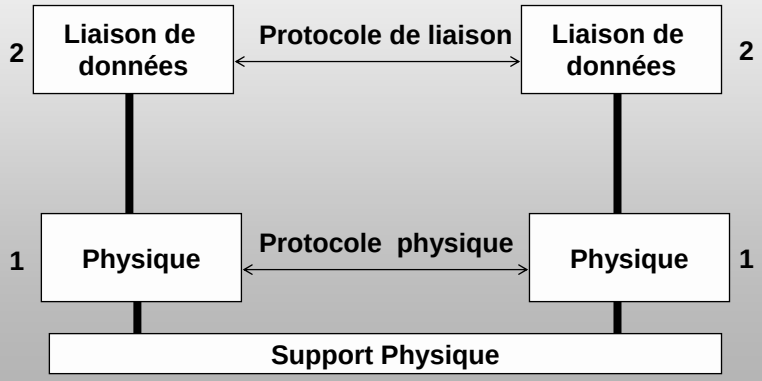
ARCHITECTURE O.S.I.
INTRODUCTION

- **PRINCIPE D'ECHANGE D'INFORMATIONS**



ARCHITECTURE O.S.I.
GESTION DES RESEAUX PHYSIQUES

- **Gestion de transport de données entre stations d'un réseau physique**



ARCHITECTURE O.S.I.
GESTION DES RESEAUX PHYSIQUES

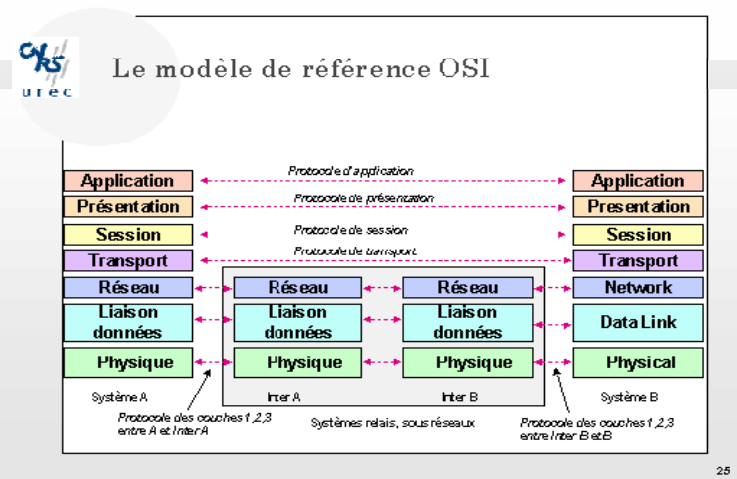
- **COUCHE PHYSIQUE ET SUPPORTS PHYSIQUE**
  - La norme ISO 10022 et la recommandation X.211 de l'UIT définit le service qui doit être rendu
  - Elle fournit les moyens mécaniques, électriques, fonctionnels, à l'activation, au maintien et à la désactivation des connexions physiques destinées à la transmission des éléments binaires entre entités de liaisons
  - La transmission est effectuée comme une séquence des bits sur un circuit de communication
  - Eléments de la couche physique :
    - Support physique (hertzien, électromagnétique, laser)
    - Codeur, MODEM (MOdulation et DEModulation)
    - Multiplexeurs, Concentrateurs (HUB, SWITCH)
  - La conception de la couche physique peut être réellement considérée comme faisant partie du domaine de l'ingénierie électronique

ARCHITECTURE O.S.I.
GESTION DES RESEAUX PHYSIQUES

- **COUCHE LIAISON DE DONNEES**
  - Utilisation de la couche physique
  - Gestion de la liaison de données
    - Rassembler les données de l'émetteur en « TRAME DE DONNEES »
    - Transmettre les trames en séquence
    - Gestion des trame d'acquittement
    - Reconnaissance des frontières de trame envoyées par la couche physique
  - Détection et reprise sur erreur
    - Régulation du trafic
    - Gestion des erreurs
  - Gestion de l'allocation du support physique (méthodes d'accès) dans le réseaux locaux (ETHERNET, TOKEN RING, ATM, …) et procédures de transmission
  - La norme ISO 8886 ou la recommandation UIT X.212 définit le service fourni par la couche 2

ARCHITECTURE O.S.I.
GESTION DES RESEAUX LOGIQUES

- **Transport de données à travers de plusieurs réseaux physiques**



ARCHITECTURE O.S.I.
GESTION DES RESEAUX LOGIQUES

- **COUCHE RESEAU**
  - Interconnexion des réseaux physiques hétérogènes dans un réseau logique unique
  - fournir les moyen d'établir de maintenir et de libérer des connexions de réseau entre des systèmes ouverts
    - Gestion du sous-réseau
    - Acheminement des parquets de source vers la destination à travers des réseaux physiques
  - Fonctionnalités
    - Adressage logique
    - Routage des paquets
    - Contrôle de flux
  - Plusieurs protocoles : IP, X25, Frame relais, …

ARCHITECTURE O.S.I.
GESTION DES RESEAUX LOGIQUES

- **COUCHE TRANSPORT**
  - Gestion de liaison de données entre l'émetteur et le récepteur à travers des réseaux physique
  - Indépendance des réseaux sous-jacents
    - Les découper éventuellement
    - S'assurer de l'ordonnancement
  - Optimisation des ressources réseaux
  - Fonctionnalités de bout en bout :
    - Multiplexage de plusieurs messages sur un canal
    - Nécessité d'indiquer quel message appartient à quelle connexion
  - Dépendance du service réseau (Q°S)
  - Protocoles de transport :
    - TP0, 1, 2, 3, 4
    - TCP, UDP

ARCHITECTURE O.S.I.
GESTION DE L'APPLICATION

- **Gestion de synchronisation**
- **Gestion des terminaux**
- **Gestion des application**

ARCHITECTURE O.S.I.
GESTION DE L'APPLICATION

- **COUCHE SESSION (5)**
  - Responsable de la synchronisation
  - Fonction de type :
    - Gestion du dialogue (bi- ou unidirectionnel)
    - Points de reprise,
    - Retour arrière
    - Cohérence
  - Orchestration des échanges de données entre les applications
  - Gestion des transactions

ARCHITECTURE O.S.I.
GESTION DE L'APPLICATION

- **COUCHE PRESENTATION (6)**
  - Syntaxe et sémantique des information
    - Représentation des données transférées entre entités d'application
    - Représentation de la structure de données et représentation de l'ensemble des actions effectuées sur cette structure de données
    - Encodage dans la norme agrée permettant à des équipement « ASCII » et « EBCDIC » par exemple de communiquer
    - Compression des données, chiffrement
  - Exemple : la syntaxe abstraite ASN.1 (ISO 8824, UIT X208) normalisée par l'ISO est utilisée dans la messagerie X400 et les annuaires X500

ARCHITECTURE O.S.I.
GESTION DE L'APPLICATION

- **COUCHE APPLICATION (7)**
  - **Elle offre aux processus d'application le moyen d'accéder à l'environnement OSI**
  - **Les processus d'application échangent leurs informations par l'intermédiaire des entités d'application**
  - **Exemple : le terminal de réseau virtuel transfert de fichiers, courrier électronique, consultation des annuaires, consultation web**

TCP/IP et INTERNET
Introduction

- **TCP/IP : but = interconnexion de réseaux sur une base planétaire**
- **Technologie issue des années 1970, de projets DARPA**
- **Interconnecte divers réseaux : Ethernet, T.R., X25, FR, FDDI, etc.**
- **La technologie est constituée par des protocoles de base (suite TCP/IP) qui offrent les services de base du transfert des données**
- **transport de datagrammes : service élémentaire de la commutation de paquets.**
- **transport de messages sécurisés : service orienté connexion permettant d'acheminer des données en garantissant leur intégrité**
- **adaptation de la technologie TCP / IP à la plupart des interfaces matérielles.**
- **Ces services de base sont indépendants du support de transmission; adaptables à toute sorte de media depuis les réseaux locaux jusqu'aux réseaux longue distance**
- **IP 4 et IP 6**

TCP/IP et INTERNET
Introduction

- **Interconnexion universelle : les machines ont une adresse unique sur l'Internet.**
- **Interconnexion d'égal à égal (peer to peer systems) : il n'y a pas de machines prioritaires (en opposition à une structure hiérarchique).**
- **Dans le cadre du transport sécurisé, les acquittements sont effectués entre les systèmes finaux (source et destinataire) plutôt que continuellement entre chaque noeud relayant les messages.**
- **Applications standards bâties sur la technologie de base : courrier électronique, transfert de fichier, émulation terminal, etc.**
- **Technologie publique et largement diffusée au travers de RFC's.**
- **Indépendante des constructeurs et disponible sur tous types de matériel (micro, station, super-calculateur et équipements de réseaux)**
- **Largement validée depuis de nombreuses années dans un monde hétérogène.**

TCP/IP et INTERNET
Concepts de l'interconnexion

- **<u>Point de départ</u> : les réseaux interconnectés sont de nature diverse**
- **Les différences entre tous ces réseaux ne doivent pas apparaître à l'utilisateur de l'interconnexion.**
- **Abstraction à chaque niveau de fonctionnalité (couches de protocoles) qui encapsule les fonctionnalités de niveau inférieur**
- **Les premiers systèmes d'interconnexion ont traité le problème au niveau applicatif : messagerie relayant le message de noeud en noeud. Cette solution présente plusieurs inconvénients :**
- **si les applications interfacent elles-mêmes le réseau (aspects physiques), elles sont victimes de toute modification de celui-ci,**
- **plusieurs applications différentes sur une même machine dupliquent l'accès au réseau,**
- **lorsque le réseau devient important, il est impossible de mettre en oeuvre toutes les applications nécessaires à l'interconnexion sur tous les noeuds des réseaux.**

TCP/IP et INTERNET
Concepts de l'interconnexion

- **Alternative à cette solution : mise en oeuvre de l'interconnexion au niveau des protocoles gérant la couche réseau de ces systèmes.**
- **Avantage considérable : les données sont routées par les noeuds intermédiaires sans que ces noeuds aient la moindre connaissance des applications responsables des ces données**
- **Autres avantages :**
  - **la commutation est effectuée sur la base de paquets de petite taille plutôt que sur la totalité de fichiers pouvant être de taille très importante,**
  - **le système est flexible puisqu'on peut facilement introduire de nouveaux interfaces physiques en adaptant la couche réseau alors que les applications demeurent inchangées,**
  - **les protocoles peuvent être modifiés sans que les applications soient affectées.**

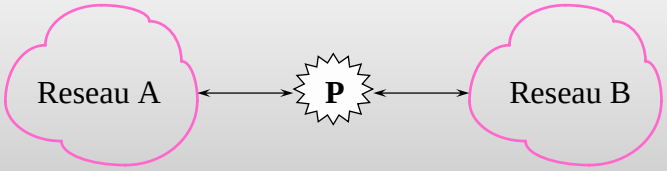
TCP/IP et INTERNET
Concepts de l'interconnexion

- **Le concept d'interconnexion ou d'*internet* repose sur la mise en oeuvre d'une couche réseau masquant les détails de la communication physique du réseau et détachant les applications des problèmes de routage.**
- **L'interconnexion : faire transiter des informations depuis un réseau vers un autre réseau par des noeuds spécialisés appelés passerelles (*gateway*) ou routeurs (*router*)**

TCP/IP et INTERNET

Concepts de l'interconnexion

- **Les routeurs possèdent une connexion sur chacun des réseaux:**



La passerelle P interconnecte les réseaux A et B.

- **Le rôle de la passerelle P est de transférer sur le réseau B, les paquets circulant sur le réseau A et destinés au réseau B et inversement.**

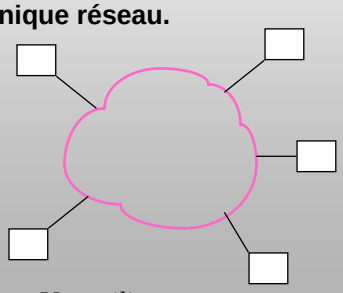
TCP/IP et INTERNET

Concepts de l'interconnexion

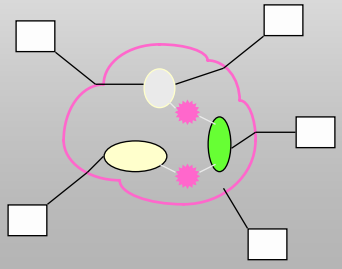
- **P1 transfère sur le réseau B, les paquets circulant sur le réseau A et destinés aux réseaux B et C**
- **P1 doit avoir connaissance de la topologie du réseau; à savoir que C est accessible depuis le réseau B.**
- **Le routage n'est pas effectué sur la base de la machine destinataire mais sur la base du réseau destinataire**

TCP/IP et INTERNET
Adressage

- **A l'intérieur de chaque réseau, les noeuds utilisent la technologie spécifique de leur réseau (Ethernet, X25, etc)**
- **Le logiciel d'interconnexion (couche réseau) encapsule ces spécificités et offre un service commun à tous les applicatifs, faisant apparaître l'ensemble de ces réseaux disparates comme un seul et unique réseau.**



Vue utilisateur



Vue réelle du réseau

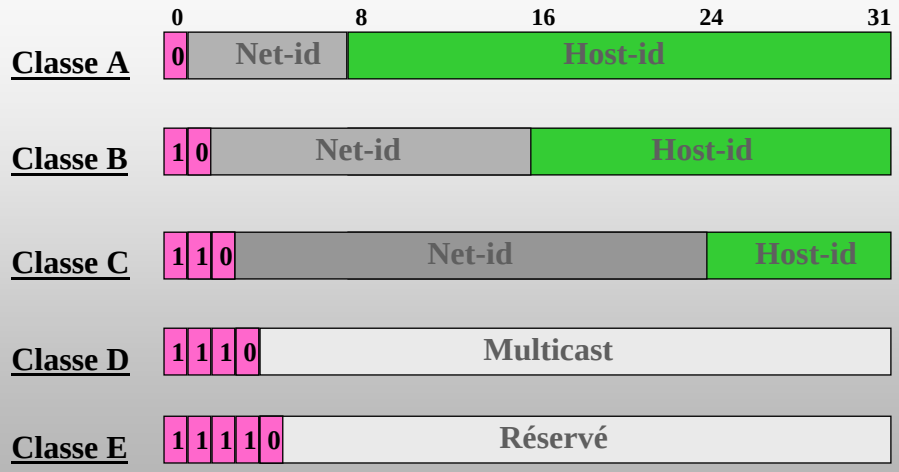
TCP/IP et INTERNET
Adressage

- **But : fournir un service de communication universel permettant à toute machine de communiquer avec toute autre machine de l'interconnexion**
- **Une machine doit être accessible aussi bien par des humains que par d'autres machines**
- **Une machine doit pouvoir être identifiée par :**
  - **un nom (mnémotechnique pour les utilisateurs),**
  - **une adresse qui doit être un identificateur universel de la machine,**
  - **une route précisant comment la machine peut être atteinte.**

TCP/IP et INTERNET
Adressage

- **Solution : adressage binaire compact assurant un routage efficace**
- **Adressage "à plat" par opposition à un adressage hiérarchisé permettant la mise en oeuvre de l'interconnexion d'égal à égal**
- **Utilisation de noms pour identifier des machines (réalisée à un autre niveau que les protocoles de base)**
- **Les classes d'adressage**
  - **Une adresse = 32 bits dite "internet address" ou "IP address" constituée d'une paire (netid, hostid) où netid identifie un réseau et hostid identifie une machine sur ce réseau.**
  - **Cette paire est structurée de manière à définir cinq classes d'adresse**

TCP/IP et INTERNET
Adressage



TCP/IP et INTERNET
Adressage

- **Notation décimale**

  L'interface utilisateur concernant les adresses IP consiste en la notation de quatre entiers décimaux séparés par un point, chaque entier représentant un octet de l'adresse IP :

  11000000 00001011 00000110 00011111 est écrit : 192.11.7.31

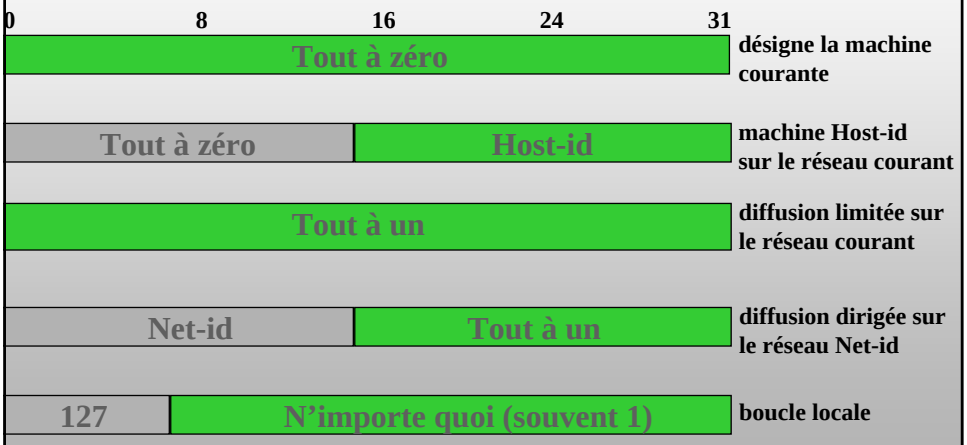
- **Adresses particulières**
  - **Adresses réseau : adresse IP dont la partie hostid ne comprend que des zéros; => la valeur zéro ne peut être attribuée à une machine réelle : 134.59.0.0 désigne le réseau de classe B 134.59.**
  - **Adresse machine locale : adresse IP dont le champ réseau (netid) ne contient que des zéros;**
  - **hostid = 0 (=> tout à zéro), l'adresse est utilisée au démarrage du système afin de connaître l'adresse IP (Cf RARP).**
  - **hostid != 0, hostid spécifie l'adresse physique de la machine (si la longueur le permet; c'est le cas pour T. R., ce n'est pas possible avec Ethernet). permet de ne pas utiliser RARP (ne franchit pas les ponts) n'est valide qu'au démarrage du système pour des stations ne connaissant pas leur adresse IP.**

TCP/IP et INTERNET
Adressage

- **Adresses de diffusion : la partie hostid ne contient que des 1**
- **Adresse de diffusion limitée : netid ne contient que des 1 : l'adresse constituée concerne uniquement le réseau physique associé**
- **L'adresse de diffusion dirigée : netid est une adresse réseau spécifique => la diffusion concerne toutes les machines situées sur le réseau spécifié : 134.59.255.255 désigne toutes les machines du réseau 134.59.**
- **En conséquence, une adresse IP dont la valeur hostid ne comprend que des 1 ne peut être attribuée à une machine réelle.**
- **Adresse de boucle locale : l'adresse réseau 127.0.0.0 est réservée pour la désignation de la machine locale, c'est à dire la communication intra-machine. Une adresse réseau 127 ne doit, en conséquence, jamais être véhiculée sur un réseau et un routeur ne doit jamais router un datagramme pour le réseau 127.**

TCP/IP et INTERNET
Adressage

☞ **Résumé**



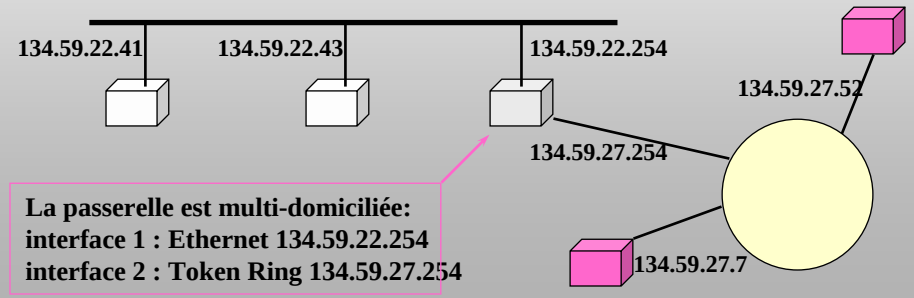
TCP/IP et INTERNET
Adressage

☞ **Adresses et connexions**

Une adresse IP => une interface physique => une connexion réseau.

S'applique particulièrement aux routeurs qui possèdent par définition plusieurs connexions à des réseaux différents

A une machine, est associé un certain nombre N d'adresses IP. Si N > 0 la machine (ou passerelle) est multi-domiciliée.

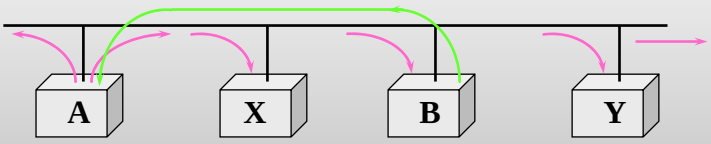


TCP/IP et INTERNET
ARP: Address Resolution Protocol

- **Le besoin**
  - **La communication entre machines ne peut s'effectuer qu'à travers l'interface physique**
  - **Les applicatifs ne connaissant que des adresses IP, comment établir le lien adresse IP / adresse physique?**
- **La solution : ARP**
  - **Mise en place dans TCP/IP d'un protocole de bas niveau appelé Adress Resolution Protocol (ARP)**
  - **Rôle de ARP : fournir à une machine donnée l'adresse physique d'une autre machine située sur le même réseau à partir de l'adresse IP de la machine destinatrice**
- **La technique :**
  - **Diffusion d'adresse sur le réseau physique**
  - **La machine d'adresse IP émet un message contenant son adresse physique**
  - **Les machines non concernées ne répondent pas**
  - **Gestion cache pour ne pas effectuer de requête ARP à chaque émission**

TCP/IP et INTERNET
ARP: Address Resolution Protocol

- **L'association adresse physique - adresse IP de l'émetteur est incluse dans la requête ARP de manière à ce que les récepteurs enregistrent l'association dans leur propre mémoire cache**



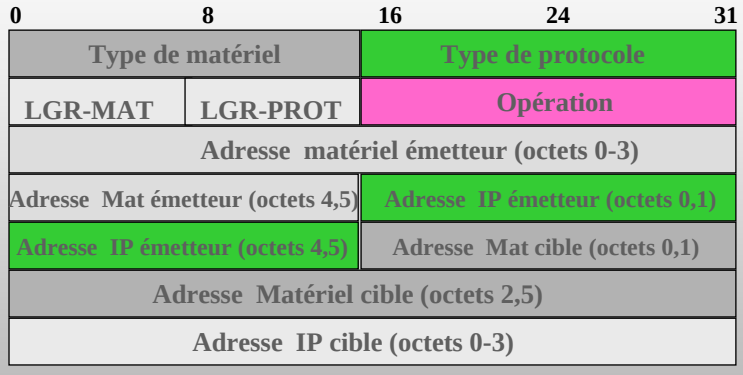
- **Pour connaître l'adresse physique de B, PB, à partir de son adresse IP IB, la machine A diffuse une requête ARP qui contient l'adresse IB vers toutes les machines; la machine B répond avec un message ARP qui contient la paire (IB, PB).**

TCP/IP et INTERNET
ARP: Address Resolution Protocol

- **Format du message ARP**

- **La requête ARP est véhiculée dans un message protocolaire lui-même encapsulé dans la trame de liaison de données.**
- **Lorsque la trame arrive à destination, la couche liaison de données détermine l'entité responsable du message encapsulé; Ex: champ type de la trame Ethernet: 0806 pour ARP**
- **La structure du message ARP/RARP gère une association adresse de protocole / adresse physique indépendamment de l'interface physique et du protocole utilisé :**

TCP/IP et INTERNET
ARP: Address Resolution Protocol



Autre technique : proxy Arp

TCP/IP et INTERNET

RARP: ReverseAddress Resolution Protocol

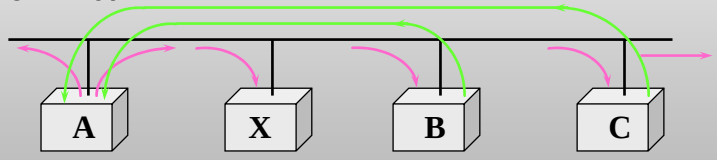
- **Le besoin**
  - **L'adresse IP d'une machine est configurable (elle dépend du réseau sur lequel elle se trouve) et est souvent enregistrée sur la mémoire secondaire où le système d'exploitation l'accède au démarrage.**
  - **Ce fonctionnement usuel n'est plus possible dès lors que la machine est une station sans mémoire secondaire.**
- **Problème : déterminer un mécanisme permettant à la station d'obtenir son adresse IP depuis le réseau.**
- **La solution**
  - **Protocole de bas niveau appelé Reverse Adress Resolution Protocol**
  - **Permet d'obtenir son adresse IP à partir de l'adresse physique qui lui est associée.**
- **Fonctionnement**

  Serveur RARP sur le réseau physique; son rôle: fournir les adresses IP associées aux adresses physiques des stations du réseau;

TCP/IP et INTERNET

RARP: ReverseAddress Resolution Protocol

- **Le serveur possède une base de données contenant les couples adresse physique/adresse IP,**
- **les stations émettent une requête RARP sur le réseau, consistant à demander l'adresse IP qui est associée à leur adresse physique,**
- **Les requêtes RARP sont propagées vers le ou les serveur(s) RARP par mécanisme de diffusion. Le(s) serveur(s) RARP réponde(nt) par un message de type RARP.**



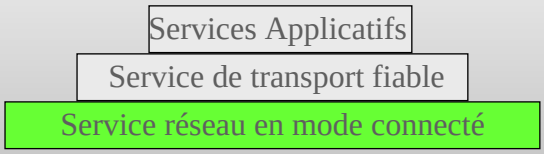
Pour conaître son adresse IP, A diffuse sur le réseau, une requête RARP qui la désigne comme destinataire

Les Serveurs RARP (B et C) répondent à la requête.

TCP/IP et INTERNET

IP : Internet Protocol

- **Le protocole Internet (Internet Protocol ou IP) :**
  - **réalise les fonctionnalités de la couche réseau selon le modèle OSI**
  - **se situe au coeur de l'architecture TCP/IP qui met en oeuvre un mode de transport fiable (TCP) sur un service réseau en mode non connecté :**



- **Le service offert par le protocole IP est dit non fiable :**
  - **remise de paquets non garantie,**
  - **sans connexion (paquets traités indépendamment les uns des autres),**
  - **pour le mieux (*best effort*, les paquets ne sont pas éliminés sans raison).**

TCP/IP et INTERNET

IP : Internet Protocol

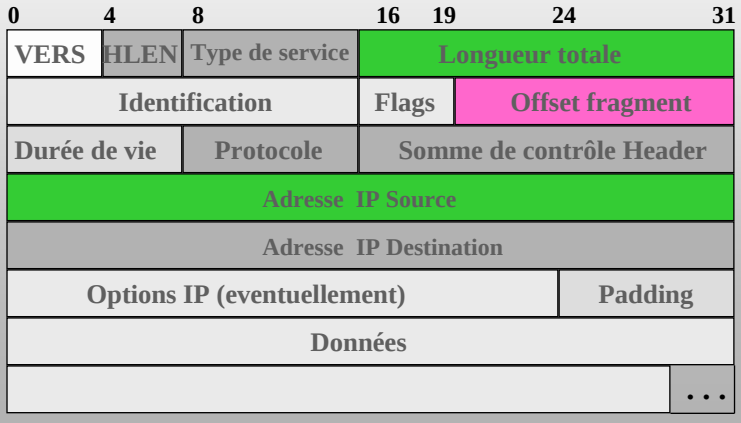
- **Le protocole IP définit :**
  - **l'unité de donnée transférée dans les interconnexions (datagramme),**
  - **la fonction de routage,**
  - **les règles qui mettent en oeuvre la remise de paquets en mode non connecté**

TCP/IP et INTERNET

IP : Internet Protocol

☞ **Le datagramme IP**

L'unité de transfert de base dans un réseau internet est le datagramme qui est constituée d'un en-tête et d'un champ de données:



TCP/IP et INTERNET

IP : Internet Protocol

Signification des champs du datagramme IP :

- **VERS : numéro de version de protocole IP, actuellement version 4,**
- **HLEN : longueur de l'en-tête en mots de 32 bits, généralement égal à 5 (pas d'option),**
- **Longueur totale : longueur totale du datagramme (en-tête + données)**
- **Type de service : indique comment le datagramme doit être géré :**

- **PRECEDENCE (3 bits) : définit la priorité du datagramme; en général ignoré par les machines et passerelles (pb de congestion).**
- **Bits D, T, R : indiquent le type d'acheminement désiré du datagramme, permettant à une passerelle de choisir entre plusieurs routes (si elles existent) : D signifie délai court, T signifie débit élevé et R signifie grande fiabilité.**

TCP/IP et INTERNET

IP : Internet Protocol

- **FRAGMENT OFFSET, FLAGS, IDENTIFICATION : les champs de la fragmentation.**
  - **Sur toute machine ou passerelle mettant en oeuvre TCP/IP une unité maximale de transfert (*Maximum Transfert Unit* ou MTU) définit la taille maximale d'un datagramme véhiculé sur le réseau physique correspondant**
  - **lorsque le datagramme est routé vers un réseau physique dont le MTU est plus petit que le MTU courant, la passerelle fragmente le datagramme en un certain nombre de fragments, véhiculés par autant de trames sur le réseau physique correspondant,**
  - **lorsque le datagramme est routé vers un réseau physique dont le MTU est supérieur au MTU courant, la passerelle route les fragments tels quels (rappel : les datagrammes peuvent emprunter des chemins différents),**
  - **le destinataire final reconstitue le datagramme initial à partir de l'ensemble des fragments reçus; la taille de ces fragments correspond au plus petit MTU emprunté sur le réseau. Si un seul des fragments est perdu, le datagramme initial est considéré comme perdu : la probabilité de perte d'un datagramme augmente avec la fragmentation.**

TCP/IP et INTERNET

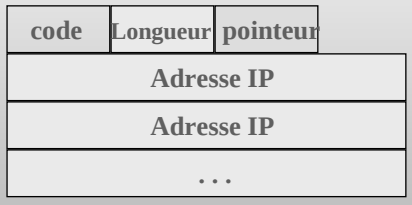
IP : Internet Protocol

- **FRAGMENT OFFSET : indique le déplacement des données contenues dans le fragment par rapport au datagramme initial. C'est un multiple de 8 octets; la taille du fragment est donc également un multiple de 8 octets.**
- **chaque fragment a une structure identique à celle du datagramme initial, seul les champs FLAGS et FRAGMENT OFFSET sont spécifiques.**
- **Durée de vie**
  - **Ce champ indique en secondes, la durée maximale de transit du datagramme sur l'internet. La machine qui émet le datagramme définit sa durée de vie.**
  - **Les passerelles qui traitent le datagramme doivent décrémenter sa durée de vie du nombre de secondes (1 au minimum) que le datagramme a passé pendant son séjour dans la passerelle; lorsque celle-ci expire le datagramme est détruit et un message d'erreur est renvoyé à l'émetteur.**
- **Protocole**

  Ce champ identifie le protocole de niveau supérieur dont le message est véhiculé dans le champ données du datagramme :
  - **6 : TCP,**
  - **17 : UDP,**
  - **1 : ICMP.**

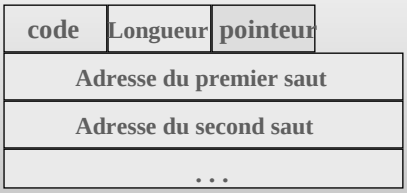
TCP/IP et INTERNET
IP : Internet Protocol

- **<u>Enregistrement de route</u> (classe = 0, option = 7) : permet à la source de créer une liste d'adresse IP vide et de demander à chaque passerelle d'ajouter son adresse dans la liste.**



TCP/IP et INTERNET
IP : Internet Protocol

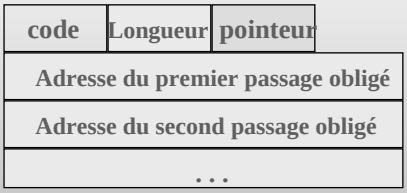
- **<u>Routage strict prédéfini par l'émetteur</u> (classe = 0, option = 9): prédéfinit le routage qui doit être utilisé dans l'interconnexion en indiquant la suite des adresses IP dans l'option :**



  - **Le chemin spécifié ne tolère aucun autre intermédiaire; une erreur est retournée à l'émetteur si une passerelle ne peut appliquer le routage spécifié.**
  - **Les passerelles enregistrent successivement leur adresse à l'emplacement indiqué par le champ *pointeur*.**

TCP/IP et INTERNET
IP : Internet Protocol

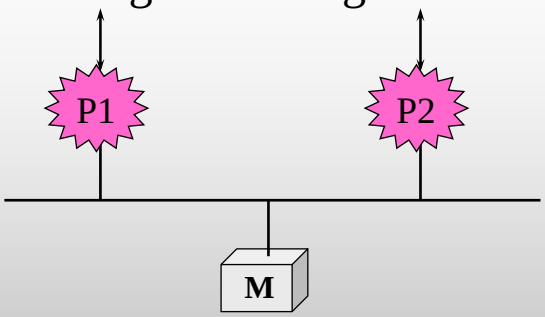
- **<u>Routage lâche prédéfini par l'émetteur</u> (classe = 0, option = 3): Cette option autorise, entre deux passages obligés, le transit par d'autres intermédiaires :**



TCP/IP et INTERNET
Routage des datagrammes

- **Le routage est le processus permettant à un datagramme d'être acheminé vers le destinataire lorsque celui-ci n'est pas sur le même réseau physique que l'émetteur.**
- **Le chemin parcouru est le résultat du processus de routage qui effectue les choix nécessaires afin d'acheminer le datagramme.**
- **Les routeurs forment une structure coopérative de telle manière qu'un datagramme transite de passerelle en passerelle jusqu'à ce que l'une d'entre elles le délivre à son destinataire. Un routeur possède deux ou plusieurs connexions réseaux tandis qu'une machine possède généralement qu'une seule connexion.**
- **Machines et routeurs participent au routage :**
  - **les machines doivent déterminer si le datagramme doit être délivré sur le réseau physique sur lequel elles sont connectées (routage direct) ou bien si le datagramme doit être acheminé vers une passerelle; dans ce cas (routage indirect), elle doit identifier la passerelle appropriée.**
  - **les passerelles effectuent le choix de routage vers d'autres passerelles afin d'acheminer le datagramme vers sa destination finale.**

TCP/IP et INTERNET
Routage des datagrammes



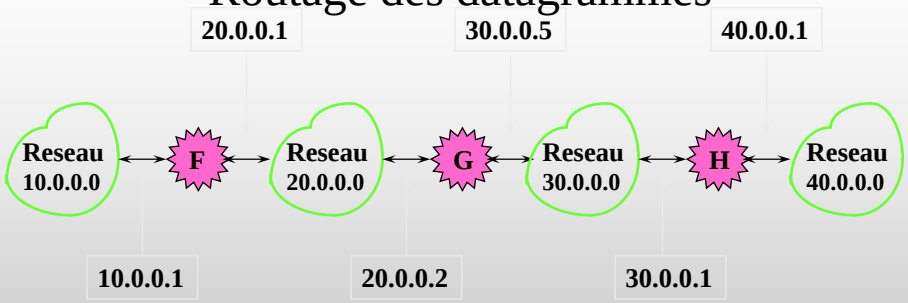
M est mono-domiciliée et doit acheminer les datagrammes vers une des passerelles P1 ou P2; elle effectue donc le premier routage. Dans cette situation, aucune solution n'offre un meilleur choix.

Le routage indirect repose sur une table de routage IP, présente sur toute machine et passerelle, indiquant la manière d'atteindre un ensemble de destinations.

TCP/IP et INTERNET
Routage des datagrammes

- **Les tables de routage IP, pour des raisons évidentes d'encombrement, renseignent seulement les adresses réseaux et non pas les adresses machines.**
- **Typiquement, une table de routage contient des couples (R, P) où R est l'adresse IP d'un réseau destination et P est l'adresse IP de la passerelle correspondant au prochain saut dans le cheminement vers le réseau destinataire.**
- **La passerelle ne connaît pas le chemin complet pour atteindre la destination.**
- **Pour une table de routage contenant des couples (R, P) et appartenant à la machine M, P et M sont connectés sur le même réseau physique dont l'adresse de niveau réseau (partie Netid de l'adresse IP) est R.**

TCP/IP et INTERNET
Routage des datagrammes



Pour atteindre les machines du réseau	10.0.0.0	20.0.0.0	30.0.0.0	40.0.0.0
Router vers	20.0.0.1	direct	direct	30.0.0.1

Table de routage de G

TCP/IP et INTERNET
Routage des datagrammes

Route_Datagramme_IP(datagramme, table_de_routage)

- Extraire l'adresse IP destination, ID, du datagramme,
- Calculer l'adresse du réseau destination, IN.
- Si IN correspondant à une adresse de réseau directement accessible,
  envoyer le datagramme vers sa destination, sur ce réseau.
- sinon si dans la tablede routage, il existe une route vers ID
  router le datagramme selon les informations contenues dans la table de routage.
- sinon si IN apparaît dans la table de routage,
  router le datagramme selon les informations contenues dans la table de routage.
- sinon s'il existe une route par défaut
  router le datagramme vers la passerelle par défaut.
- sinon déclarer une erreur de routage.

TCP/IP et INTERNET
Routage des datagrammes

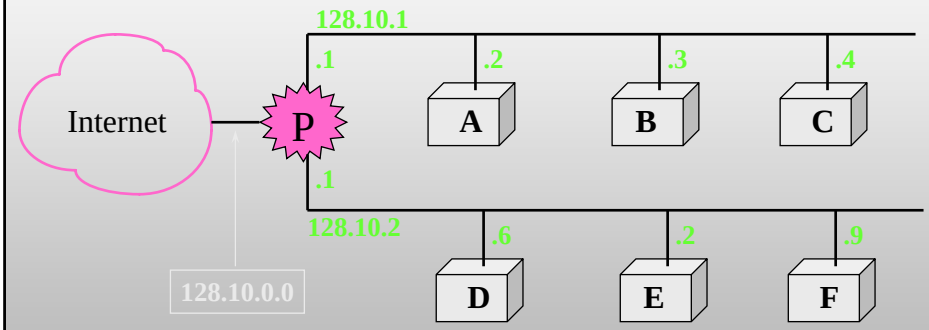
- **Après exécution de l'algorithme de routage, IP transmet le datagramme ainsi que l'adresse IP determinée, à l'interface réseau vers lequel le datagramme doit être acheminé.**
- **L'interface physique détermine alors l'adresse physique associée à l'adresse IP et achemine le datagramme sans l'avoir modifié (l'adresse IP du prochain saut n'est sauvegardée nulle part).**
- **Si le datagramme est acheminé vers une autre passerelle, il est à nouveau géré de la même manière, et ainsi de suite jusqu'à sa destination finale.**

TCP/IP et INTERNET : sous-réseaux

- **Le sous-adressage est une extension du plan d'adressage initial**
- **Devant la croissance du nombre de réseaux de l'Internet, il a été introduit afin de limiter la consommation d'adresses IP qui permet également de diminuer :**
  - **la gestion administrative des adresses IP,**
  - **la taille des tables de routage des passerelles,**
  - **la taille des informations de routage,**
  - **le traitement effectué au niveau des passerelles.**
- **Principes**
  - **A l'intérieur d'une entité associée à une adresse IP de classe A, B ou C, plusieurs réseaux physiques partagent cette adresse IP.**
  - **On dit alors que ces réseaux physiques sont des sous-réseaux (*subnet*) du réseau d'adresse IP.**

TCP/IP et INTERNET : sous-réseaux

Les sous-réseaux 128.10.1.0 et 128.10.2.0 sont notés seulement avec le NetId, les machines seulement avec le Hostid ; exemple IP(F) = 128.10.2.9



Un site avec deux réseaux physiques utilisant le sous-adressage de manière à ce que ses deux sous-réseaux soient couverts par une seule adresse IP de classe B.
La passerelle P accepte tout le trafic destiné au réseau 128.10.0.0 et sélectionne le sous-réseau en fonction du troisième octet de l'adresse destination.

TCP/IP et INTERNET : sous-réseaux

- **Le site utilise une seule adresse pour les deux réseaux physiques.**
- **A l'exception de P, toute passerelle de l'internet route comme s'il n'existait qu'un seul réseau.**
- **La passerelle doit router vers l'un ou l'autre des sous-réseaux ; le découpage du site en sous-réseaux a été effectué sur la base du troisième octet de l'adresse :**
  - **les adresses des machines du premier sous-réseau sont de la forme 128.10.1.X,**
  - **les adresses des machines du second sous-réseau sont de la forme 128.10.2.X.**
- **Pour sélectionner l'un ou l'autre des sous-réseaux, P examine le troisième octet de l'adresse destination : si la valeur est 1, le datagramme est routé vers réseau 128.10.1.0, si la valeur est 2, il est routé vers le réseau 128.10.2.0.**

TCP/IP et INTERNET : sous-réseaux

- **Conceptuellement, la partie locale dans le plan d'adressage initial est subdivisée en "partie réseau physique" + "identification de machine (hostid) sur ce sous-réseau" :**

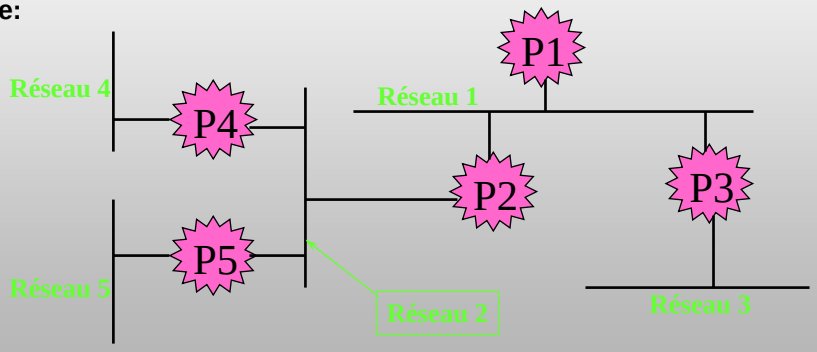
Partie Internet	Partie locale	
Partie Internet	Réseau physique	Id Machine

- **«Partie Internet» correspond au NetId (plan d'adressage initial)**
- **«Partie locale» correspond au hostid (plan d'adressage initial)**
- **les champs «Réseau physique» et «identificateur Machine» sont de taille variable; la longueur des 2 champs étant toujours égale à la longueur de la «Partie locale».**

TCP/IP et INTERNET : sous-réseaux

Structure du sous-adressage

- **Structuration souple : chaque site peut définir lui-même les longueurs des champs réseau physique et identificateur de machine.**
- **Flexibilité indispensable pour adapter la configuration réseau d'un site:**



Ce site a cinq réseaux physiques organisés en trois niveau : le découpage rudimentaire en réseau physique et adresse machine peut ne pas être optimal.

TCP/IP et INTERNET : sous-réseaux

- **Le choix du découpage dépend des perspectives d'évolution du site:**
  - **Exemple Classe B : 8 bits pour les parties réseau et machine donnent un potentiel de 256 sous-réseaux et 254 machines par sous-réseau, tandis que 3 bits pour la partie réseau et 13 bits pour le champ machine permettent 8 réseaux de 8190 machines chacun.**
  - **Exemple Classe C : 4 bits pour la partie réseau et 4 bits pour le champ machine permettent 16 réseaux de 14 machines chacun.**
- **Lorsque le sous-adressage est ainsi défini, toutes les machines du réseau doivent s'y conformer sous peine de dysfonctionnement du routage ==> configuration rigoureuse.**

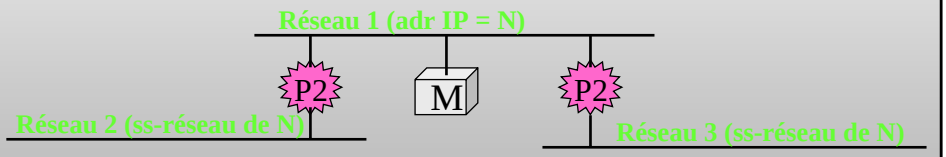
TCP/IP et INTERNET : sous-réseaux

- **Utilisation de masques**
- **Le sous-adressage ==> masque de 32 bits associé au sous-réseau.**
- **Bits du masque de sous-réseau (*subnet mask*) :**
  - **positionnés à 1 : partie réseau,**
  - **positionnés à 0 : partie machine**
- **11111111 11111111 11111111 00000000**
  ==> 3 octets pour le champ réseau, 1 octet pour le champ machine
- **Les bits du masque identifiant sous-réseau et machine peuvent ne pas être contigus : 11111111 11111111 00011000 01000000**
- **Les notations suivantes sont utilisées :**
  - **décimale pointée; exemple : 255.255.255.0**
  - **triplet : { <ident. réseau>, <ident. sous-réseau> <ident. machine> } ; cette notation renseigne les valeurs mais pas les champs de bits;**
    exemple { -1, -1, 0 } , { 128.10, 27, -1 }.
  - **adresse réseau/masque : 193.49.60.0/27 (27=# bits contigüs du masque)**

TCP/IP et INTERNET : sous-réseaux

Routage avec sous-réseaux

- **Le routage IP initial a été étendu à l'adressage en sous-réseaux;**
- **l'algorithme de routage obtenu doit être présent dans les machines ayant une adresse de sous-réseau, mais également dans les autres machines et passerelles du site qui doivent acheminer les datagrammes vers ces sous-réseaux.**



M doit utiliser le routage de sous-réseaux pour décider si elle route vers les passerelles P1 ou P2 bien qu'elle même soit connectée à un réseau (Réseau 1) n'ayant pas de sous-adressage

TCP/IP et INTERNET : sous-réseaux

Le routage unifié : Une entrée dans la table de routage =
(masque de sous-réseau, adresse sous-réseau, adresse de la passerelle)
Algorithme de routage unifié :

- **Route_IP_Datagram(datagram, routing_table)**
- **Extraire l'adresse ID de destination du datagramme,**
- **Calculer l'adresse IN du réseau destination,**
- **Si IN correspond à une adresse réseau directement accessible**
  envoyer le datagramme sur le réseau physique correspondant,
- **Sinon**
  - **Pour chaque entrée dans la table de routage,**
    - **N = (ID & masque de sous-réseau de l'entrée)**
    - **Si N est égal au champ adresse réseau de l'entrée**
      router le datagramme vers la passerelle correspondante,
  - **Fin_Pour**
- **Si aucune entrée ne correspond, déclarer une erreur de routage.**

TCP/IP et INTERNET : sous-réseaux

- **Diffusion sur les sous-réseaux**
- **Elle est plus complexe que dans le plan d'adressage initial.**
- **Dans le plan d'adressage Internet initial, Hostid = 11..1, ==> diffusion vers toutes les machines du réseau.**
- **D'un point de vue extérieur à un site doté de sous-réseaux, la diffusion n'a de sens que si la passerelle qui connaît les sous-réseaux propage la diffusion à tous ses réseaux physiques : { réseau, -1, -1 }.**
- **Depuis un ensemble de sous-réseau, il est possible d'émettre une diffusion sur un sous-réseau particulier : { réseau, sous-réseau, -1 }.**

TCP/IP et INTERNET : Domain Name Server Besoins

- **L'Internet est constitué de réseaux (dizaines de milliers)**
- **Les réseaux sont constitués de sous-réseaux**
- **Les sous-réseaux sont constitués de machines,**
- **La technologie de base (TCP/IP) permet l'accès aux machines par leur adresse IP,**
- **Il est pratiquement devenu impossible aux humains de connaître les adresses (IP) des machines auxquelles ils veulent accéder.**
- **Le système DNS permet d'identifier une machine par un (des) nom(s) représentatif(s) de la machine et du (des) réseau(x) sur le(les)quel(s) elle se trouve ; exemple :**

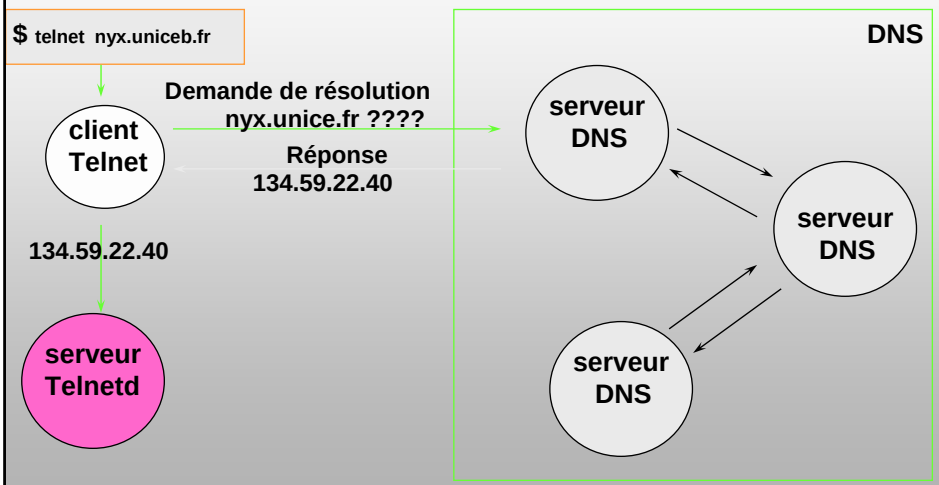
  nyx.unice.fr identifie la machine nyx sur le réseau unice.fr
- **Le système est mis en œuvre par une base de données distribuée au niveau mondial**
- **Les noms sont gérés par un organisme mondial : l'interNIC et les organismes délégués : RIPE, NIC France, NIC Angleterre, etc.**

TCP/IP et INTERNET : Domain Name Server Principes

- **basé sur le modèle client / serveur**
- **le logiciel client interroge un serveur de nom; typiquement :**
  - **l'utilisateur associe un nom de domaine à une application ; exemple :**

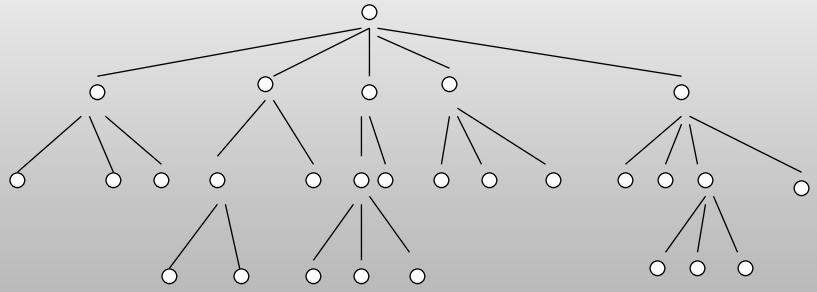
    telnet nyx.unice.fr
  - **l'application cliente requiert la traduction du nom de domaine auprès d'un serveur de nom (DNS) : cette opération s'appelle la résolution de nom**
  - **le serveur de nom interroge d'autres serveurs de nom jusqu'à ce que l'association nom de domaine / adresse IP soit trouvée**
- **le serveur de nom retourne l'adresse IP au logiciel client : 134.59.22.40**
- **le logiciel client contacte le serveur (telnetd) comme si l'utilisateur avait spécifié une adresse IP : telnet 134.59.22.40**

TCP/IP et INTERNET : Domain Name Server Principes



TCP/IP et INTERNET : Domain Name Server
Espace Nom de domaine

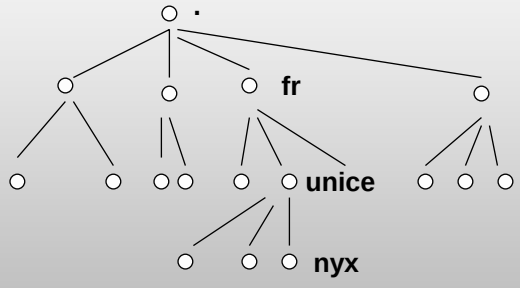
- **Chaque unité de donnée dans la base DNS est indexée par un nom**
- **Les noms constituent un chemin dans un arbre inversé appelé l'espace Nom de domaine**
- **Organisation similaire à un système de gestion de fichiers**



• Chaque noeud est identifié par un nom
• Racine appelée root, identifiée par «.»
• 127 niveaux au maximum

TCP/IP et INTERNET : Domain Name Server
Nom de domaine

Un nom de domaine est est la séquence de labels depuis le noeud de l'arbre correspondant jusqu'à la racine

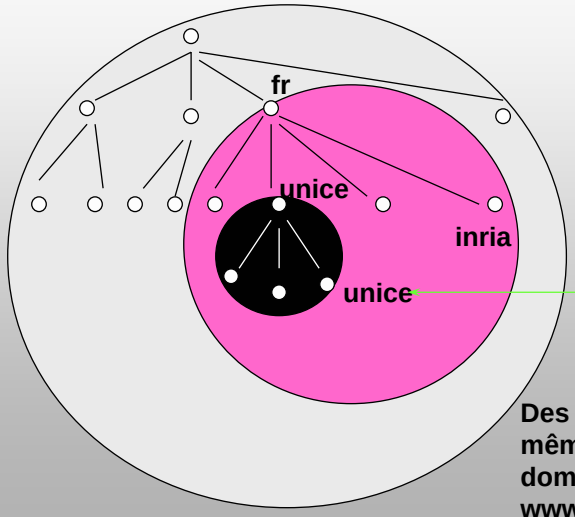


nyx.unice.fr

Deux noeuds fils ne peuvent avoir le même nom ==> unicité d'un nom de domaine au niveau mondial

TCP/IP et INTERNET : Domain Name Server
Domaine

Un domaine est un sous-arbre de l'espace nom de domaine



Domaine complet

Domaine fr

Domaine unice

noeud nyx.unice.fr

Des noeuds peuvent avoir les mêmes noms dans des domaines différents : www.unice.fr et www.inria.fr

TCP/IP et INTERNET : Domain Name Server
Résumé

- **Un domaine est un sous-arbre de l'espace Nom de domaine**
- **Un domaine est constitué de noms de domaine et d' autres domaines**
- **Un domaine intérieur à un autre domaine est appelé un sous domaine**
- **Exemple : le domaine fr comprend le noeud fr et tous les noeuds contenus dans tous les sous-domaines de fr**

- **Un nom de domaine est un index dans la base DNS; exemple :**
  - **m1.centralweb.fr pointe vers une adresse IP**
  - **centralweb.fr pointe vers des informations de routage de mail et éventuellement des informations de sous-domaines**
  - **fr pointe vers des informations structurelles de sous-domaines**

- **Les machines sont reliées entre elles dans un même domaine logiquement et non par adressage. Exemple : 10 machines d'un même domaine appartiennent à 10 réseaux différents et recouvrent 6 pays différents.**

TCP/IP et INTERNET : Domain Name Server
Domaines racines

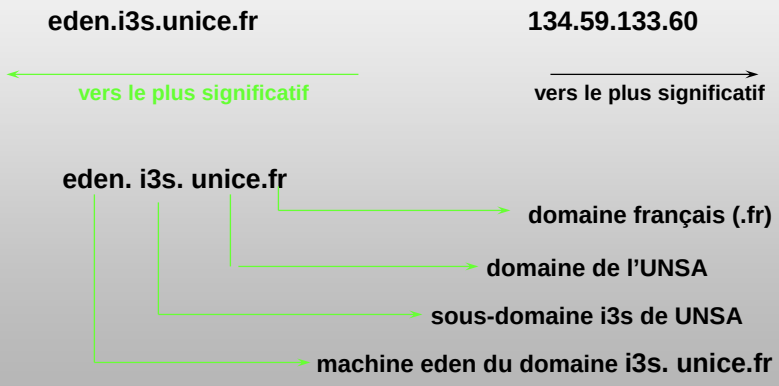
- **Le système DNS impose peu de règles de nommage :**
  - **noms < 63 caractères**
  - **majuscules et minuscules non significatives**
  - **pas de signification imposée pour les labels**
- **Le premier niveau de l'espace DNS fait exception à la règle :**
  - **7 domaines racines prédéfinis :**
    - **com : organisations commerciales ; ibm.com**
    - **edu : organisations concernant l'éducation ; mit.edu**
    - **gov : organisations gouvernementales ; nsf.gov**
    - **mil : organisations militaires ; army.mil**
    - **net : organisations réseau Internet ; worldnet.net**
    - **org : organisations non commerciales ; eff.org**
    - **int : organisations internationales ; nato.int**
  - **arpa : domaine réservé à la résolution de nom inversée**
  - **organisations nationales : fr, uk, de, it, us, au, ca, se, etc.**

TCP/IP et INTERNET : Domain Name Server
Domaines racines

- **Nouveaux domaines racine en cours de normalisation:**
  - **firm, store, web, arts, rec, info, nom**
- **Certaines organisations nationales peuvent être gérées administrativement par un consortium : RIPE**
- **Les divisions en sous-domaines existent dans certains pays et pas dans d'autres :**
  - **edu.au, com.au, etc.**
  - **co.uk, ac.uk, etc.**
  - **ca.ab, ca.on, ca.gb**
  - **pas de division du .fr**

TCP/IP et INTERNET : Domain Name Server Interprétation

- **A l'inverse de l'adressage IP la partie la plus significative si situe à gauche de la syntaxe :**

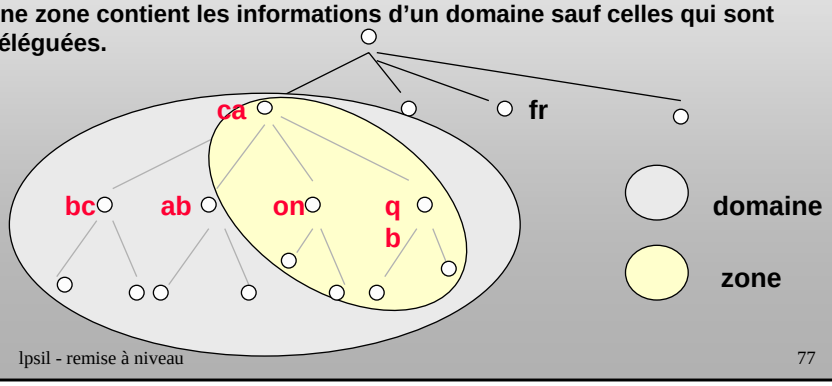


TCP/IP et INTERNET : Domain Name Server Délégation

- **Le système DNS est entièrement distribué au niveau planétaire; Le mécanisme sous-jacent est la délégation de domaine**
- **A tout domaine est associé une responsabilité administrative**
- **Une organisation responsable d'un domaine peut**
  - **découper le domaine en sous-domaines**
  - **déléguer les sous-domaines à d'autres organisations :**
    - **qui deviennent à leur tour responsables du (des) sous-domaine(s) qui leurs sont délégué(s)**
    - **peuvent, à leur tour, déléguer des sous-domaines des sous-domaines qu'elles gèrent**
- **Le domaine parent contient alors seulement un pointeur vers le sous-domaine délégué; exemple :**
  - **unice.fr est délégué à l'UNSA**
  - **L'UNSA gère donc les données propres à ce domaine.**
  - **unice.fr (en théorie seulement) pourrait être géré par l'organisation responsable du domaine .fr (NIC France) qui gèrerait alors les données de unice.fr**

TCP/IP et INTERNET : Domain Name Server
Serveur de noms

- Les logiciels qui gèrent les données de l'espace nom de domaine sont appelés des serveurs de nom (*name servers*)
- Les serveurs de nom enregistrent les données propres à une partie de l'espace nom de domaine dans une zone.
- Le serveur de nom à autorité administrative sur cette zone.
- Un serveur de nom peut avoir autorité sur plusieurs zone.
- Une zone contient les informations d'un domaine sauf celles qui sont déléguées.



TCP/IP et INTERNET : Domain Name Server
Serveur de noms

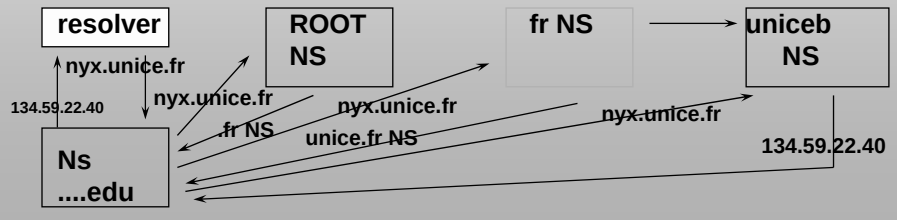
- Serveur de nom primaire : maintient la base de données de la zone dont il a l'autorité administrative
- Serveur de nom secondaire : obtient les données de la zone via un autre serveur de nom qui a également l'autorité administrative
  - interroge périodiquement le serveur de nom primaire et met à jour les données
- Il y a un serveur primaire et généralement plusieurs secondaires
- La redondance permet la défaillance éventuelle du primaire et du (des) secondaire(s)
- Un serveur de nom peut être primaire pour une (des) zone(s) et secondaire pour d'autre(s).

TCP/IP et INTERNET : Domain Name Server Resolvers

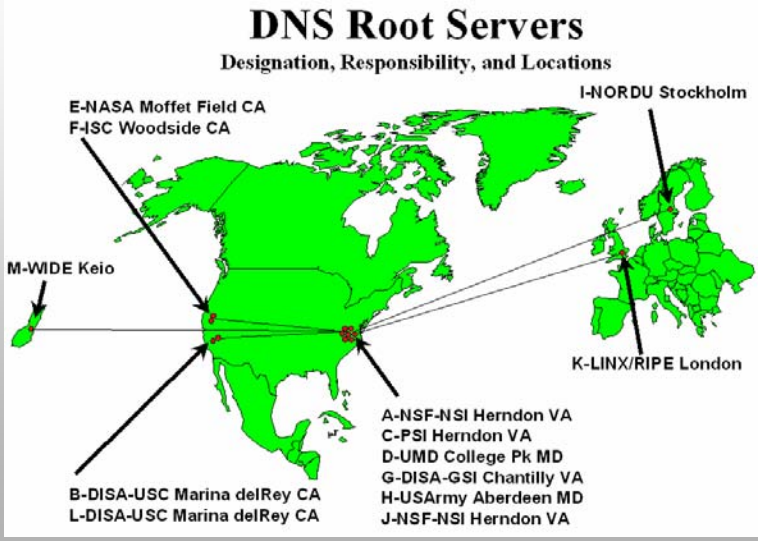
- **Les «resolvers» sont les processus clients qui contactent les serveurs de nom**
- **Fonctionnement :**
  - **contacte un name serveur (dont l' (les) adresse(s) est (sont) configurées sur la machine exécutant ce resolver)**
  - **interprète les réponses**
  - **retourne l'information au logiciel appelant**
  - **gestion de cache (dépend de la mise en œuvre)**
- **Le serveur serveur de nom interroge également d'autres serveurs de nom, lorsqu'il n'a pas autorité sur la zone requise (fonctionnement itératif ou récursif)**
- **Si le serveur de nom est en dehors du domaine requis, il peut être amené à contacter un serveur racine (ne pas confondre avec un domaine racine)**

TCP/IP et INTERNET : Domain Name Server Serveurs racines

- **Les serveurs racine connaissent les serveurs de nom ayant autorité sur tous les domaines racine**
- **Les serveurs racine connaissent au moins les serveurs de noms pouvant résoudre le premier niveau (.com, .edu, .fr, etc.)**
- **Pierre angulaire du système DNS : si les serveurs racine sont inoperationnels ==> plus de communication sur l'Internet**
  - **==> multiplicité des serveurs racines**
  - **actuellement jusqu'à 14 éparpillés sur la planète**
  - **chaque serveur racine reçoit environ 100000 requêtes / heure**
- **Exemple de résolution : nyx.unice.fr à partir de ....edu**



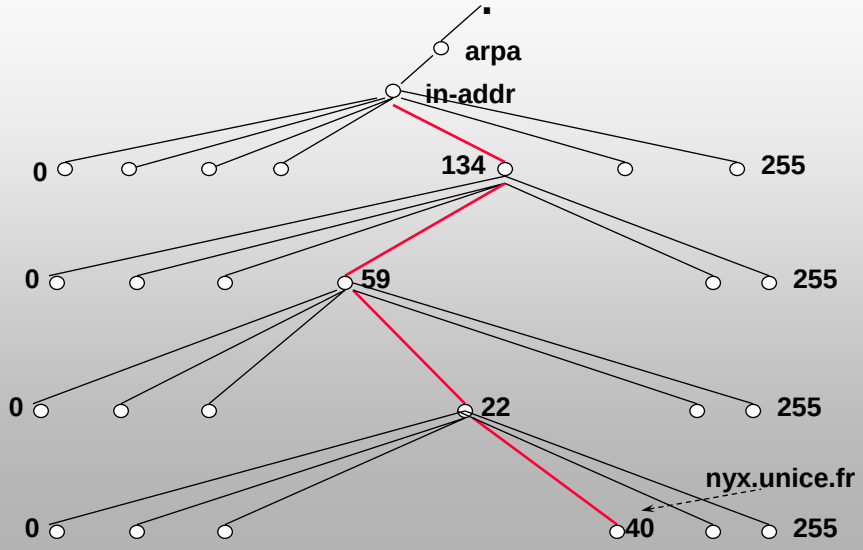
TCP/IP et INTERNET : Domain Name Server
Serveurs racines



TCP/IP et INTERNET : Domain Name Server
Résolution inverse

- **Consiste a obtenir le nom de domaine à partir de l'adresse IP**
  - **pour faciliter la compréhension des humains**
  - **pour des raisons de sécurité**
- **Plus délicate que nom -> IP car le système DNS est organisé pour la résolution de nom ==> recherche exhaustive ???**
- **Solution : utiliser les adresses comme des noms :**
  - **le domaine in-addr.arpa**
  - **les noms des noeuds correspondent aux octets de l'adresse IP en ordre inverse**
  - **le domaine in-addr.arpa a 256 sous-domaines,**
  - **chacun de ces sous-domaines a 256 sous-domaines,**
  - **chacun de ces sous-domaines a, à son tour, 256 sous-domaines,**
  - **le 4ème niveau correspond à un NS connaissant le nom de domaine associé à cette adresse IP**

TCP/IP et INTERNET : Domain Name Server
Résolution inverse



TCP/IP et INTERNET : Domain Name Server
Résolution inverse

- **le nom de domaine associé à la résolution inverse est noté selon l'adresse IP inversée :**
  - **car la résolution d'un nom de domaine se fait de droite à gauche**
  - **exemple : 210.37.148.193.in-addr.arpa**
  - **résolution :**
    - **in-addr.arpa -> A.ROOT-SERVER.NET**
    - **134.in-addr.arpa -> NS.RIPE.NET**
    - **59. 134.in-addr.arpa -> NS.UNICE.FR**
  - **Organismes gérant les classes**
    - **Classe A et B -> internic US.**
    - **Classe C**
      - **192 : internic**
      - **193, 194, 195 RIPE avec délégations nationales**

TCP/IP et INTERNET : Domain Name Server Enregistrement

- **Les données d'un serveur DNS sont enregistrées dans une base identifiée par les noms de domaine correspondants; exemple :**
  - **db. unice.fr, unice.fr.dns**
  - **db.134.59, 134.59.dns**
  - **db.127.0.0, 127.0.0.dns**
  - **db.cache, cache.dns**
- **Types d'enregistrements**
  - **SOA: décrit l'autorité administrative,**
  - **NS : liste de serveurs de nom pour ce domaine**
  - **A : correspondance nom -> adresse**
  - **PTR : correspondance adresse -> nom**
  - **CNAME : alias**
  - **TXT : texte**
  - **HINFO : description machine**

TCP/IP et INTERNET : Domain Name Server Utilisation

- **Utiliser un serveur de nom**
  - **machine elle-même serveur de nom : 127.0.0.1**
  - **machine non serveur de nom : spécfier un ou plusieurs serveur de nom : adresses IP obligatoirement. éventuellement son domaine.**
  - **sous UNIX : fichier /etc/resolv**
  - **sous NT, W95 : administration TCP/IP**
- **Administrer un serveur de nom**
  - **plateformes UNIX, NT**
  - **mémoire importante : mini 16/32 MB pour le service.**
  - **impératif : ne pas swapper**
  - **opérationnelle 24/24**
  - **laisser passer le port 53 sur UDP et TCP**
- **Debugging : Nslookup**

TCP/IP et INTERNET
UDP : User Datagram Protocol

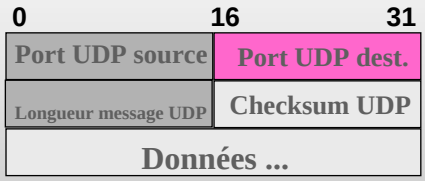
- **UDP : protocole de transport sans connexion de service applicatif :**
  - **émission de messages applicatifs : sans établissement de connexion au préalable**
  - **l'arrivée des messages ainsi que l'ordonnancement ne sont pas garantis.**
- **Identification du service : les ports**
  - **les adresses IP désignent les machines entre lesquelles les communications sont établies. Lorsqu'un processus désire entrer en communication avec un autre processus, il doit adresser le processus s'exécutant cette machine.**
  - **L'adressage de ce processus est effectué selon un concept abstrait indépendant du système d'exploitation des machines car :**
    - **les processus sont créés et détruits dynamiquement sur les machines,**
    - **il faut pouvoir remplacer un processus par un autre (exemple reboot) sans que l'application distante ne s'en aperçoive,**
    - **il faut identifier les destinations selon les services offerts, sans connaître les processus qui les mettent en oeuvre,**
    - **un processus doit pouvoir assurer plusieurs services.**

TCP/IP et INTERNET
UDP : ports

- **Ces destinations abstraites permettant d'adresser un service applicatif s'appellent des ports de protocole.**
- **L'émission d'un message se fait sur la base d'un port source et un port destinataire.**
- **Les processus disposent d'une interface système leur permettant de spécifier un port ou d'y accéder (socket, TLI, ...).**
- **Les accès aux ports sont généralement synchrones, les opérations sur les ports sont tamponnés (files d'attente).**

TCP/IP et INTERNET
UDP : format des messages

Les messages UDP sont également appelés des datagrammes UDP. Ils contiennent deux parties : un en-tête UDP et les données UDP.



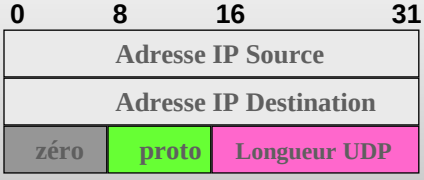
Format des messages UDP

Les ports source et destination contiennent les numéros de port utilisés par UDP pour démultiplexer les datagrammes destinés aux processus en attente de les recevoir. Le port source est facultatif (égal à zéro si non utilisé).

La longueur du message est exprimée en octets (8 au minimum) (en-tête + données), le champ de contrôle est optionnel (0 si non utilisé).

TCP/IP et INTERNET
UDP : pseudo en-tête

- **Lorsqu'il est utilisé, le champ de contrôle couvre plus d'informations que celles contenue dans le datagramme UDP; En effet, le checksum est calculé avec un pseudo-en-tête non transmis dans le datagramme:**



Format du pseudo en-tête

Le champ PROTO indique l'identificateur de protocole pour IP (17= UDP)

Le champ LONGUEUR UPD spécifie la longueur du datagramme UPD sans le pseudo-en-tête.

TCP/IP et INTERNET
UDP : Multiplexage

- **UDP multiplexe et démultiplexe les datagrammes en sélectionnant les numéros de ports :**
  - **une application obtient un numéro de port de la machine locale; dès lors que l'application émet un message via ce port, le champ PORT SOURCE du datagramme UDP contient ce numéro de port,**
  - **une application connaît (ou obtient) un numéro de port distant afin de communiquer avec le service désiré.**
- **Lorsque UDP reçoit un datagramme, il vérifie que celui-ci est un des ports actuellement actifs (associé à une application) et le délivre à l'application responsable (mise en queue)**
- **si ce n'est pas le cas, il émet un message ICMP *port unreachable*, et détruit le datagramme.**

TCP/IP et INTERNET
UDP : les ports standards

- **Certains ports sont réservés (*well-kown port assignements*) :**

No port	Mot-clé	Description
7	ECHO	Echo
11	USERS	Active Users
13	DAYTIME	Daytime
37	TIME	Time
42	NAMESERVER	Host Name Server
53	DOMAIN	Domain Name Server
67	BOOTPS	Boot protocol server
68	BOOTPC	Boot protocol client
69	TFTP	Trivial File transfert protocol
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management prot.

- **D'autres numéros de port (non réservés) peuvent être assignés dynamiquement aux applications.**

TCP/IP et INTERNET
TCP : Transmission Control Protocol

- **transport fiable de la technologie TCP/IP.**
  - **fiabilité = illusion assurée par le service**
  - **transferts tamponés : découpage en segments**
  - **connexions bidirectionnelles et simultanées**
- **service en mode connecté**
- **garantie de non perte de messages ainsi que de l'ordonnancement**

TCP/IP et INTERNET
TCP : La connexion

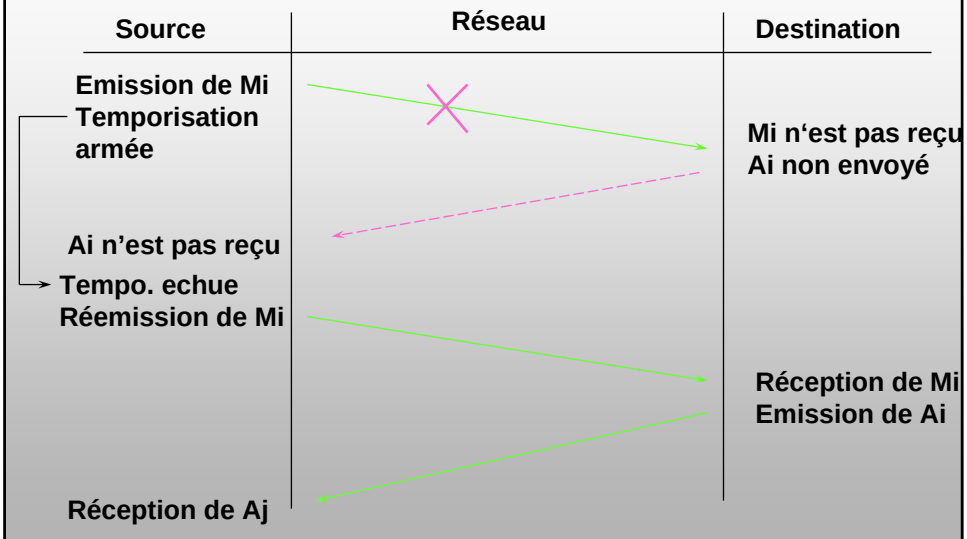
- **une connexion de type circuit virtuel est établie avant que les données ne soient échangées : appel + négociation + transferts**

- **Une connexion = une paire d'extrémités de connexion**
- **Une extrémité de connexion = couple (adresse IP, port)**
- **Exemple de connexion : ((124.32.12.1, 1034), (19.24.67.2, 21))**
- **Une extrémité de connexion peut être partagée par plusieurs autres extrémités de connexions (multi-instanciation)**
- **La mise en oeuvre de la connexion se fait en deux étapes :**
  - **une application (extrémité) effectue une ouverture passive en indiquant qu'elle accepte une connexion entrante,**
  - **une autre application (extrémité) effectue une ouverture active pour demander l'établissement de la connexion.**

TCP/IP et INTERNET
TCP : Segmentation

- **Segmentation, contrôle de flux**
  - **Les données transmises à TCP constituent un flot d'octets de longueur variable.**
  - **TCP divise ce flot de données en segments en utilisant un mécanisme de fenêtrage.**
  - **Un segment est émis dans un datagramme IP.**
- **Acquittement de messages**
  - **Contrairement à UDP, TCP garantit l'arrivée des messages, c'est à dire qu'en cas de perte, les deux extrémités sont prévenues.**
  - **Ce concept repose sur les techniques d'acquittement de message : lorsqu'une source S émet un message Mi vers une destination D, S attend un acquittement Ai de D avant d'émettre le message suivant Mi+1.**
  - **Si l'acquittement Ai ne parvient pas à S, S considère au bout d'un certain temps que le message est perdu et reémet Mi :**

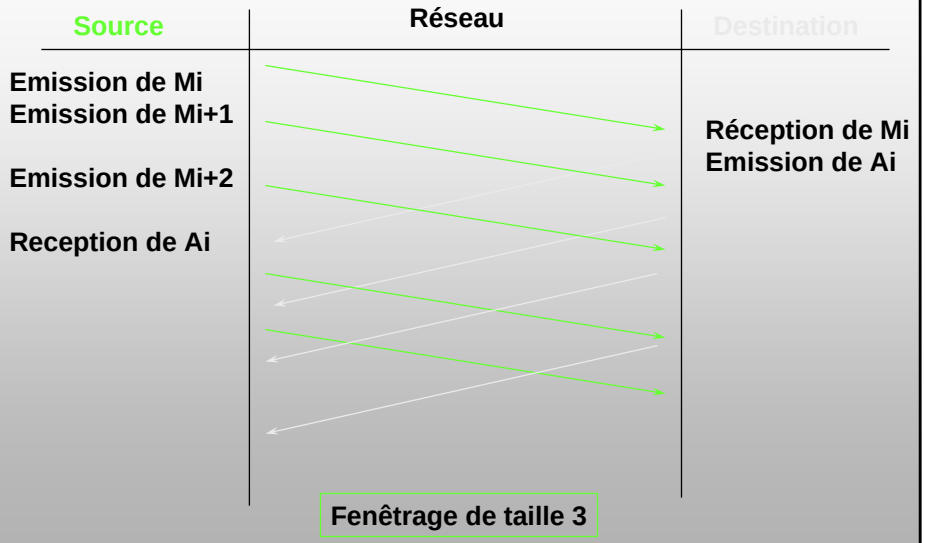
TCP/IP et INTERNET
TCP : Acquittements



TCP/IP et INTERNET
TCP : le fenêtrage

- **La technique acquittement simple pénalise les performances puisqu'il faut attendre un acquittement avant d'émettre un nouveau message. Le fenêtrage améliore le rendement des réseaux.**
- **La technique du fenêtrage : une fenêtre de taille T, permet l'émission d'au plus T messages "*non acquittés*" avant de ne plus pouvoir émettre :**

TCP : le Fenêtrage

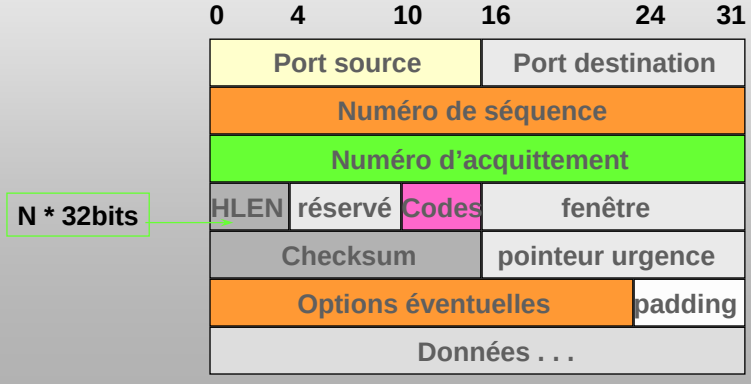


TCP/IP et INTERNET T
CP : Technique de fenêtrage

- **fenêtrage glissante permettant d'optimiser la bande passante**
- **permet également au destinataire de faire diminuer le débit de l'émetteur donc de gérer le contrôle de flux.**
- **Le mécanisme de fenêtrage mis en oeuvre dans TCP opère au niveau de l'octet et non pas au niveau du segment; il repose sur :**
  - **la numérotation séquentielle des octets de données,**
  - **la gestion de trois pointeurs par fenêtrage :**

TCP/IP et INTERNET
TCP : Segments

- **Segment : unité de transfert du protocole TCP.**
  - **échangés pour établir les connexions,**
  - **transférer les données,**
  - **émettre des acquittements,**
  - **fermer les connexions;**



TCP/IP et INTERNET
TCP : la congestion

Gestion de la congestion

- **TCP gère le contrôle de flux de bout en bout mais également les problèmes de congestion liés à l'interconnexion.**
- **La congestion correspond à la saturation de noeud(s) dans le réseau provoquant des délais d'acheminement de datagrammes jusqu'a leur pertes éventuelles.**
- **Les extrémité ignorent tout de la congestion sauf les délais. Habituellement, les protocoles retransmettent les segments ce qui agrave encore le phénomène.**
- **Dans la technologie TCP/IP, les passerelles (niveau IP) utilisent la réduction du débit de la source mais TCP participe également à la gestion de la congestion en diminuant le débit lorsque les délais s'allongent :**

TCP/IP et INTERNET
TCP : la congestion

- **TCP maintient une fenêtre virtuelle de congestion**
- **TCP applique la fenêtre d'émission suivante:**
  - **fenêtre_autorisée = min (fenêtre_récepteur, fenêtre_congestion).**
- **Dans une situation de non congestion:**
  - **fenêtre_récepteur = fenêtre_congestion.**
- **En cas de congestion, TCP applique une diminution dichotomique :**
  - **à chaque segment perdu, la fenêtre de congestion est diminuée par 2 (minimum 1 segment)**
  - **la temporisation de retransmission est augmentée exponentiellement.**

TCP/IP et INTERNET
TCP retransmissions

- **Si la congestion disparaît, TCP définit une fenêtre de congestion égale à 1 segment et l'incrémente de 1 chaque fois qu'un acquittement est reçu; ce mécanisme permet un démarrage lent et progressif :**

  Fenêtre_congestion = 1,
  émission du 1er segment,
  attente acquittement,
  réception acquittement,

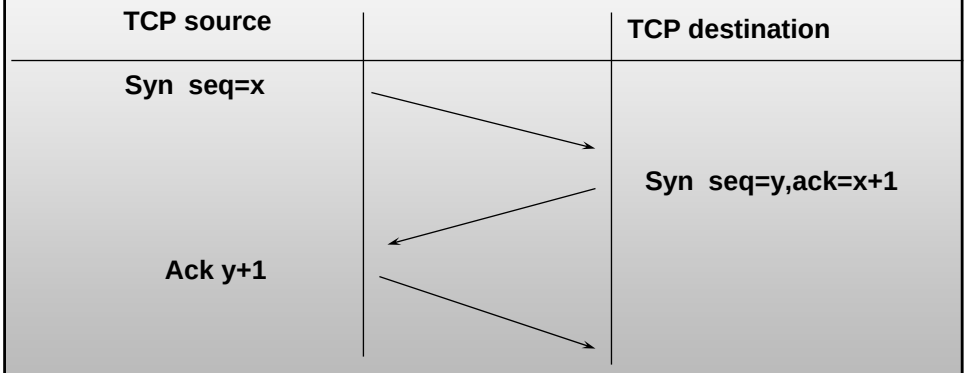
  Fenêtre_congestion = 2,
  émission des 2 segments,
  attente des acquittements,
  réception des 2 acquittements,

  Fenêtre_congestion = 4,
  émission des 4 segments, …

Log2 N itérations pour envoyer N segments. Lorsque la fenêtre atteint une fois et demie sa taille initiale, l'incrément est limité à 1 pour tous les segments acquittés de la fenêtre.

TCP/IP et INTERNET
TCP : connexion

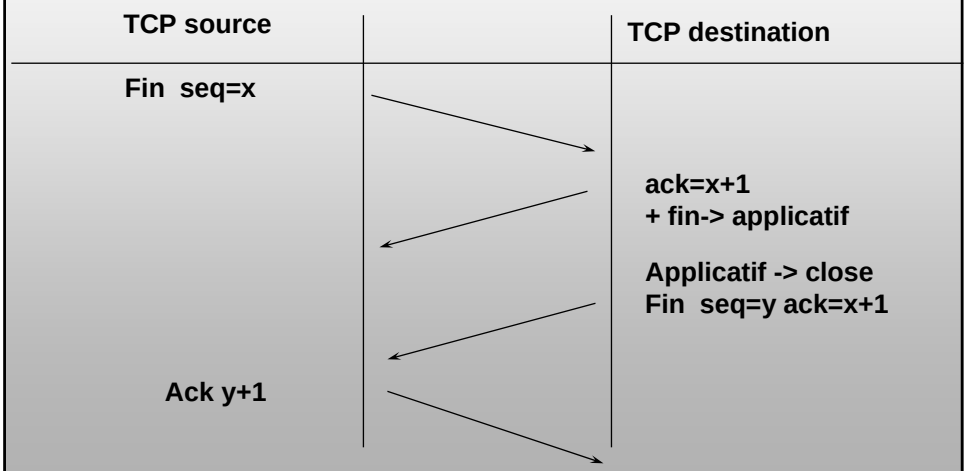
Une connexion TCP est établie en trois temps de manière à assurer la synchronisation nécessaire entre les extrémités :



Ce schéma fonctionne lorsque les deux extrémités effectuent une demande d'établissement simultanément. TCP ignore toute demande de connexion, si cette connexion est déjà établie.

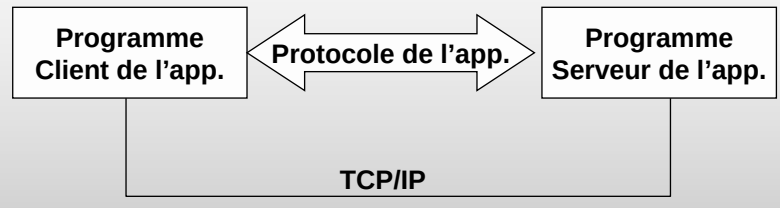
TCP/IP et INTERNET
TCP : déconnexion

- **Une connexion TCP est libérée en un processus dit "trois temps modifié":**



SERVICES INTERNET
MODELE CLIENT/SERVEUR

- **Modèle standard de programmation sur Internet : une application comprend deux programmes distincts :**



Programme Client

- **Interface d'utilisateur**
- **Non permanent**
- **Orienté graphique**
- **Langage intermédiaire**

Programme Serveur

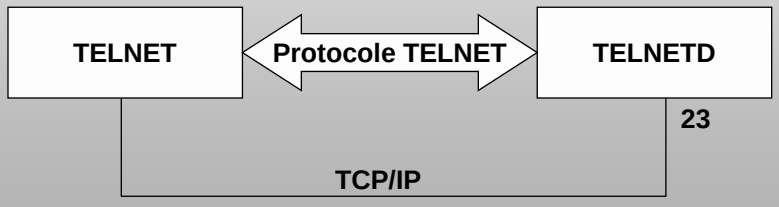
- **Réalisation de services**
- **Processus permanent**
- **Orienté traitement**
- **Ports TCP exclusifs**

SERVICES INTERNET
MODELE CLIENT/SERVEUR

- **CLIENT TRANSPARENT : Un programme client est dit transparent s'il affiche les données reçues du serveur d'une manière fidèle sans aucune transformation**
- **CLIENT RECURSIF : Un programme client est dit récursif s'il on peut le relancer à l'intérieur du fenêtre du même programme**
- **SERVICE ORIENTE SESSION : Un service Internet est dit orienté session s'il crée une connexion permanente pour chaque client qui devra être fermée par le client concerné**
- **SERVICE ORIENTE REQUETE : Un service Internet est dit orienté requête si la connexion avec un client sera fermée automatiquement par le serveur à la fin du traitement d'une requête client.**

SERVICES INTERNET
SERVICE TELNET

- **Service standard TELNET : Terminal Network Protocol**
  - **Objectif : pouvoir travailler avec une machine à distance à travers Internet**
  - **Principe : une fenêtre sur machine client joue le rôle de terminal de contrôle de la machine serveur**
  - **Propriétés : service orienté session, client transparent et récursif**
  - **Port TCP par défaut : 23**



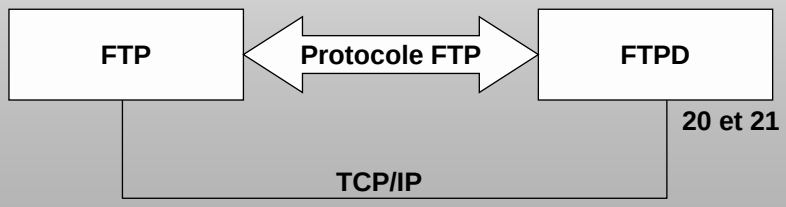
SERVICES INTERNET
SERVICE TELNET

- **Les commandes de TELNET :**

c	**close**	**Fermer la connexion**
d	**display**	**Afficher**
o	**open hostname [port]**	**Connexion à la machine hôte (port par défaut 23)**
q	**quit**	**Quitter Telnet**
set	**set options (taper 'set ?' pour lister)**	**Mettre des options**
sen	**Send strings**	**Envoi d'une chaîne de caractères au serveur**
st	**status**	**Lister les états**
u	**Unset options (taper 'unset ?' pour lister)**	**Enlever les options**
?/h	**help**	**Liste des commandes**

SERVICES INTERNET
SERVICE FTP

- **Service standard FTP : File Transfert Protocol**
  - **Objectif : permettre les échanges de fichiers entre la machine client et la machine serveur**
  - **Principe : un langage de commandes pour parcourir dans les deux hiérarchies de fichiers et pour échanger les fichiers**
  - **Propriétés : service orienté session, client non transparent et non récursif**
  - **Port TCP par défaut : 20 (commandes) et 21 (données)**



SERVICES INTERNET
SERVICE FTP

- **Les commandes de FTP :**

!/quit/bye	**Quitter FTP**
Open host	**Ouvrir une connexion**
open hostname	**Connexion à la machine hôte**
close	**Fermer une connexion**
user	**Compte d'utilisateur (login)**
pwd	**Pass word**
ls / dir	**Lister le répertoire courant sur serveur**
cd	**Changement de répertoire sur serveur**
? / help	**Liste des commandes**

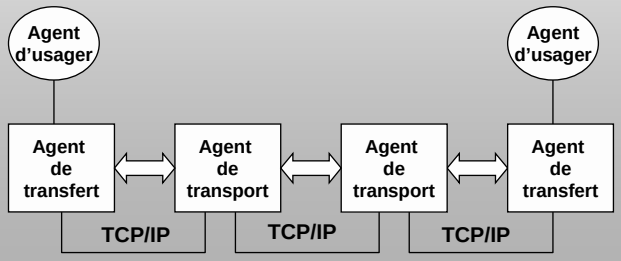
SERVICES INTERNET
SERVICE FTP

- **Les commandes de FTP :**

lcd	**Changement du répertoire sur la machine client**
ascii	**Envoi des fichiers textes avec conversion**
bin	**Envoi des fichiers sans aucune modification**
send / msend	**Envoi au serveur d'un ou de plusieurs fichiers**
get / mget	**Recevoir du serveur d'un ou plusieurs fichiers**
Mkdir/rmdir	**Création/supression d'un répertoire sur le serveur**
rename	**Change de nom d'un fichier sur le serveur**
delete	**Suppression d'un fichier sur le serveur**
status	**Les états**

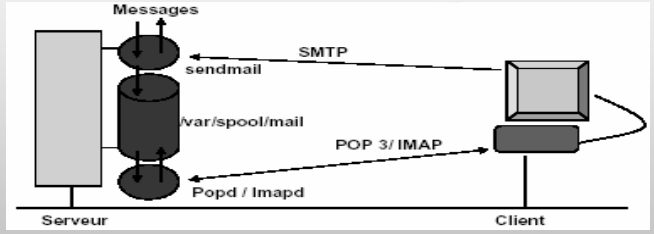
SERVICES INTERNET
SERVICE COURRIER ELECTRONIQUE

- **Service de courrier électronique se compose de plusieurs agents qui jouent des rôles symétriques :**
  - **Agent d'usager qui est l'interface d'utilisateur : outlook, eudora, Netscape, messager, … (protocoles POP3 ou IMAP)**
  - **Agent de transfert de messages qui gère les messages sur un site : sendmail, exchange, worldmail, …**
  - **Agent de transport de messages entre les sites : SMTP (Simple Mail Transport Protocol), UUCP, …**



SERVICES INTERNET
SERVICE COURRIER ELECTRONIQUE

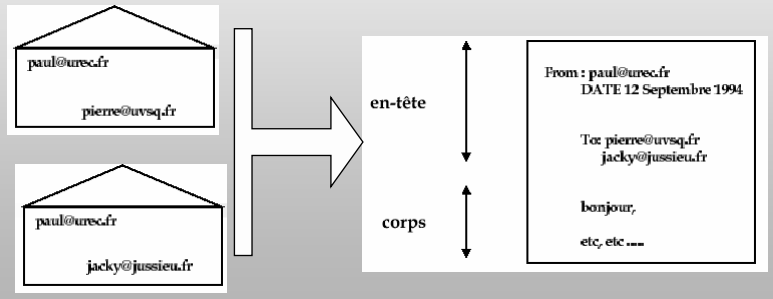
- **Modèle client-serveur pour les courriers électroniques :**
  - **Accès au boites à lettres des utilisateurs : POP 3 ou IMAP**
  - **Transfert du courrier émis => SMTP avec le serveur**



POP3	Post Office Protocol : transfert les nouveaux messages de la boîte aux lettre, sur le serveur, vers la machine cliente
IMAP	Interactive Mail Access Protocol : les courriers restent sur le serveur, sont triés et rapatriés à la demande de l'utilisateur, puis remis en place à la fin de la session

SERVICES INTERNET
SERVICE COURRIER ELECTRONIQUE

- **SENDMAIL :**
  - **Rôle de d'un centre tri de courrier : il possède des interfaces avec les agents d'usager (protocoles POP3 ou IMAP), des interfaces avec les agents de transport (protocole MAIL) et il gère les boîtes aux lettres locales**



SERVICES INTERNET
SERVICE COURRIER ELECTRONIQUE

- **SENDMAIL : RFC (Request For Comments)**
  - Des RFC (Request For Comments) définissent :
    - L'envoi,
    - La réception
    - La structure des adresses
    - Le format des lettres
  - Principaux RFC
    - RFC 822 Format des messages
    - RFC 821 Protocole SMTP
    - RFC 974 Courrier et DNS
    - RFC 1035 DNS
    - RFC 1123 prérequis pour les sites Internet

SERVICES INTERNET
SERVICE COURRIER ELECTRONIQUE

- **SENDMAIL : RFC 822 Format des messages**
  - Format de lignes en tête :

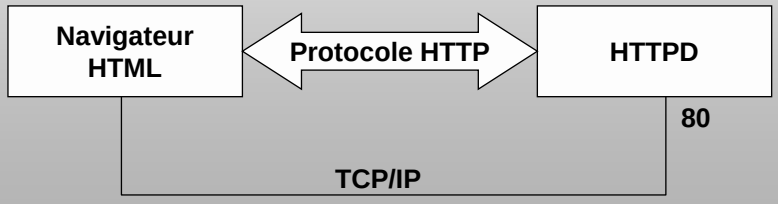
FROM	**Expéditeur**
TO	**Destinataire(s)**
CC	**Copie à**
BCC	**Copie aveugle**
REPLY-TO	**Adresse de réponse**
ERROR-TO	**Adresse en cas d'erreurs**
DATE	**Date d'expédition**
RECEIVED	**Information de transferts**
MESSAGE-ID	**Identificateur unique de message**
SUBJECT	**Sujet**

SERVICES INTERNET
SERVICE COURRIER ELECTRONIQUE

- **SENDMAIL : RFC 822 Format des messages**
  - Format de l'adresse électronique :
    - Personne@Machine.Domaines
      - **Exemple :** Nhan.Le-Thanh@nyx.unice.fr
    - Personne@Domaines
      - **Exemple :** Nhan.Le-Thanh@unice.fr
  - Remarques :
    - Pas de différence entre minuscules et majuscules
    - Attention aux caractères autorisées (limitation par RFC du DNS)
    - Adresse d'utilisateur qui reçoit tous les messages en erreur (postmaster) est obligatoire

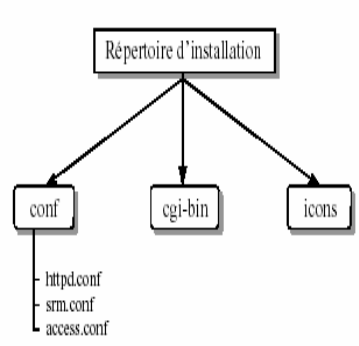
SERVICES INTERNET
SERVICE HTTP ET APACHE

- **Service HTTP : HyperText Transport Protocol**
  - **Objectif : Afficher et naviguer depuis la machine client les documents hypertextes stockés sur les serveurs http**
  - **Principe : Le Client dispose d'un langage intermédiaire, HTTP (HyperText Markup Language) permettant la présentation graphique**
  - **Propriétés : service orienté session, client non transparent et non récursif**
  - **Port TCP par défaut : 80**



SERVICES INTERNET
SERVICE HTTP ET APACHE

- **APACHE :** Le logiciel Apache est actuellement le logiciel serveur http le plus utilisé dans l'Internet. Doté de nombreuses fonctionnalités, performant et gratuit, il constitue un choix très intéressant pour ceux voulant mettre en place un service WWW.



L'installation du logiciel Apache se fait, par défaut, dans le répertoire (sous unix) **/usr/local/etc/httpd**.
Ce répertoire contient en particulier un répertoire **conf** qui va contenir les fichiers de configuration d'Apache :
- **httpd.conf** :directives de configuration générale
-**srm.conf** : directives de ressources du serveur
- **access.conf**:directives de la politique d'accès au serveur.

SERVICES INTERNET
SERVICE HTTP ET APACHE

- **Quelques directives principales d'APACHE**

httpd.conf	**User**	**Utilisateur, en général NOBODY**
	Group	**Groupe d'utilisateur, en général NOBODY**
	Port	**Port TCP, en général, port 80**
srm.conf	**DocumentRoot**	**Répertoire racine du site officiel en général :** /usr/local/etc/httpd/htdocs
	ScriptAlias	**Alias du Répertoire contenant des programmes CGI, en général** ScriptAlias /cgi-bin/ /usr/local/etc/httpd/cgi-bin/
	AddHandler	**Suffixe spécifique des fichiers CGI, par exemple :** AddHandler cgi-script .cgi

SERVICES INTERNET
SERVICE HTTP ET APACHE

- **Politique d'accès (access.conf)**
  - **Protection par domaine :** une protection par domaine, qui permet de définir des droits d'accès en fonction des noms de machines ou de domaines
  - **Protection par domaine :** une protection par utilisateur, qui permet de protéger tout ou partie du serveur par nom d'utilisateur et mot de passe
- **Directive 'Directory ' :** mettre en place une politique d'accès avec des sous directives suivantes
  - Option : définir les options d'accès aux répertoires et fichiers : Indexes (indique la liste des des répertoires accessibles), Includes (accepte des directives « Server Side Include » (SSI)), includesNOEXEC (même chose que Includes mais on interdit la commande #exec ainsi que l'inclusion de script CGI), FollowSymLinks (on autorise l'accès aux liens symboliques), ExecCGI (on autorise des programmes CGI dans cette arborescence)
  - AllowOverride : indiquer si on peut utiliser le fichier .htaccess pour protéger un répertoire : all (si oui), none (si non), AuthConfig (accepte des directives d'autorisation), FilleInfo accept des directives de contrôle), Indexes (contrôle de répertoires), Limit (sous bloc de définition des droits d'accès : allow, deny et order)

SERVICES INTERNET
SERVICE HTTP ET APACHE

- **Exemple 1 : Politique d'accès**

```
<Directory /usr/local/etc/httpd/htdocs>
    Options Indexes SymLinksIfOwnerMatch Includes
    AllowOverride None
    <Limit GET>
       order allow,deny
       allow from all
    </Limit>
</Directory>
<Directory /usr/local/etc/httpd/htdocs/docs>
    Options +ExecCGI
    AllowOverride None
    <Limit GET>
       order allow,deny
       allow from all
    </Limit>
</Directory>
```

<Limit> : est un bloc contenant des sous-directives permettant de définir les droits d'accès associés à une ou plusieurs méthodes d'accès (GET, POST,...) :
- ORDER : indique l'ordre de définition des droits : ORDER ALLOW DENY
ou ORDER DENY ALLOW
- ALLOW : autorise un ou plusieurs domaines
- DENY : interdit un ou plusieurs domaines
- require : dans le cas d'accès par utilisateur et mot de passe, indique les groupes ou les utilisateurs ayant accès.

SERVICES INTERNET
SERVICE HTTP ET APACHE

- **Exemple 2 : Protection par domaine**

```
<Directory /usr/local/etc/httpd/htdocs/>
    Options Indexes SymLinksIfOwnerMatch Includes
    AllowOverride None
    <Limit GET>
      order allow,deny
      allow from all
    </Limit>
</Directory>
<Directory /usr/local/etc/httpd/htdocs/local>
    <Limit GET>
       order deny,allow
      deny from all
      allow from .urec.fr
    </Limit>
</Directory>
```

SERVICES INTERNET
SERVICE HTTP ET APACHE

- **Exemple 3 : Protection par utilisateurs**

```
<Directory /usr/local/etc/httpd/htdocs/>
    Options Indexes SymLinksIfOwnerMatch Includes
    AllowOverride None
    AuthType Basic
    AuthUserFile /usr/local/etc/httpd/conf/htpasswd
    AuthGroupFile /usr/local/etc/httpd/conf/htgroup
    <Limit GET>
       order allow,deny
       allow from all
    </Limit>
</Directory>
<Directory /usr/local/etc/httpd/htdocs/prive1>
    AuthName Groupe Urec
    <Limit GET POST>
        require group urec
    </Limit>
</Directory>
<Directory /usr/local/etc/httpd/htdocs/prive2>
    AuthName Prive2
    <Limit GET POST>
        require user gross
    </Limit>
</Directory>
```

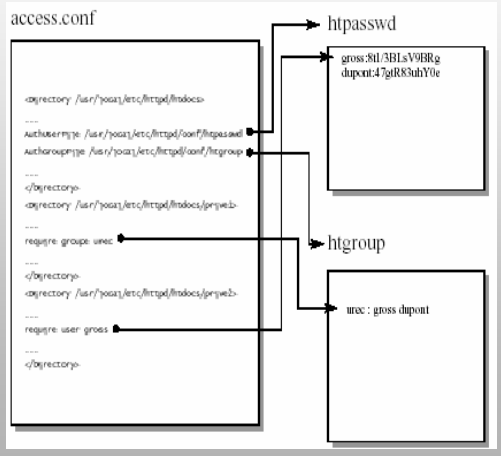
Dans cette exemple, on crée une protection par utilisateur pour tous les fichiers sous les arborescences /usr/local/etc/httpd/htdocs/prive1 et /usr/local/etc/httpd/htdocs/prive2.

-Dans le premier cas, l'accès est réservé aux utilisateurs appartenant au groupe urec
- Dans le deuxième, seul l'utilisateur gross aura droit d'accès.

SERVICES INTERNET
SERVICE HTTP ET APACHE

- **Exemple 3 : Protection par utilisateurs**

- AuthType : type d'authentification
- AuthUserFile : nom du fichier utilisateurs
- AuthGroupFile : nom du fichier groupe
- AuthName : chaîne de caractère utilisé dans la fenêtre du navigateur pour saisir le nom d'utilisateur et le mot de passe.
- require : indique les utilisateurs ou les groupes d'utilisateurs qui ont le droit d'accès.



SERVICES INTERNET
ports TCP des services standards

No port	Mot-clé	Description
20	FTP-DATA	File Transfer [Default Data]
21	FTP	File Transfer [Control]
23	TELNET	Telnet
25	SMTP	Simple Mail Transfer
37	TIME	Time
42	NAMESERVER	Host Name Server
43	NICNAME	Who Is
53	DOMAIN	Domain Name Server
79	FINGER	Finger
80	HTTP	WWW
110	POP3	Post Office Protocol - Version 3
111	SUNRPC	SUN Remote Procedure Call