

Jean-Yves Didier & Samir Otmane
LSC – Université d'Evry

{didier, otmane}@iup.univ-evry.fr}
<http://lsc.univ-evry.fr/~otmane/>

Définition générale

- Réseau :
 - Ensemble d'objets ou de personnes connectés ou maintenus en liaison,
 - Par extension, l'ensemble des liaisons établies,
 - Vient du latin *rete* qui signifie *filet*,
 - *Les objets reliés sont appelés “noeuds du réseau”.*
- *Exemples:*
 - *Réseau social, réseau ferroviaire, réseau téléphonique, réseau informatique, etc ...*

- Définition :
 - Ensemble de machines interconnectées qui servent à échanger des flux d'information,
 - Un réseau répond à un besoin d'échanger des informations.

- Attention ! Le terme réseau peut désigner :
 - L'ensemble des machines,
 - Le protocole de communications,
 - La manière dont les équipements sont connectés.

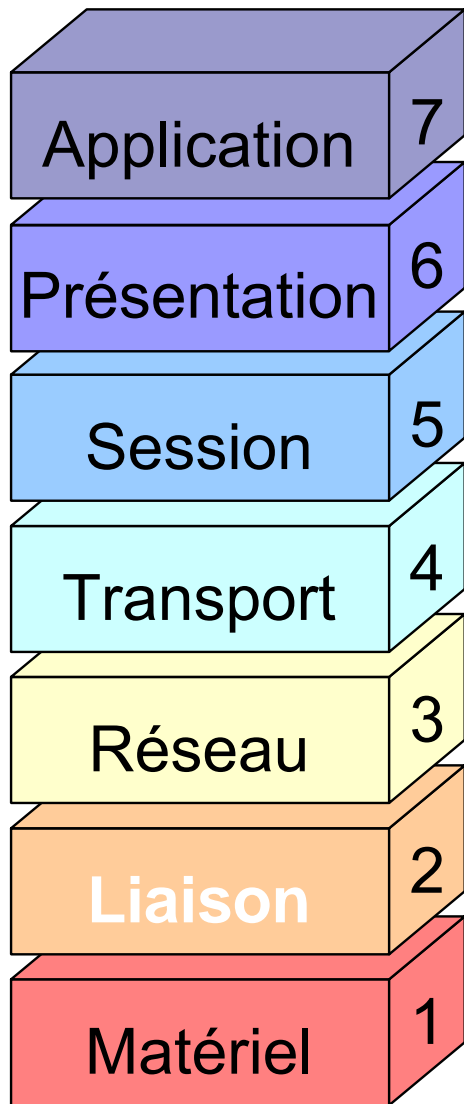
Echelle géographique

- PAN, LAN, MAN, WAN :
- PAN : **Personal Area Network**
 - Réseau personnel (< dizaine de machines).
- LAN : **Local Area Network**
 - A l'échelle d'un bâtiment (ex: IUP).
- MAN : **Metropolitan Area Network**
 - A l'échelle d'une ville ou d'un campus (ex : REVE).
- WAN : **Wide Area Network**
 - A l'échelle d'un pays ou mondiale (ex: Renater).

Normalisation OSI

- Pourquoi normaliser ?
 - Échanges profitables si tout le monde se comprend !
- Deux stratégies de circulation de l'information:
Messages complets (inusité) ou fragmentés en *paquets*.
- Norme OSI de l'ISO :
 - OSI : Open Systems Interconnections, créé en 1984,
 - S'intéresse aux réseaux à *commutations de paquets*,
 - Modèle à 7 couches employé lors de la conception :
 - Mise en place d'un réseau : 1 solution par couche,
 - La modification d'une couche n'affecte pas les autres.

Normalisation OSI

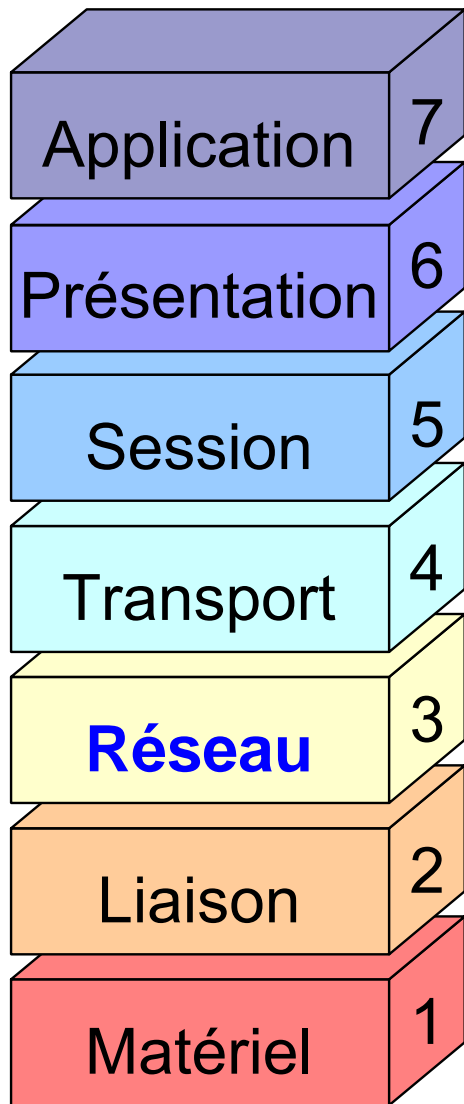


Couche 2 : Liaison

Problèmes :

- Comment identifier deux stations sur le même support physique ?
- Comment transmettre sans erreur les données d'une station à une autre sur le même support physique ?

Ex : ethernet, token ring.

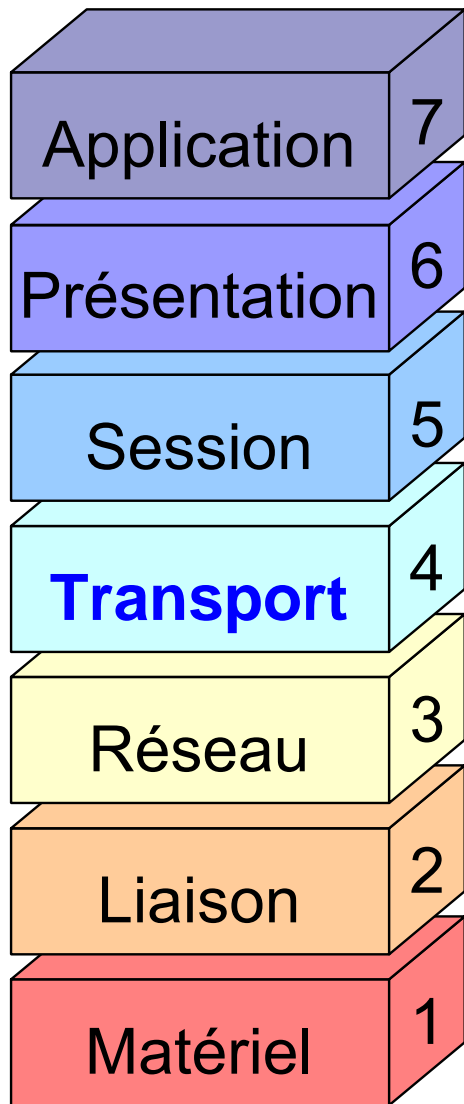


Couche 3 : Réseau

Problèmes :

- Comment acheminer un paquet entre 2 stations qui ne sont pas sur le même support physique (*routage*) ?
- Comment assurer l'interconnexion de réseaux hétérogènes ?
- Comment contrôler et réguler le trafic sur le réseau ?

Ex: protocole IP



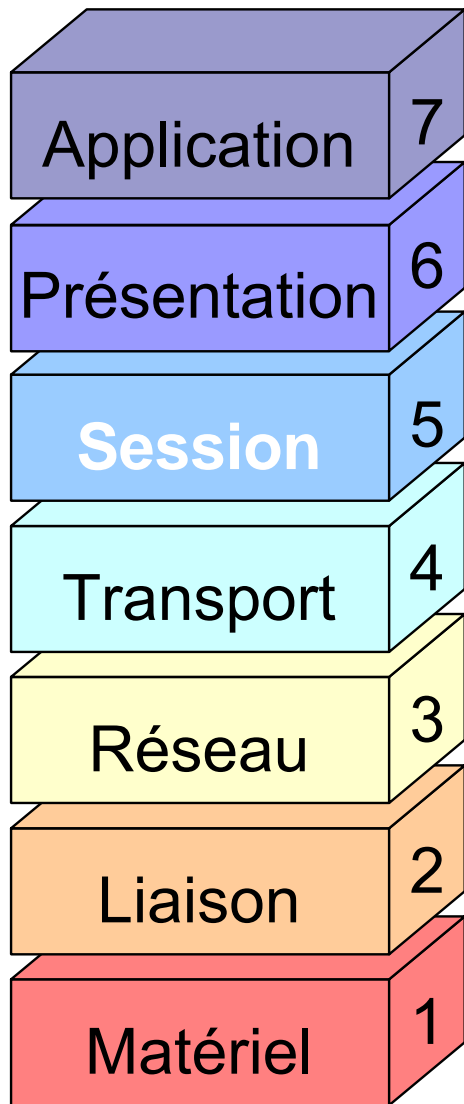
Couche 4 : Transport

Problèmes :

- Comment découper les messages en paquets ?
- Comment s'assurer de leur bonne réception ?
- Comment reconstituer le message à partir des paquets ?

Ex: Protocoles TCP, UDP

Normalisation OSI

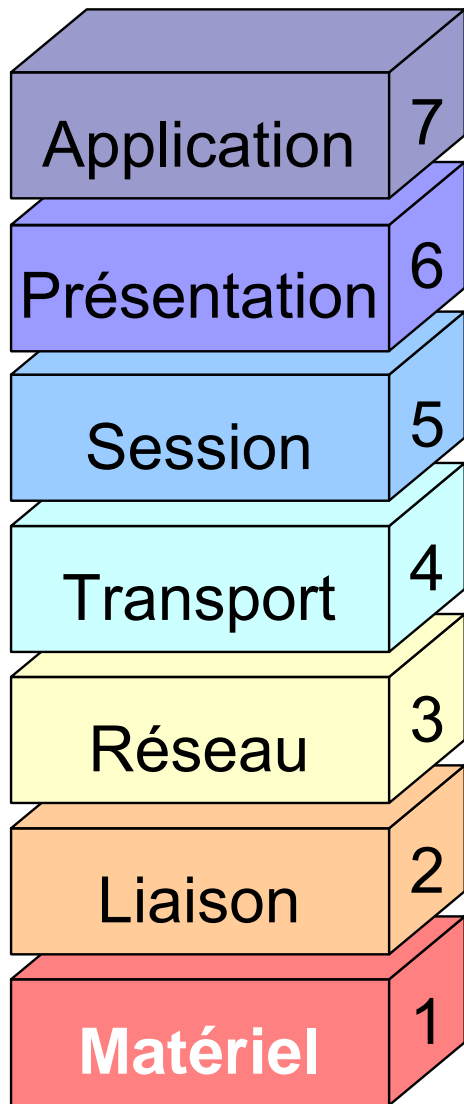


Couche 5 : Session

Problèmes :

- Comment établir une session entre deux utilisateurs distants ?
- Comment gérer les problèmes de synchronisation ?

Ex: Protocole RPC



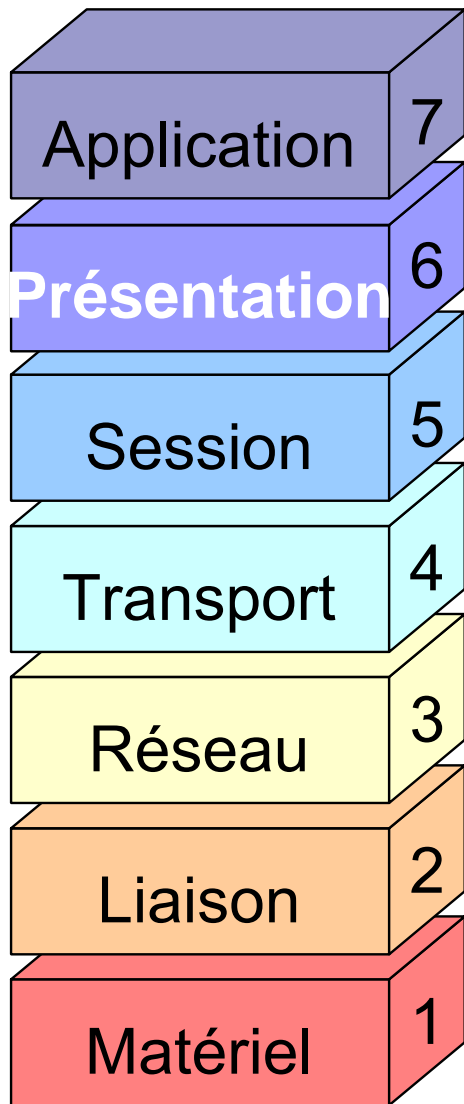
Couche 1 : Matériel

Problèmes à résoudre :

- caractéristiques du support physique pour le réseau :
 - Pour du câble : type, blindage, type de signal, nature des signaux, limitations,
 - Communications hertziennes : fréquences, type de modulation,
 - Fibre optique : couleur du laser, section du câble, nombre de brins

- Topologie du réseau :
 - Cablâge en maille, bus, anneau, étoile, etc ...

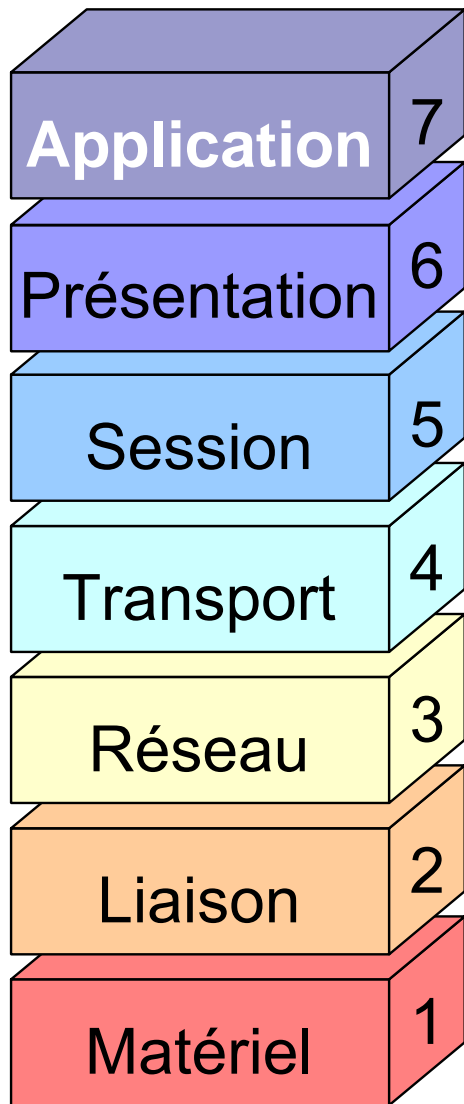
Normalisation OSI



Couche 6 : Présentation

Problèmes :

- Quelle est la forme de l'information transmise ?
- Comment les données sont elles codées ?
- Doit on compresser ou crypter les données ?



Couche 7 : Application

Problèmes :

- Quels sont les protocoles spécifiques aux programmes applicatifs ?

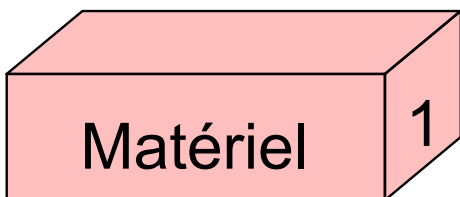
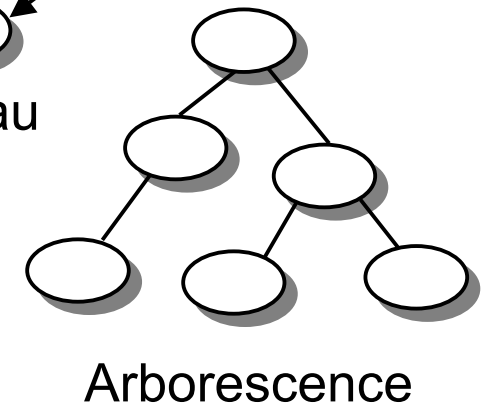
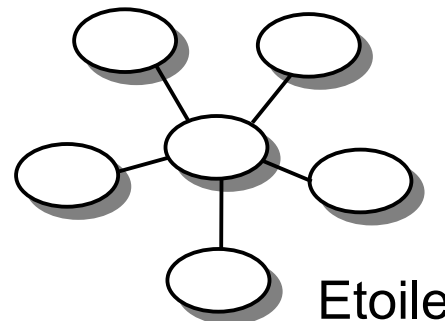
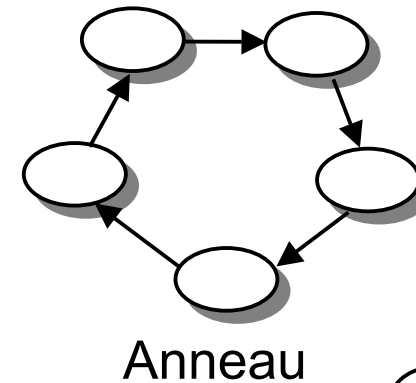
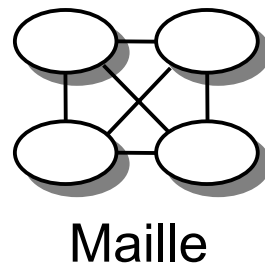
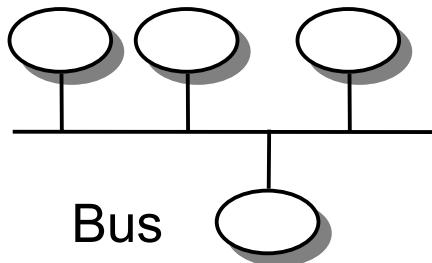
Ex de protocoles :

- POP3, IMAP, SMTP : e-mail,
- Ftp : transferts de fichiers,
- Http : transferts de pages web,
- Etc ...

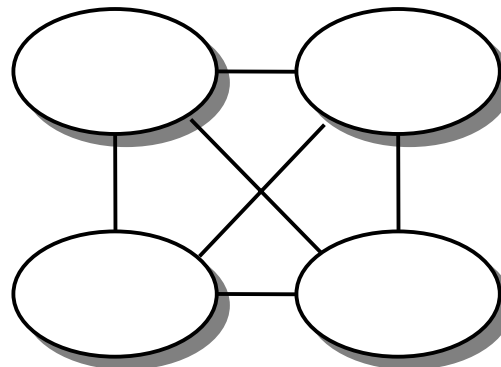
• Problème :

- Pour connecter 2 ordinateurs, un fil suffit.
- Comment connecter N ordinateurs pour que chaque ordinateur puisse communiquer avec n'importe quel ordinateur ?

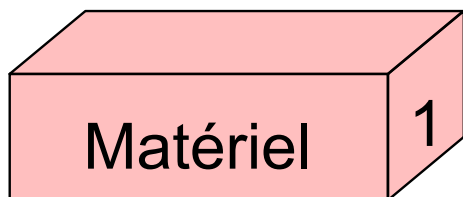
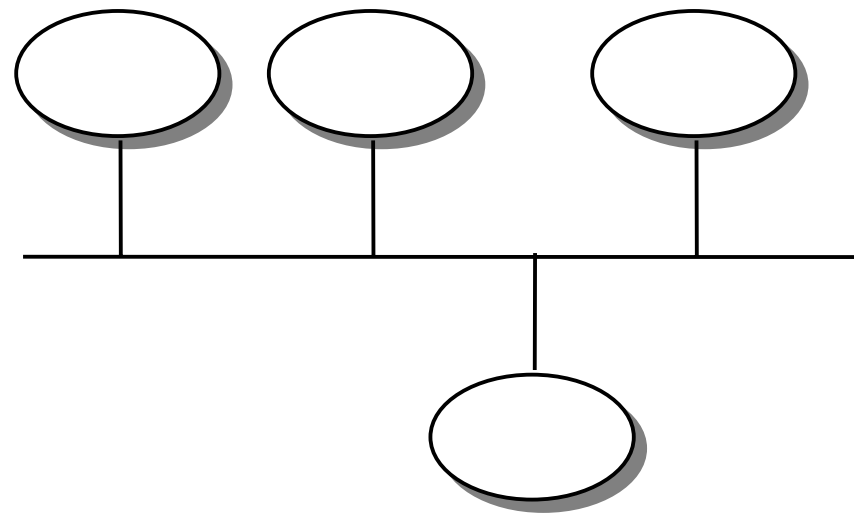
• Topologies :



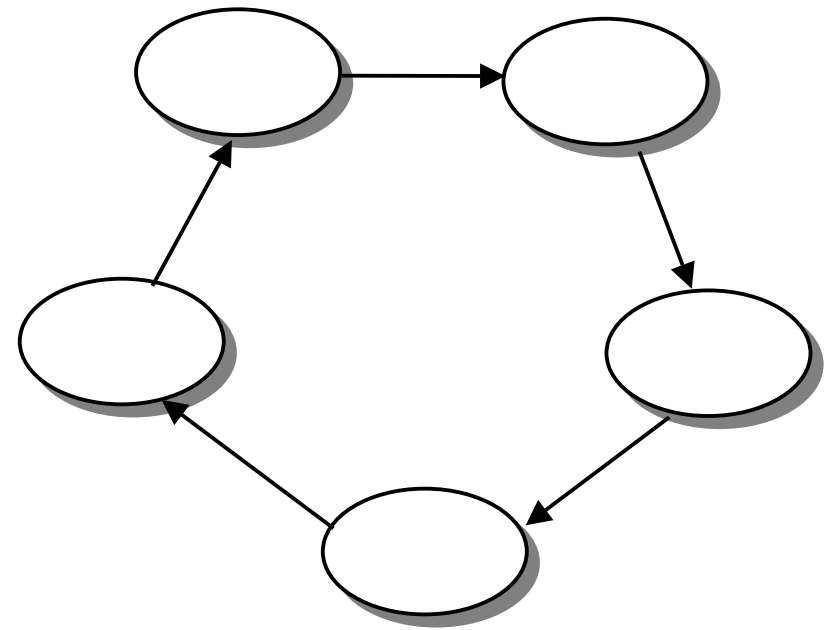
- Généralisation du cas à 2 ordinateurs,
- Chaque machine est reliée à toutes les autres par un câble,
- **Inconvénient majeur** : nécessite beaucoup de câbles (pour n machines, il faut $n(n-1)/2$ câbles),
- Inusité de nos jours.



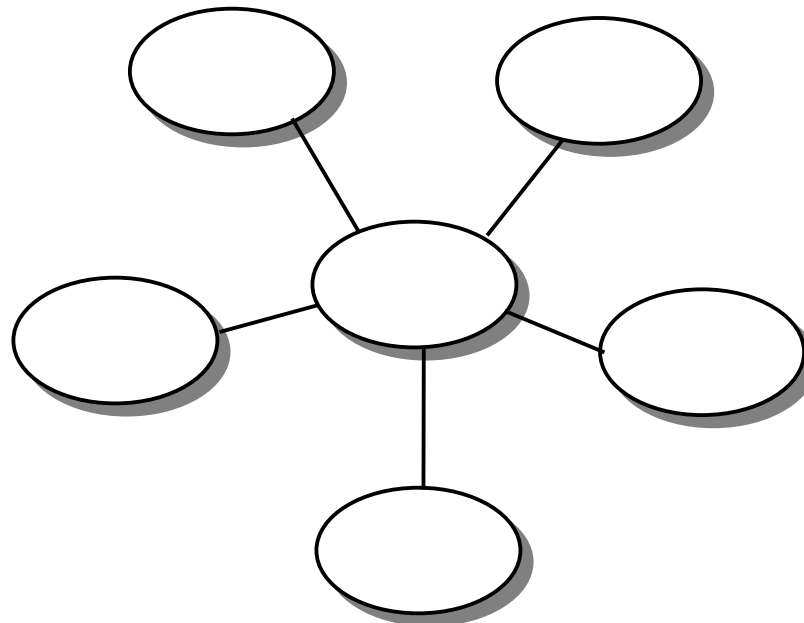
- Toutes les stations sont reliées à un support commun,
- Problème de partage du support physique (collisions).



- Les stations sont enchaînées les unes aux autres pour former un anneau,
- L'anneau est unidirectionnel,
- **Inconvénient** : si une machine tombe en panne, le réseau est coupé,
- **Solution** : un réseau à double anneau



- Toutes les stations sont reliées à un noeud central (le câblage en arborescence est un généralisation du câblage en étoile),
- **Inconvénient:** la fiabilité du réseau est conditionnée par le noeud central

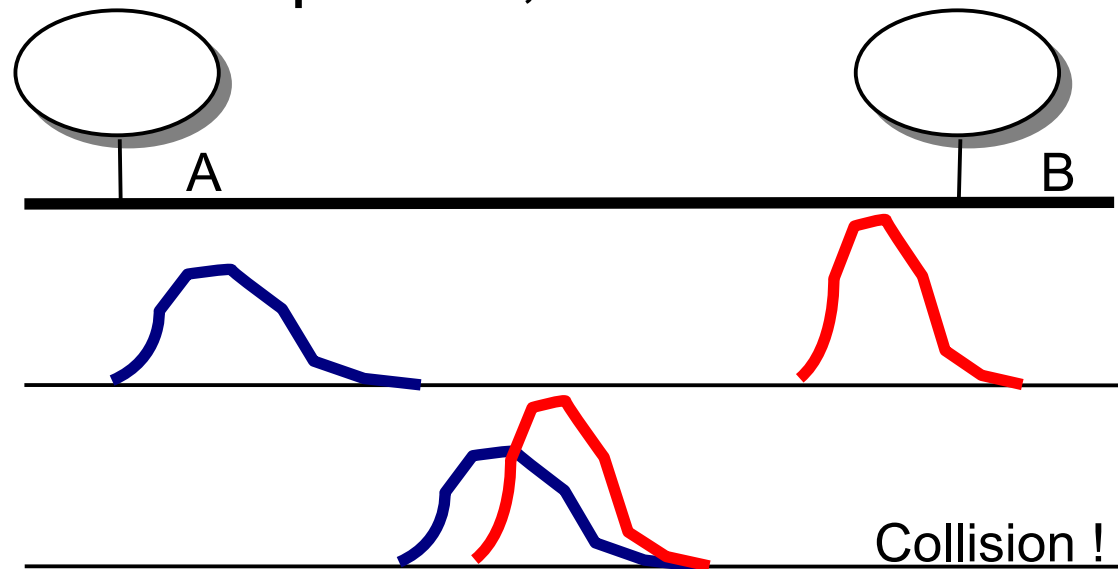


- Tiens à la fois de l'étoile et du bus,
- Le centre de l'étoile: appareil actif qui duplique l'information sur chacun des câbles,
- Panne du réseau = panne du centre de l'étoile, nécessité d'un appareil actif fiable,
- Système de câblage répandu car permet d'utiliser les câbles du réseau téléphonique.



- Problématique :

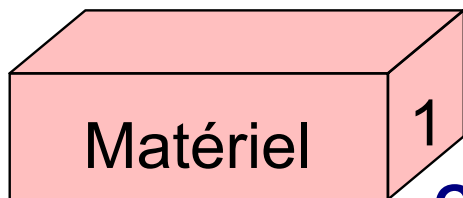
- N ordinateurs cherchent à accéder au canal de transmission,
- Collision : si deux ordinateurs transmettent en même temps, une collision se produit,



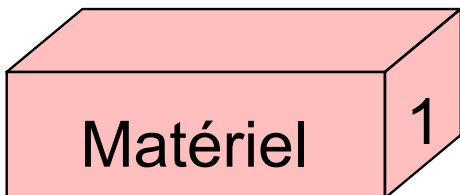
Allocation statique

- N ordinateurs, réseau de capacité C bits/sec.
 - Idée : réserver C/N bits/sec par ordinateur,
 - Utiliser du multiplexage temporel pour réguler la transmission,
 - Conséquence :
 - Chaque utilisateur obtient C/N bits du débit total,
 - Satisfaisant pour les réseaux téléphoniques,
 - Insatisfaisant en cas d'utilisation sporadique,
 - > mauvaise gestion du canal,
 - > trouver d'autres méthodes plus efficaces.

- Principe :
 - Définir des règles de contrôle d'accès,
 - Apprendre la politesse aux ordinateurs,
 - Règles de politesse :
 - Ecouter le canal avant de commencer à transmettre,
 - Ne pas transmettre si quelqu'un transmet déjà,
 - Valable dans un réseau local.
- Solution :
 - Accès par compétition : Ecoute de la porteuse CSMA/CD,
 - Accès par élection : Techniques à jeton.



- Caractéristique :
 - Topologie en bus,
 - Accès simultanés au support possible (Multiple Access),
 - Ecoute et détection du signal sur le réseau (Carrier Sense),
- Principe: CSMA
 - **Si aucun signal détecté Alors émettre,**
 - **Si signal détecté Alors différer la transmission,**
- **Problème : transmission simultanée = collision**

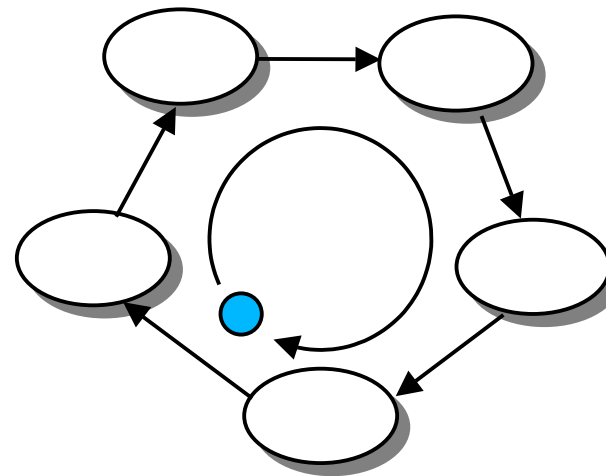


- Solution : CSMA/CD (Collision Detection)
 - Minimiser les pertes par détection de collisions,
 - Ecoute préalable + écoute pendant la transmission d'un message pour détecter une collision,
 - Ecoute pendant 2*temps de propagation vers le point le plus éloigné du bus,
 - **Si collision alors arrêt de la transmission et ré-émission après un temps tiré aléatoirement**

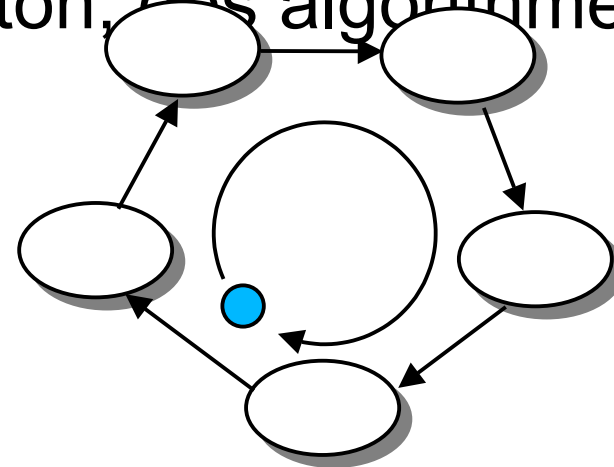
Ex : réseau Ethernet



- Caractéristique :
 - Topologie en anneau,
 - Une seule trame circule en permanence,
 - Une seule station transmet à tout moment,
 - Le jeton contrôle l'accès au support.



- Une station qui souhaite émettre :
 - Capture le jeton quand il passe à sa portée,
 - Emet une trame,
 - Constate que le destinataire a reçu le message,
 - Libère le jeton et le passe à la station suivante,
 - En cas de destruction du jeton, des algorithmes permettent de le régénérer.



- Normes pour les réseaux locaux (février 1980),
- Compatible OSI bien que antérieure,
- Modèle en 12 catégories :

Numéro	Objet de la norme	Nom anglophone
802.1	Fonctionnement inter-réseaux	INTERNETWORKING
802.2	Le contrôle des liaisons logique	Logical Link Control
802.3	Les réseaux locaux en bus logique	Ethernet LAN
802.4	Les réseaux locaux en bus à jeton	Token Bus LAN
802.5	Le réseau local en anneau logique	Token Ring LAN
802.6	Les réseaux métropolitains MAN	Metropolitan Area Network
802.7	La transmission en large bande	Broadband Technical Advisory Group
802.8	La fibre optique	Fiber-Optic Technical Advisory Group
802.9	Les réseaux intégrant la voix et les données	Integrated Voice / Data Networks
802.10	La sécurité des réseaux	Network security
802.11	Les réseaux sans fil	Wireless network

Matériel

1

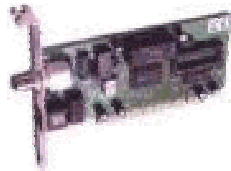
- Caractéristique : (IEEE 802.3 ou ISO 8802.3)
 - Topologie en **bus**, en **anneau** ou en **étoile**,
 - **Contrôle d'accès au support de type CSMA/CD**

Norme	Débit	Support	Longueur max	Exemple
802.3 10B5	10Mbit/s	Coaxial 50W	500m	Ethernet standard
802.3 10B2	10Mbit/s	Coaxial 50W	200m	Ethernet fin
802.3 10Broad36	10Mbit/s	Coaxial 75W	3600m	
802.3 1B5	1Mbit/s	Paire torsadée	500m, 5 hubs	Starlan
802.3 10BT	10Mbit/s	Paire torsadée	100m, hubs illimités	Starlan
802.3 10BF	10Mbit/s	Fibre optique	2 km	Starlan



- Câblages courants :

- câble coaxial (BNC – ethernet fin):



Carte BNC



Câble BNC

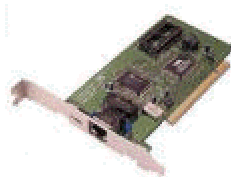


T - BNC



Terminateur

- paire torsadée (RJ45 - starlan):



Carte RJ45



Câble RJ45



Hub



Switch

Matériel

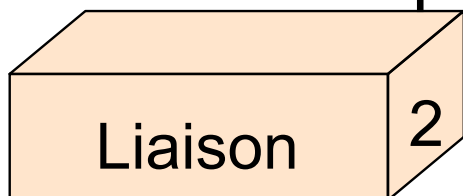
1

- Caractéristiques :
 - Topologie en **anneau**,
 - Contrôle d'accès : **token ring**,
 - **Support : paire torsadée**,
 - **Limité à 254 machines**,
 - **Convient aux environnement temps réel.**



Couche 2 : Liaison

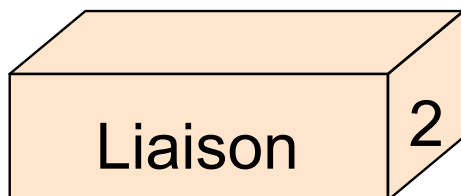
- Partage du support physique (couche 1):
 - toute trame envoyée est écoutée par toutes les machines,
 - > Comment déterminer le destinataire de la trame ?
 - > Comment déterminer la source de la trame ?
- Solution :
 - Donner une adresse physique aux machines,
 - Incorporer dans la trame les adresses physiques.



- Adresse physique :
 - Codée entre 1 et 254 sur le connecteur par des interrupteurs.

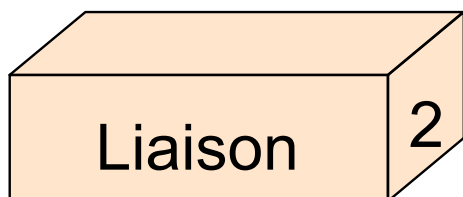
- Trame : ~ 8 à 2052 octets

Début de message	Adresse destination	Adresse source	Type de trame	Données	Fin de message	Parité	Refus
10 bits	8 bits	8 bits	24 bits	0 à 16352 bits	9 bits	1 bits	1 bits



- Adresse physique :
 - Codée sur la carte réseau (adresse MAC pour **M**edia **A**ccess **C**ontrol),
 - L'adresse physique est unique au monde !!!
 - 48 bits pour l'adresse : $2^{48} \sim 2,8 \times 10^{14}$ machines.
- Trame : ~ 72 à 1526 octets

Préambule	Adresse destination	Adresse source	Type de trame	Données	CRC
64 bits	48 bits	48 bits	16 bits	368 à 12000 bits	32 bits



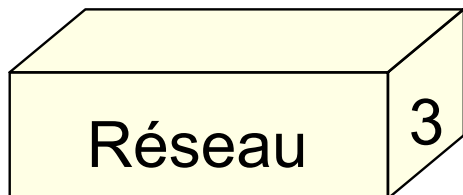
Obtenir l'adresse physique
(en mode administrateur)

Unix : ifconfig

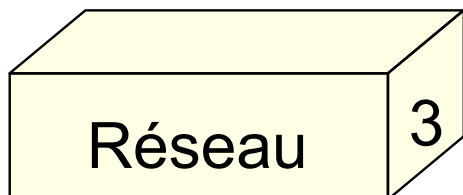
Windows : ipconfig /all

Couche 3: Réseau

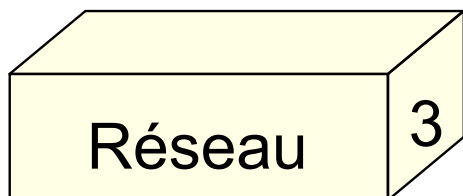
- Fonctions de la couche réseau :
 - Traduire les adresses logiques en adresses physiques,
 - Router les messages en fonction de leur priorité et l'état du réseau,
 - Gérer le trafic sur le réseau,
 - Gérer la commutation,
 - Contrôler l'encombrement des messages sur le réseau,
 - Découper et réassembler les messages en fonction de la capacité de la carte réseau,



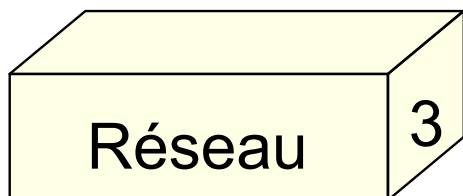
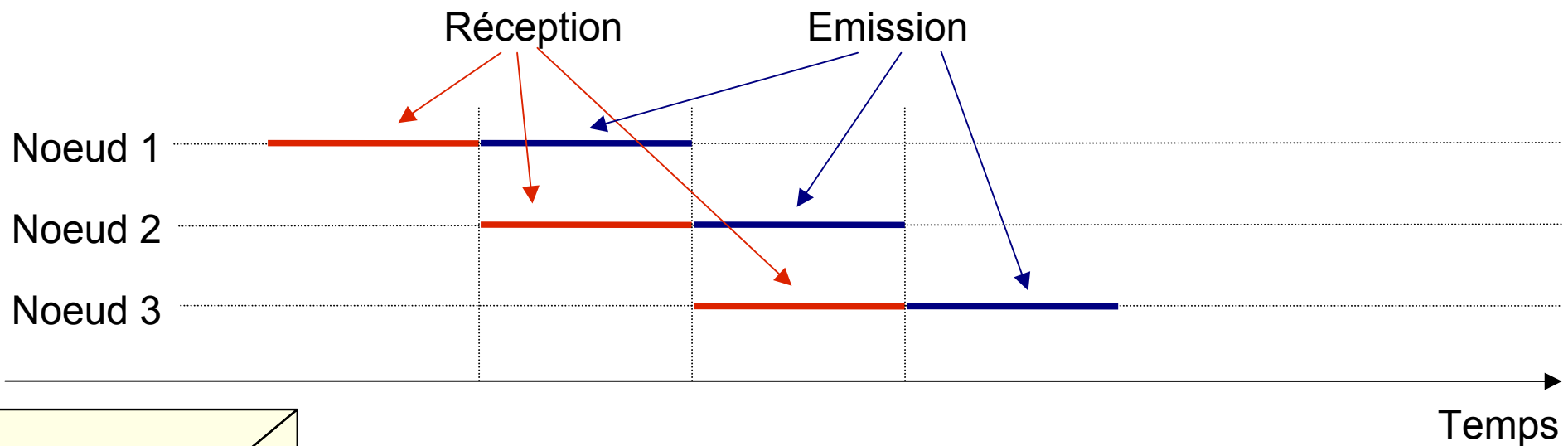
- La connexion : mise bout à bout de liens et de commutateurs,
- 5 techniques de commutation :
 - Commutation de circuits,
 - Commutation de messages,
 - Commutation de paquets,
 - Commutation de trames,
 - Commutation de cellules.



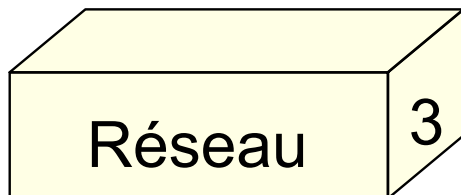
- Technique adaptée aux flux d'information (voix),
- Chaque communication passe par 3 phases :
 - Etablissement de la liaison : chercher et occuper un itinéraire (décrocher, composer, sonner),
 - Maintien de la liaison pendant toute la durée de la connexion,
 - Libération des connexions sur ordre et retour à l'état libre.



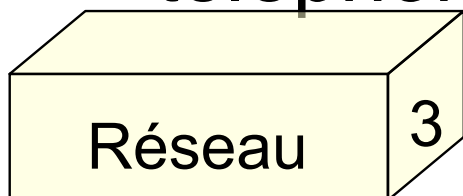
- Le message transite de noeuds en noeuds jusqu'au destinataire,
- Un noeud ne peut envoyer de message tant qu'il ne l'a pas reçu complètement,



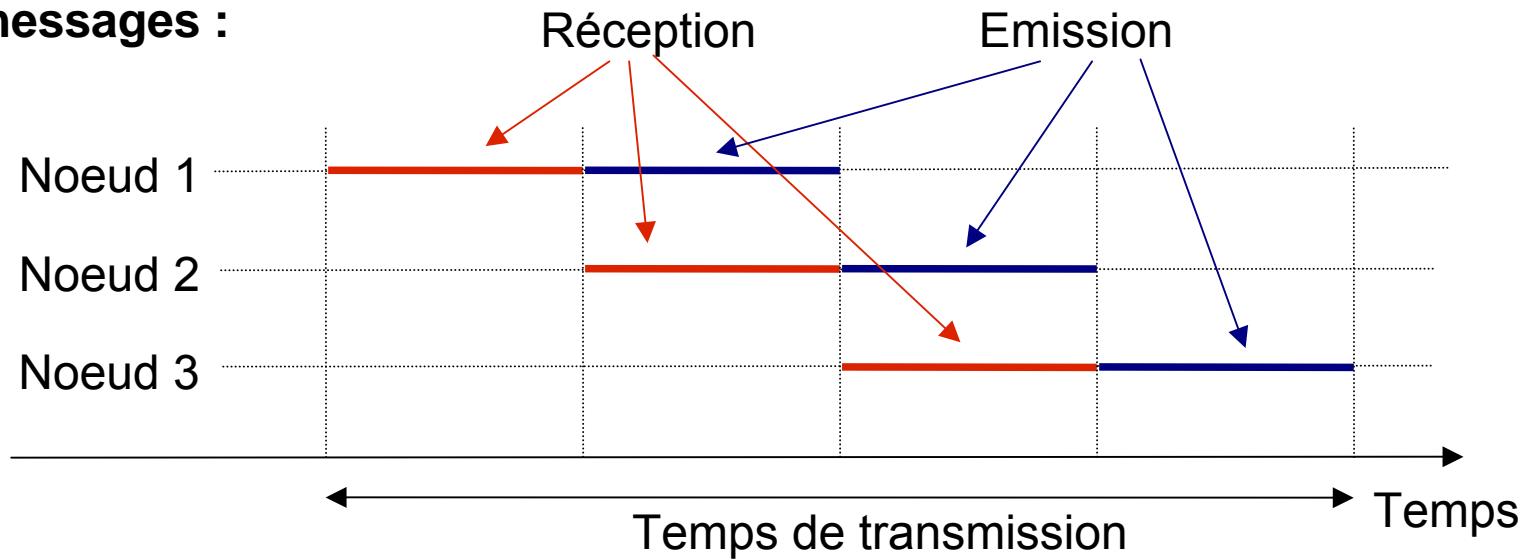
- Les messages sont découpés en paquets de faible longueur. 2 modes de service :
- Service en mode connecté (ex : TRANSPAC):
 - Les paquets utilisent toujours le même chemin.
- Service en mode non connecté (ex : Internet):
 - Les paquets empruntent des itinéraires différents,
 - Le noeud de commutation aiguille les paquets,
 - Problème : Comment réassembler les paquets ?



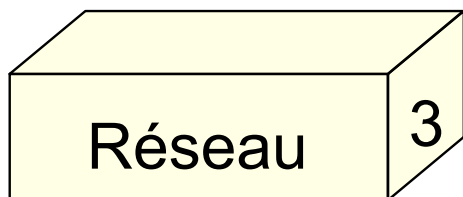
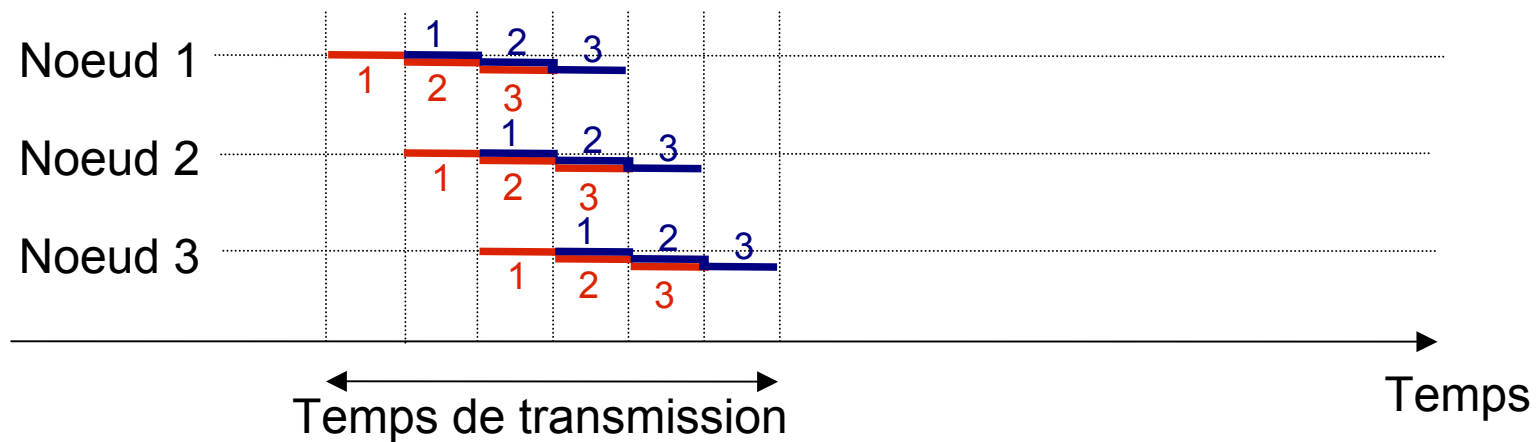
- Informations dans les en-têtes des paquets :
 - Source,
 - Destination,
 - Numéro de séquence,
 - Bloc de contenu de données,
 - Code de vérification des erreurs.
- Norme internationale X25, oeuvre des sociétés téléphoniques.



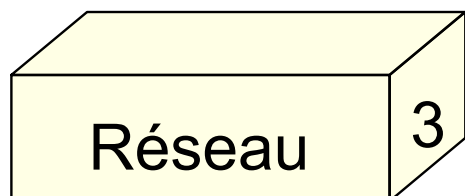
Commutation de messages :



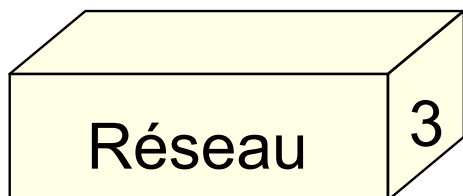
Commutation de paquets :



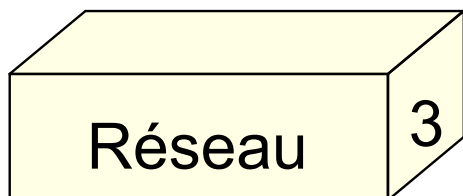
- Extension de la commutation de paquets,
- Les commutateurs de trame traitent des entités de niveau 2
 - Commutateurs plus simples, moins chers,
 - Les fonctionnalités de niveau 3 sont assurées au niveau 2
- Ex: commutation ethernet :
 - Paquet = trame ethernet
 - Adresse = adresse ethernet (adresse MAC)



- Combine les avantages de la commutation de circuits et la commutation de paquets,
- Les paquets (cellules) ont une longueur fixe de 53 octets (5 octets d'en-tête),
- Avant toute émission de cellule, une connexion doit être mise en place,
- Ex: réseau ATM (Asynchronous Transfer Mode).

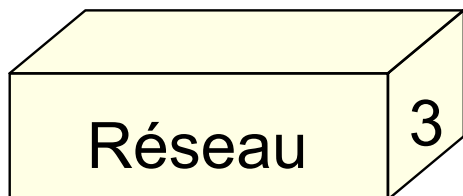


Contraintes	Circuits (RNIS)	Paquets (X25)	Cellules (ATM)
Temps réel	Oui	Non	Oui
Transparence	Oui	Non	Oui
Protocole de bout en bout	Oui	Non	Oui
Débit variable	Non	Oui	Oui
Multiplexage statistique	Non	Oui	Oui

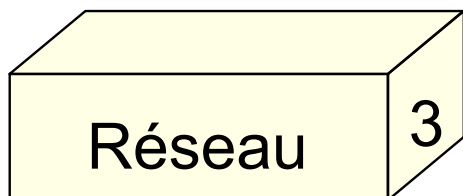


Suites de protocoles

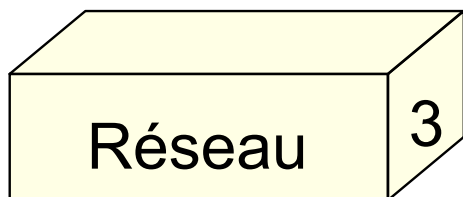
- La Norme OSI garantit l'indépendance des couches mais :
 - Nombre de protocoles réseaux développés avant que la norme n'existe,
 - Dans la réalité les couches se retrouvent interdépendantes.
- Les protocoles s'organisent en familles ou en suites :
 - La suite IP (internet): ARP, RARP, ICMP, etc ...
 - La suite IPX (Novell, jeux en réseau): RIP, etc ...
 - La suite NetBIOS (Réseau local Microsoft).



- Protocoles employés pour Internet,
- Développé en 1er par l'armée américaine pour :
 - Échanger les informations entre les bases,
 - Trouver un moyen d'échanger des données même si une partie du réseau est détruite.
- Chronologie :
 - 1er prototype : ARPANET (1969),
 - Développement du protocole TCP/IP (1974),
 - Dans les années 80, naissance d'internet,
 - 1992, fondation de l'Internet Society

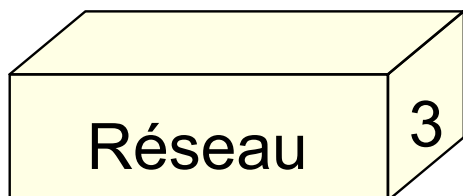


- Au niveau 3, les protocoles IP courants sont :
 - IP : adressage et fragmentation des paquets,
 - ARP: retrouve l'adresse physique à partir de l'adresse logique,
 - RARP : la conversion inverse,
 - ICMP : gestion d'erreurs,
 - RIP : routage des paquets.



Protocole IP

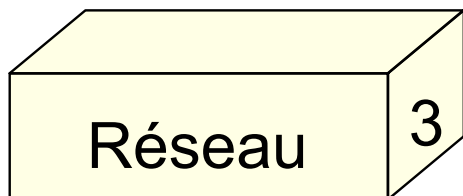
- IP (Internet Protocol) :
 - Gère les adresses et la fragmentation des paquets,
 - Spécification complète : RFC 791
- RFC (Request for Comments)
 - Série de documents techniques et organisationnels au sujet d'Internet,
 - Les RFC font office de standards,
 - <http://www.rfc-editor.org> (liste complète en anglais),
 - <http://abcdrfc.free.fr/> (traduction partielle en français).



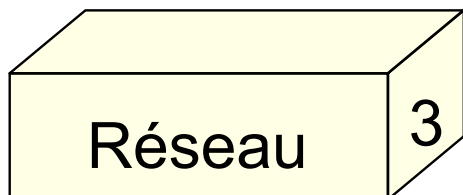
- Fonctionnalités :

- Achemine un paquet en fonction de l'adresse destinataire,
- Chaque paquet (datagramme) est indépendant,
- 4 mécanismes clés pour les services:
 - Type of service, indique la qualité de service désirée,
 - Time to live (TTL), donne l'espérance de vie maximale du paquet,
 - Options, fonctions de contrôle supplémentaires,
 - Header checksum, fonction de vérification des données.

- L'adresse IP est une adresse logique, pourquoi est elle nécessaire ?
 - Adresse physique = une machine,
 - Les machines sont regroupées en réseau,
 - Comment identifier le réseau ?
 - En attribuant une adresse logique.
 - Pourquoi identifier le réseau ?
 - Pour permettre à deux machines de réseaux différents de communiquer entre elles.



- Format d'une adresse IP: w.x.y.z (4 octets) avec w,x,y,z compris entre 1 et 254 (0 réservé pour le réseau, 255 pour le broadcast).
- Une adresse w.x.y.z peut se lire comme suit:
 - Machine d'adresse w.x.y.z,
 - Machine d'adresse z du réseau w.x.y.0,
 - Machine d'adresse y.z du réseau w.x.0.0,
 - Machine d'adresse x.y.z du réseau w.0.0.0 .
- Ces lectures favorisent le routage des paquets.



Classe	Valeur de w	Lg adresse réseau	Nb de réseaux	Nb max de machines
A	0 – 127	1 octet	127	16777216
B	128 – 191	2 octets	16384	65536
C	192 – 223	3 octets	2097152	256
D	224 – 239			
E	240 – 255			

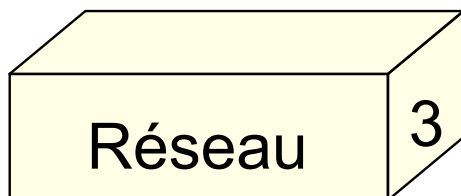
- Plages d'adresses réservées pour les réseaux locaux :

- 10.0.0.1 à 10.255.255.254,
- 172.16.0.1 à 172.31.255.254,
- 192.168.0.1 à 192.168.255.254,

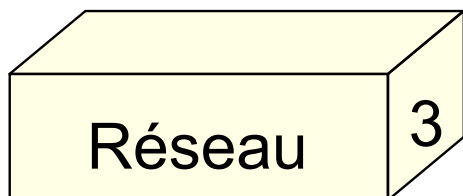
Obtenir l'adresse IP
(en mode administrateur)

Unix : ifconfig
Windows : ipconfig /all

- Adresse réservée pour les tests : 127.0.0.1



- Pourquoi ?
 - Utilisation hétérogène de moyens de couche 1,
 - Réduction de l'encombrement,
 - Economise les temps de calculs,
 - Isolation d'un réseau,
 - Renforcement de la sécurité,
 - Optimisation de l'espace réservé à une adresse IP.



- Ils permettent de segmenter un réseau en plusieurs sous-réseaux.

- Exemple de masque :

– 255.255.255.224 => 11111111.11111111.11111111.11100000

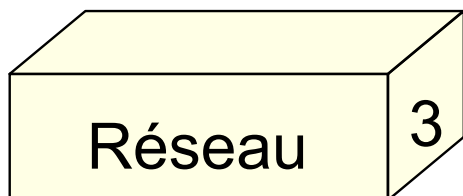
← Réseau classe C → ↑ hôte

- Détermination du sous-réseau d'une machine : sous-réseau

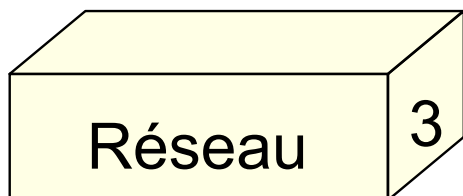
– 200.100.40.33 => 11001000.01100100.00101000.00100001

– On effectue et ET logique avec le masque de sous-réseau :

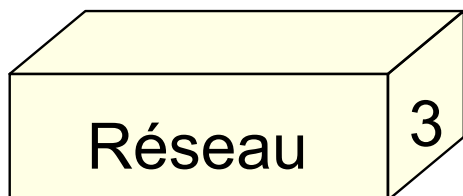
– 200.100.40.32 => 11001000.01100100.00101000.00100000



- Nombre de sous réseaux : **2 RFC s'appliquent:**
 - RFC 1860 : $2^n - 2$, n étant le nombre de bits à 1
 - RFC 1878 : 2^n
- ⇒ Adresse des sous réseaux
- Adresse de diffusion :
 - Mettre tous les bits de la partie hôte à 1
- Nombre de machines du sous réseau :
 - $2^m - 2$, m étant le nombre de bits de la partie hôte



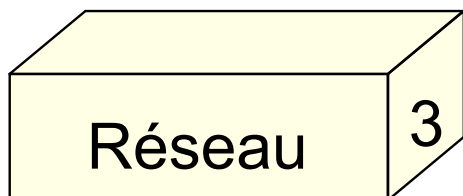
- Organisme IANA (Internet Assigned Numbers/ Naming Authority):
 - Distribue les adresses IP aux FAI (Fournisseurs d'accès à Internet).
- Organisme InterNIC (Internet Network Information Center) (AFNIC en France - <http://www.nic.fr>) :
 - Attribution des parties d'identifiant réseau pour les dispositifs directement reliés à internet.
- **Chaque noeud relié à Internet doit posséder une adresse IP unique !**



- Format de l'en-tête :

32 bits

Version	IHL	Type of service	Total length	
Identification		Flags	Fragment Offset	
Time to live	Protocol	Header Checksum		
Source address				
Destination address				
Options			Padding	



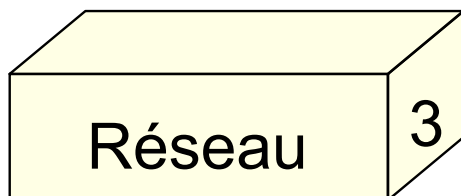
En-tête datagramme IP

32 bits

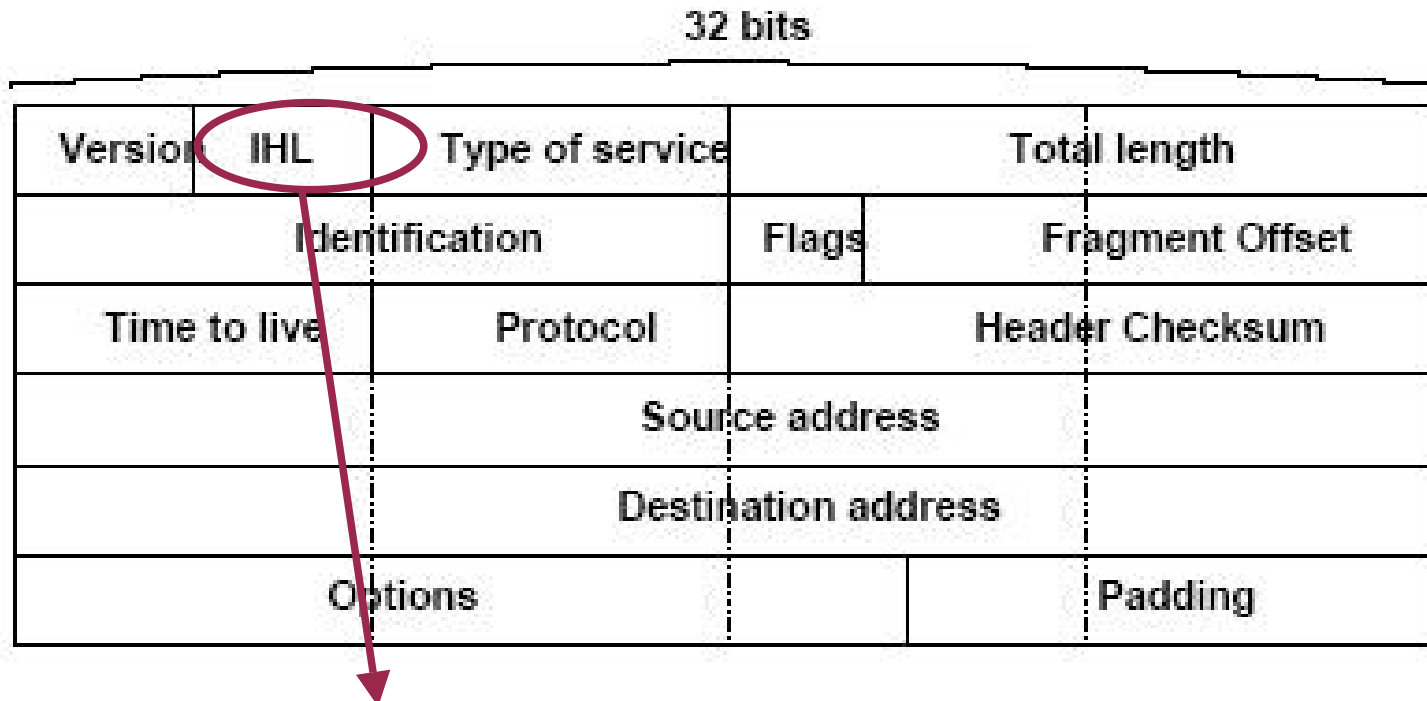
Version	IHL	Type of service	Total length	
Identification		Flags	Fragment Offset	
Time to live	Protocol	Header Checksum		
Source address				
Destination address				
Options			Padding	

Version du protocole IP (4bits):

- 00 réservé
- 04 IPv4
- 05 ST Datagram Mode
- 06 IPv6
- 15 réservé

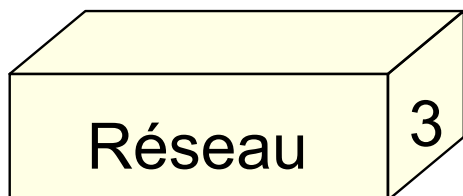


En-tête datagramme IP

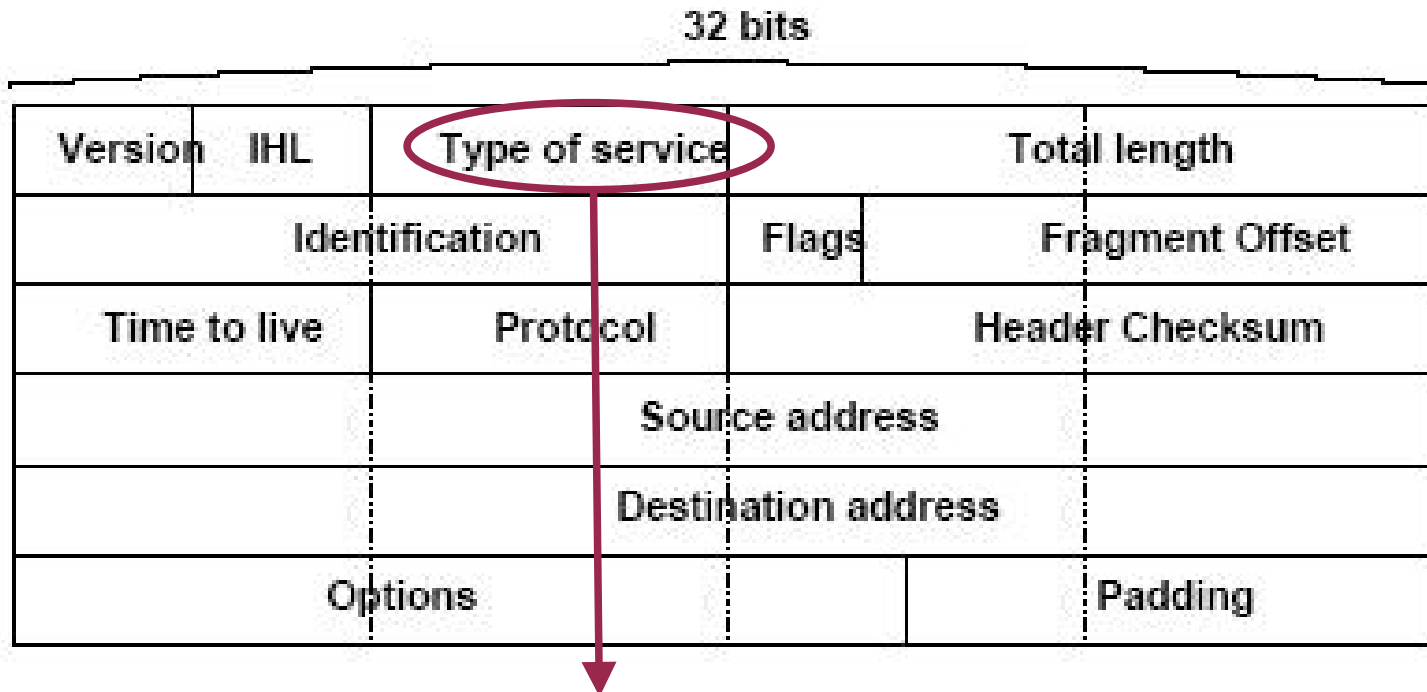


IHL (Internet Header Length) (4bits):

- Compte le nombre de mots de 32 bits constituant l'en-tête,
- Par défaut 5,
- Compris entre 6 et 15.

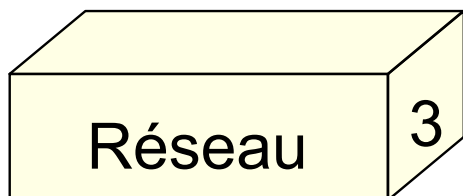


En-tête datagramme IP



Type of service (8bits):

- Essentiellement de la qualité de service,
- Règle la priorité, le délai, le débit et la fiabilité du paquets.



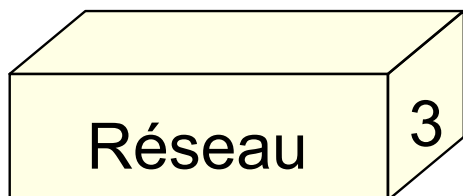
En-tête datagramme IP

32 bits

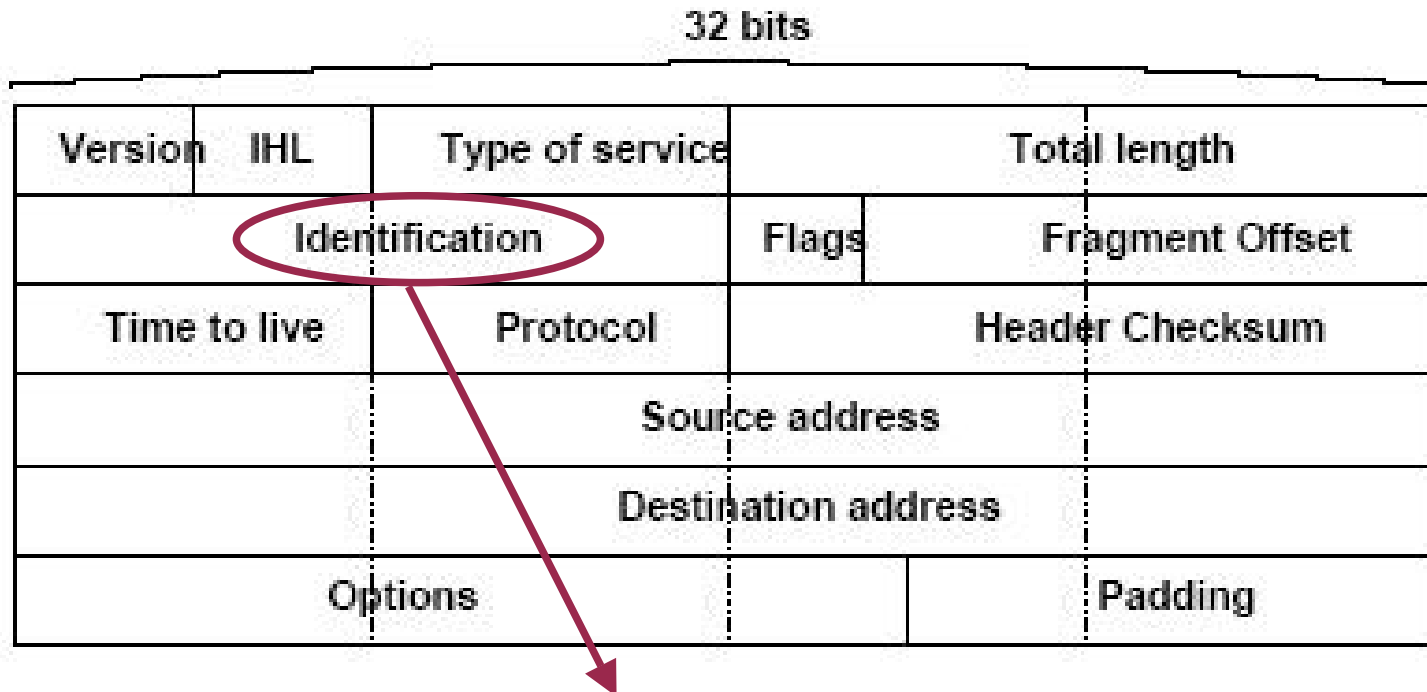
Version	IHL	Type of service	Total length	
Identification		Flags	Fragment Offset	
Time to live	Protocol	Header Checksum		
Source address				
Destination address				
Options			Padding	

Total length (16 bits):

- Longueur du paquet en octets,
- Sont inclus en-tête et données,
- 65535 octets au maximum,
- Dans la pratique 576 au plus.

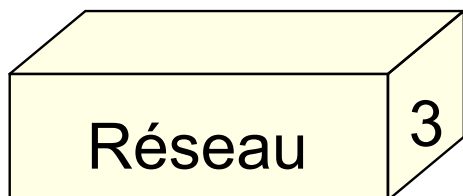


En-tête datagramme IP



Identification (16 bits):

Valeur donnée par la station émettrice aidant à reconstruire les fragments d'un message. Les détails se trouvent dans la RFC 815.



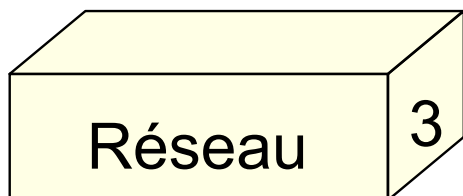
En-tête datagramme IP

32 bits

Version	IHL	Type of service	Total length	
Identification		Flags	Fragment Offset	
Time to live	Protocol		Header Checksum	
Source address				
Destination address				
Options			Padding	

Flags (3 bits):

- Bit 0 : 0
- Bit 1 : 0- may fragment, 1- don't fragment
- Bit 2 : 0- last fragment, 1- more fragments



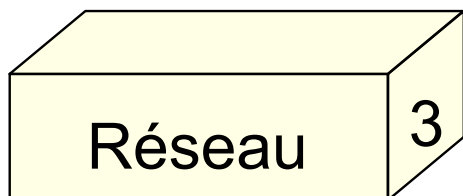
En-tête datagramme IP

32 bits

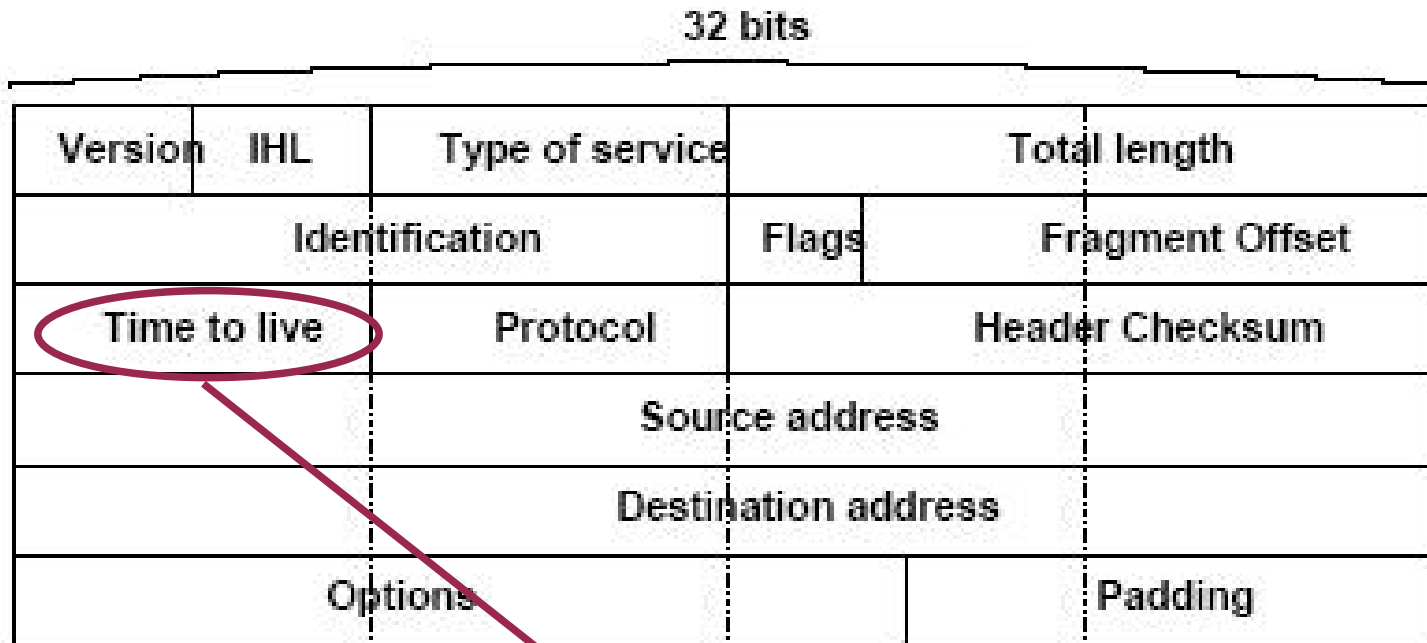
Version	IHL	Type of service	Total length	
Identification		Flags	Fragment Offset	
Time to live	Protocol		Header Checksum	
Source address				
Destination address				
Options			Padding	

Fragment offset (13 bits):

Indique la place du fragment dans le paquet. Le premier fragment a donc un 'offset' de 0. L'offset est exprimé en mots.

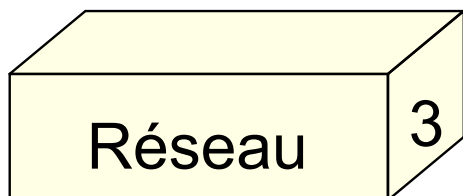


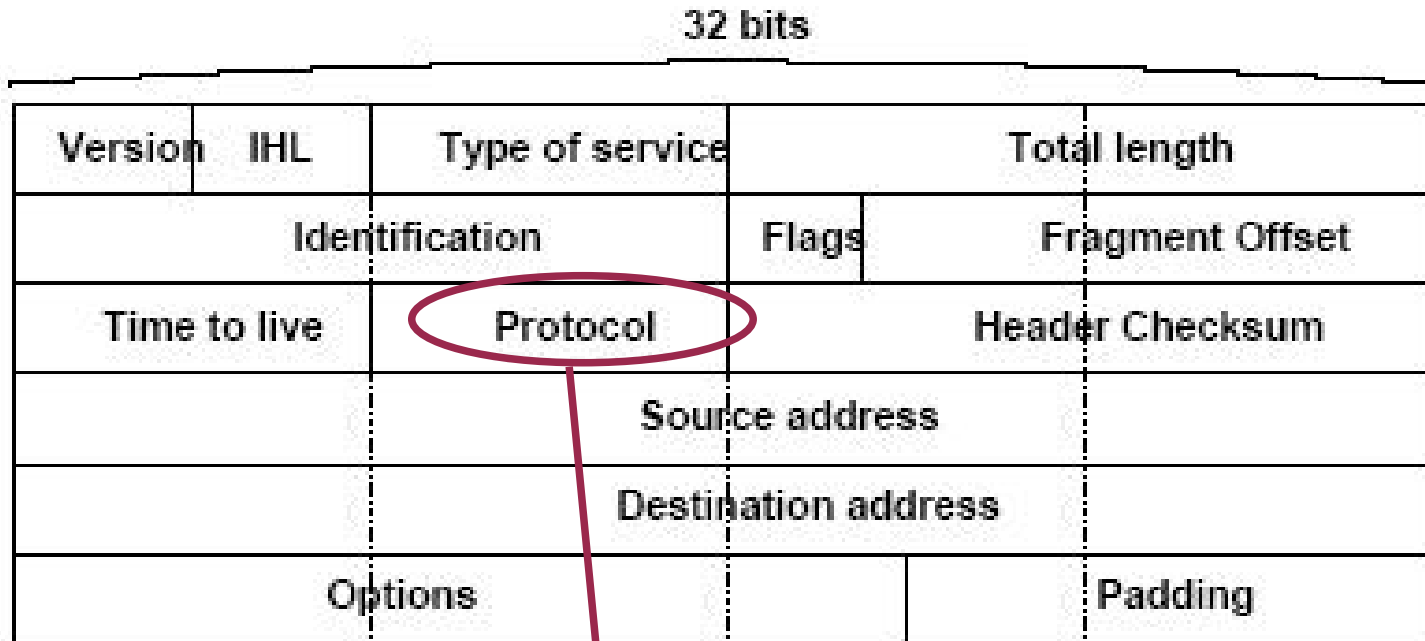
En-tête datagramme IP



Time to Live -TTL (8 bits):

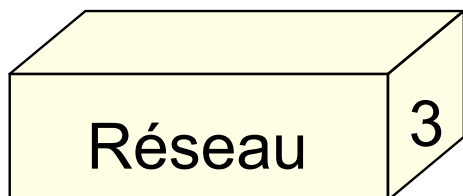
- Indique la durée de vie maximal du paquet,
- Si TTL = 0, le paquet est détruit,
- Décrémenté à chaque changement de réseau,
- Permet de détruire un paquet qui ne peut pas être acheminé à la destination.



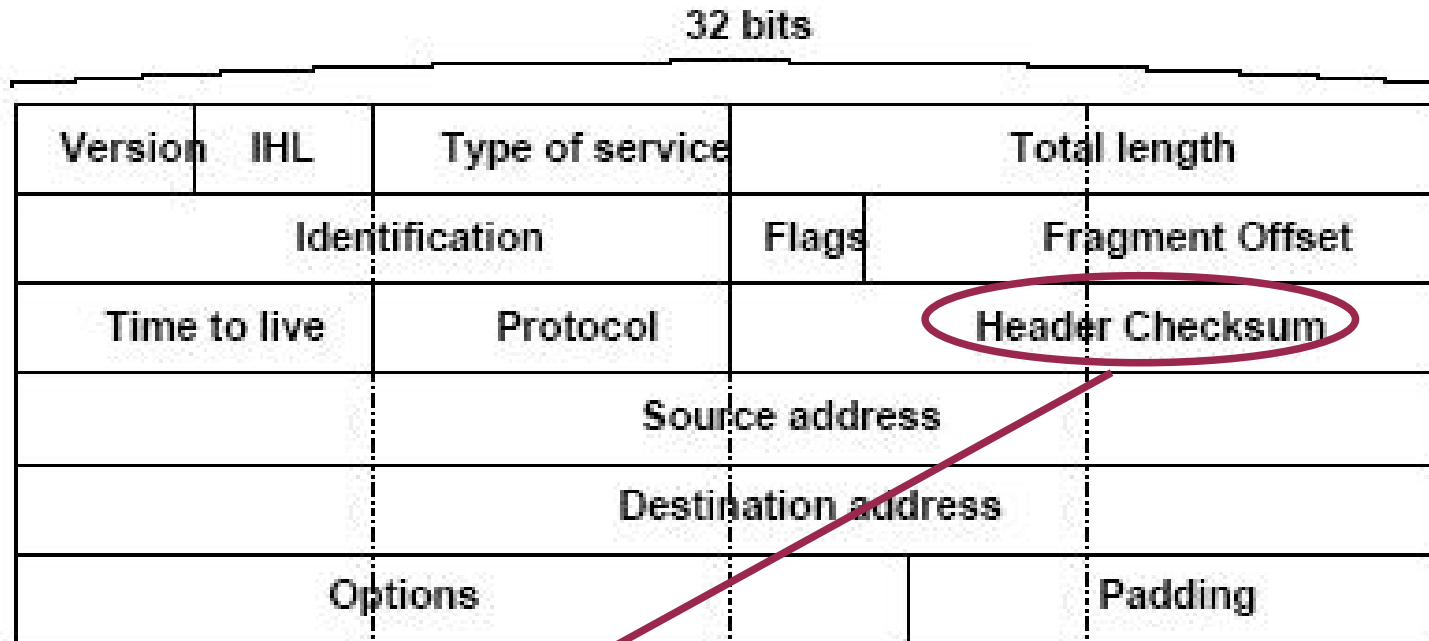


Protocol (8 bits):

- Un identifiant de protocole employé,
- Liste des identifiants dans la RFC 1340,
- Parmi les plus connus :
 - 01 ICMP
 - 06 TCP (protocole de niveau 4)
 - 17 UDP (protocole de niveau 4)

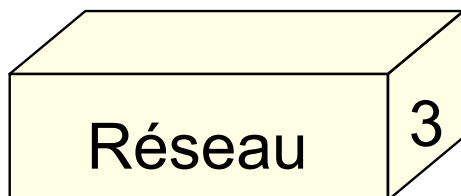


En-tête datagramme IP

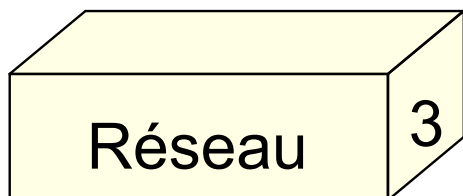


Header Checksum (16 bits):

- Contrôle de la validité de l'en-tête du paquet,
- Algorithme détaillé dans la RFC 1071.

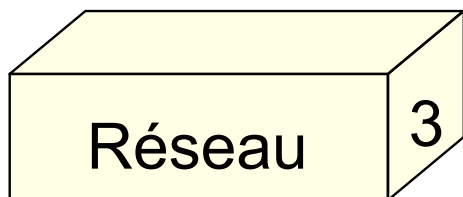


- Rappel :
 - les messages sont décomposés en paquets,
 - Les paquets sont véhiculés par les trames des réseaux,
- Que se passe t'il dans le cas où le changement de réseau implique un changement de la taille de trame ?
 - Dans le cas d'une diminution de taille, il faut fragmenter le paquet et pouvoir le reconstituer après.



Protocole ARP

- Sur un réseau local, ARP permet d'obtenir l'adresse physique à partir de l'adresse logique.
- ARP : Address Resolution Protocol
- Protocole ARP (RFC 826):
 - Émission d'une trame ARP à destination du réseau,
 - La machine visée se reconnaît et répond par une nouvelle trame ARP,
 - L'émetteur reçoit la réponse et connaît l'adresse matérielle de la machine cible.



Trame ARP

- Encapsulée dans une trame du réseau :
 - Exemple : dans une trame Ethernet :

Préambule	Adresse destination	Adresse source	Type de trame	Données	CRC
64 bits	48 bits	48 bits	16 bits	368 à 12000 bits	32 bits

0x0806

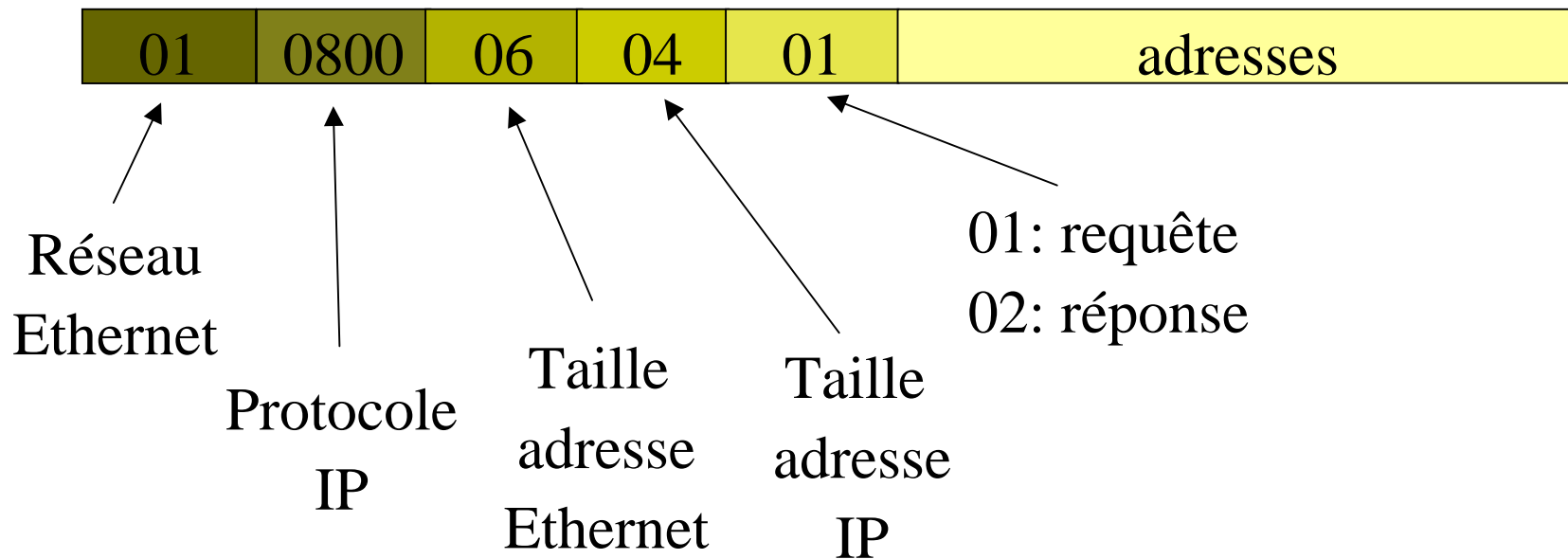
Type mat.	Type prot.	Taille mat.	Taille prot.	Op	Adresse physique émetteur	Adresse logique émetteur	Adresse physique récepteur	Adresse logique récepteur
16 bits	16 bits	8 bits	8 bits	16 bits				

Réseau

3

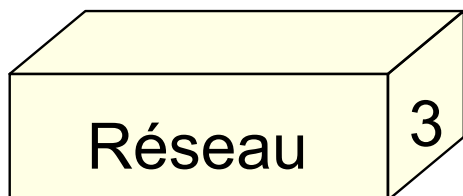
Trame ARP

- Pour Ethernet : Requête/ Réponse :

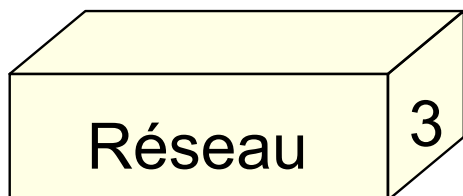


- Pour la requête, l'adresse destination Ethernet est :

- Dans l'en-tête de trame Ethernet : ff:ff:ff:ff:ff:ff
- Dans la trame ARP : 00:00:00:00:00:00



- ARP permet de trouver l'adresse physique à partir de l'adresse logique,
- Toute communication employant le protocole IP commence par une requête ARP préalable,
 - => On réduit l'encombrement réseau en stockant les résolutions déjà effectuées sur la machine,
 - => Cela s'appelle le cache ARP,
 - => Il faut toutefois le vider régulièrement !

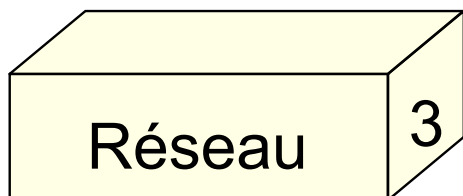


Voir le cache ARP
(en mode administrateur)

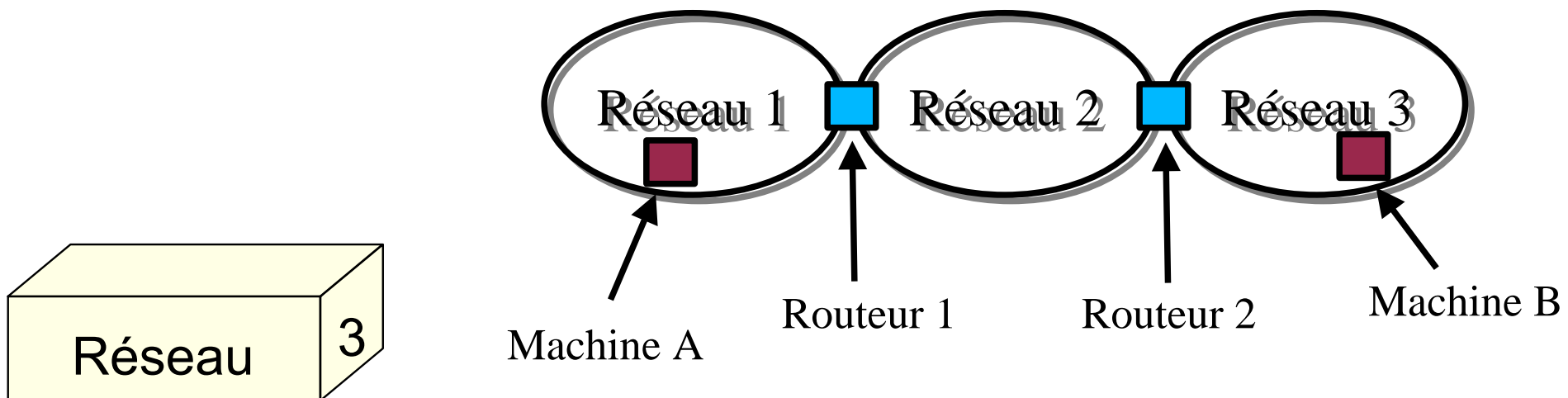
Unix	:	arp [-n]
Windows	:	arp -a

Protocole RARP

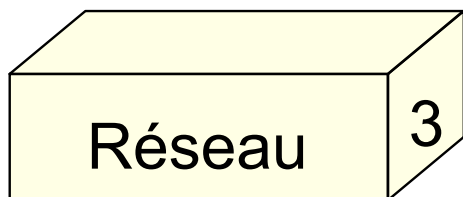
- Sert à résoudre le problème inverse d'ARP : obtenir l'adresse logique à partir de l'adresse physique.
- RARP : Reverse ARP
- Même trame employée. Seuls changent :
 - Le type de trame Ethernet : 0x0835
 - Les numéros d'opération : 3 requête, 4 réponse
- Même cache que le cache ARP.



- C'est un des rôles de la couche 3 : acheminer les informations d'un réseau à un autre,
- Les réseaux sont reliés entre eux à l'aide de **routeurs**,
- Tous les réseaux ne sont pas directement reliés, il faut passer par des réseaux intermédiaires.



- Toutes les machines (y compris les routeurs) possèdent une table de routage,
- Une table de routage contient des routes,
- Une route contient les paramètres pour déterminer par quel routeur ou passerelle passer pour accéder à un réseau donné,



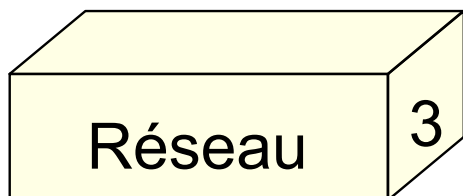
Voir la table de routage
(en mode administrateur)

Unix	:	route [-n]
Windows	:	route print

- Exemple de table de routage :

Table de routage IP du noyau

Destination	Passerelle	Genmask	Indic	Metric	Ref	Use	Iface
192.168.42.0	195.221.158.121	255.255.255.0	U	0	0	0	eth0
195.221.158.0	*	255.255.255.0	U	0	0	0	eth0
127.0.0.0	*	255.0.0.0	U	0	0	0	lo
default	192.168.42.1	0.0.0.0	UG	0	0	0	eth0



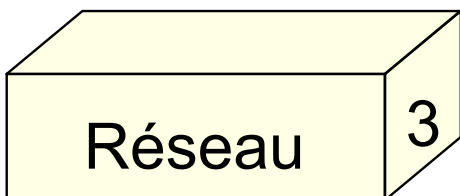
- Exemple de table de routage :

Table de routage IP du noyau

Destination	Passerelle	Genmask	Indic	Metric	Ref	Use	Iface
192.168.42.0	195.221.158.121	255.255.255.0	U	0	0	0	eth0
195.221.158.0	*	255.255.255.0	U	0	0	0	eth0
127.0.0.0	*	255.0.0.0	U	0	0	0	lo
default	192.168.42.1	0.0.0.0	UG	0	0	0	eth0

Réseaux ou machines à joindre.

“default” : route par défaut si aucune des autres ne marche.



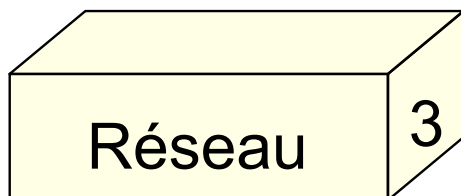
- Exemple de table de routage :

Table de routage IP du noyau

Destination	Passerelle	Genmask	Indic	Metric	Ref	Use	Iface
192.168.42.0	195.221.158.121	255.255.255.0	U	0	0	0	eth0
195.221.158.0	*	255.255.255.0	U	0	0	0	eth0
127.0.0.0	*	255.0.0.0	U	0	0	0	lo
default	192.168.42.1	0.0.0.0	UG	0	0	0	eth0

Routeur ou machine à contacter pour joindre le réseau de destination.

Si c'est '*' alors la machine est sur le même réseau que celui de l'interface (derrière colonne).

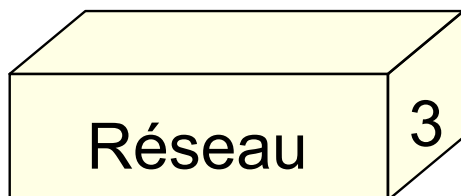


- Exemple de table de routage :

Table de routage IP du noyau

Destination	Passerelle	Genmask	Indic	Metric	Ref	Use	Iface
192.168.42.0	195.221.158.121	255.255.255.0	U	0	0	0	eth0
195.221.158.0	*	255.255.255.0	U	0	0	0	eth0
127.0.0.0	*	255.0.0.0	U	0	0	0	lo
default	192.168.42.1	0.0.0.0	UG	0	0	0	eth0

Masque de sous-réseau à utiliser conjointement avec le réseau de la 1ère colonne.



- Exemple de table de routage :

Table de routage IP du noyau

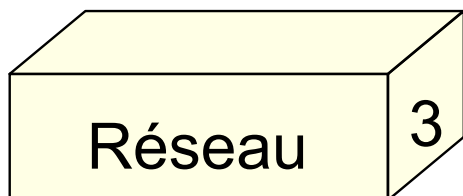
Destination	Passerelle	Genmask	Indic	Metric	Ref	Use	Iface
192.168.42.0	195.221.158.121	255.255.255.0	U	0	0	0	eth0
195.221.158.0	*	255.255.255.0	U	0	0	0	eth0
127.0.0.0	*	255.0.0.0	U	0	0	0	lo
default	192.168.42.1	0.0.0.0	UG	0	0	0	eth0

Interface réseau à utiliser pour communiquer.

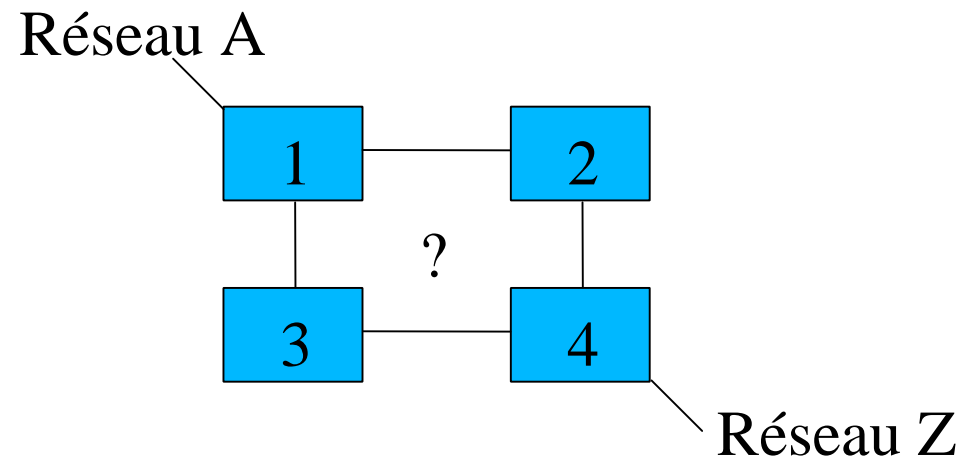
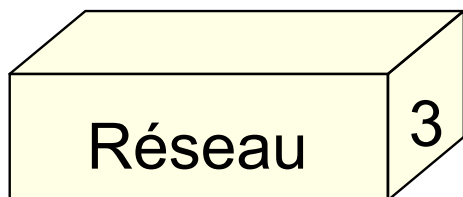
ethX : réseau ethernet

lo : loopback (interface locale)

etc ...

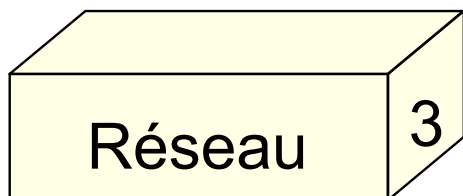


- Pour un réseau local :
 - Utilisation des routes configurées,
 - Utilisation de la route par défaut,
- Pour un réseau global :
 - Quel chemin prendre entre deux machines ?
 - Peut-on déterminer le chemin le plus court ?
- Protocoles : RIP, OSPF, ...

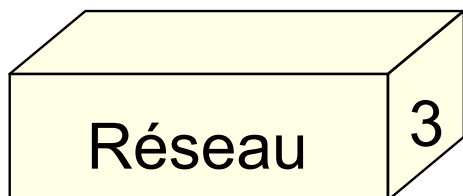


Protocole RIP

- RIP : Routing Information Protocol (RFC 1058)
- Extension de RIP : RIP2 en 1994 (RFC 1723)
- C'est l'un des protocoles de routage dynamique parmi les plus répandus malgré son âge.
- Chaque routeur échange :
 - les identificateurs des réseaux qu'il peut atteindre,
 - La distance qui le sépare de ces réseaux.
- Chaque routeur peut ainsi proposer le meilleur chemin.



- Mises à jour:
 - à des intervalles réguliers,
 - quand la topologie du réseau change,
 - Consiste en des échanges de vecteur-distance.
- Vecteur-distance (VD, 1 par route) composé de :
 - **Destination** : le vecteur destination,
 - **Coût** : le nombre de sauts à la destination (métrique),
 - **Source** : l'identifiant du routeur source.



- Chaque routeur ne connaît que son réseau direct,
- Le coût est de 0 pour chaque réseau direct,
- Exemple 3 routeurs s'échangent des informations:
 - étape 0 :

RTA		
Dest	Coût	Src
a	0	-

RTB		
Dest	Coût	Src
b	0	-

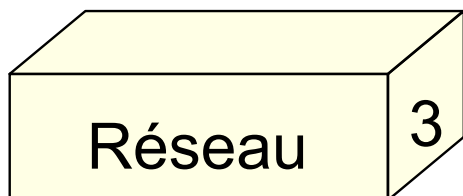
RTC		
Dest	Coût	Src

- étape 1: 1er échange:

RTA		
Dest	Coût	Src
a	0	-
b	1	rtb

RTB		
Dest	Coût	Src
b	0	-
a	1	rta

RTC		
Dest	Coût	Src
b	1	rtb
a	1	rta



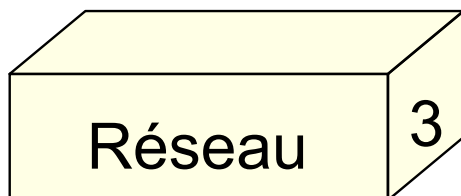
– étape 2: 2ème échange :

RTA		
Dest	Coût	Src
a	0	-
b	1	rtb
b	1	rtb
a	2	rta
b	2	rtc
a	2	rtc

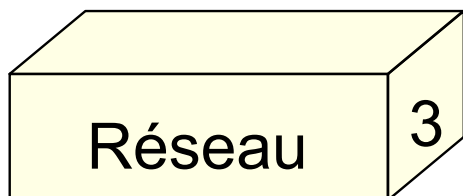
RTB		
Dest	Coût	Src
b	0	-
a	1	rta
a	1	rta
b	2	rta
b	2	rtc
a	2	rtc

RTC		
Dest	Coût	Src
b	1	rtb
a	1	rta
b	1	rtb
a	2	rtb
a	1	rta
b	2	rta

- Problème : il faut éliminer les routes redondantes pour trouver le meilleur chemin.



- Suppression des routes redondantes :
 - tout nouveau VD est comparé à la table courante :
 - une destination nouvelle est automatiquement ajoutée,
 - si la destination existe, elle est remplacée si :
 - la source est la même ,
 - la source est différente mais le coût est meilleur.
 - Au final on obtient, à la fin de l'étape 2 :



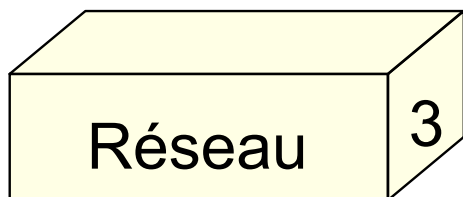
RTA		
Dest	Coût	Src
a	0	-
b	1	rtb

RTB		
Dest	Coût	Src
b	0	-
a	1	rta

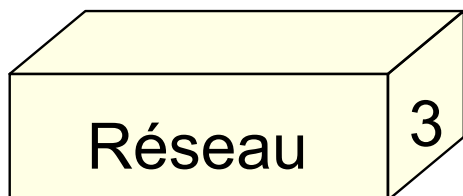
RTC		
Dest	Coût	Src
b	1	rtb
a	1	rta

- RIP améliore les VD en introduisant :
 - Le concept d'infinité,
 - Les stratégies '*Split Horizon*' et '*Poison reverse*',
 - *Une gestion temporelle.*

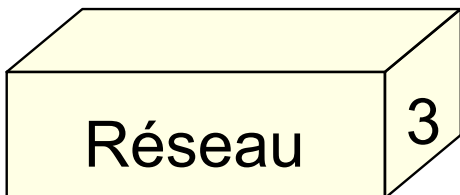
- *Le concept d'infinité :*
 - *Un réseau inatteignable a un nombre de sauts infini,*
 - *Dans la pratique l'infinité est réduite à 16 sauts.*



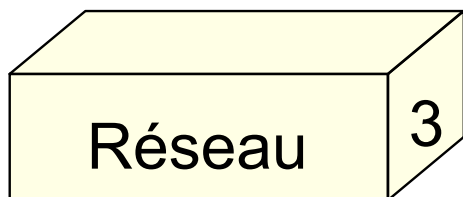
- Stratégie '*Split horizon*'
 - => *Un routeur ne renvoie pas à un autre routeur les VD qu'il a reçu de ce dernier,*
 - => **Implémentation obligatoire.**
- Stratégie '*Split horizon with Poisoned Reverse*'
 - => *Un routeur renvoie à un autre routeur les VD qu'il a reçu en leur donnant un coût de 16 (infini).*
 - => **Implémentation recommandée.**



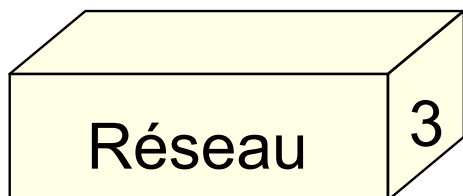
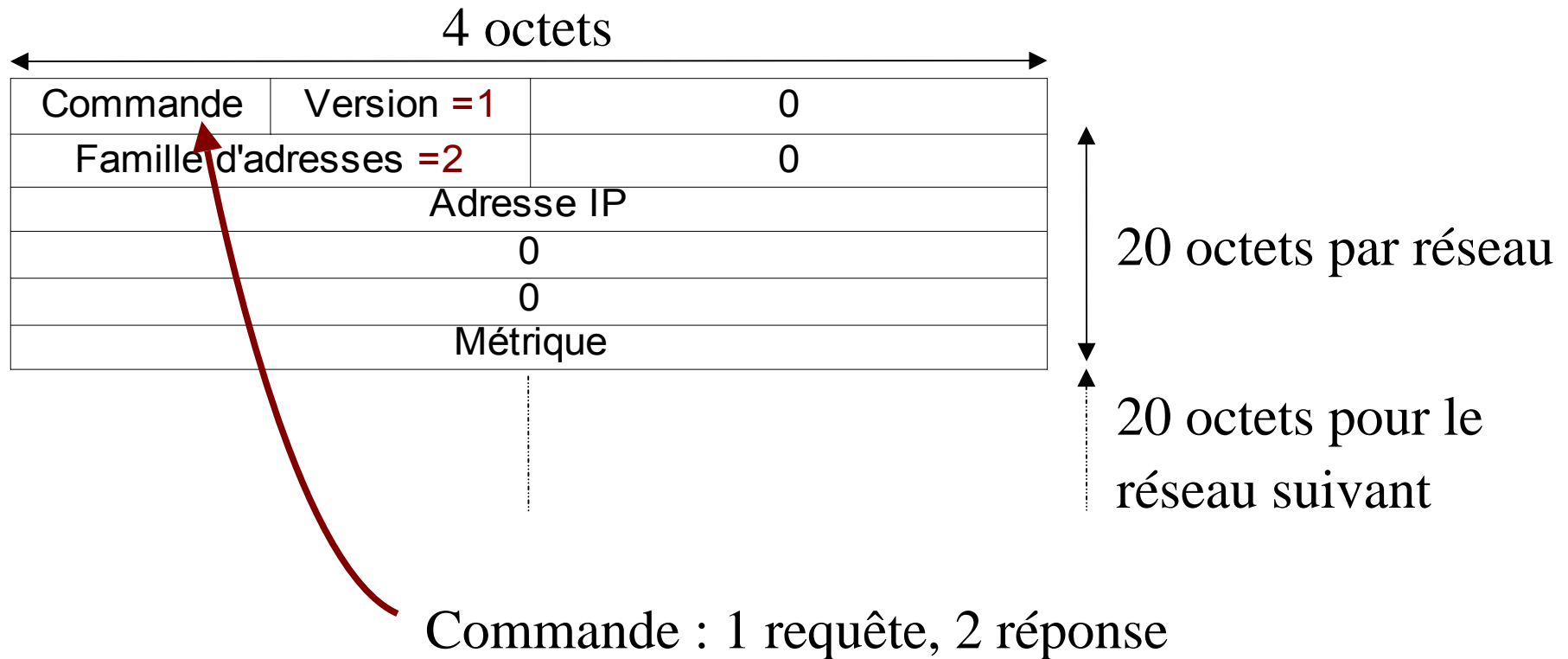
- Gestion temporelle : deux manières :
 - façon RFC :
 - temps entre mises à jour (maj): 30 s + petit délai aléatoire,
 - délai d'expiration : si 180 s s'écoulent après la dernière maj, marquer la route pour effaçage.
 - destruction : une route marquée est effacée 120 s après.
 - façon CISCO :
 - temps entre maj : 30s + petit délai aléatoire,
 - invalidité (= délai d'expiration RFC) : 180 s,
 - rétention : pour une route invalide, maj refusées pdt 180 s
 - destruction : une route invalide est effacée 240 s après.



- RIP envoie ses paquets par **UDP** sur le port 520,
 - RIPv1 utilise le broadcast, RIPv2 le multicast,
 - RIPv2 = RIPv1 + masques de sous-réseaux + mécanisme simple d'authentification
 - Avantages :
 - facile à implémenter, consomme peu de bp sur les petits réseaux
 - Inconvénients :
 - utilisation de UDP, infinité limitée à 15, gourmand en bp sur les grands réseaux (surtout si la topologie change).

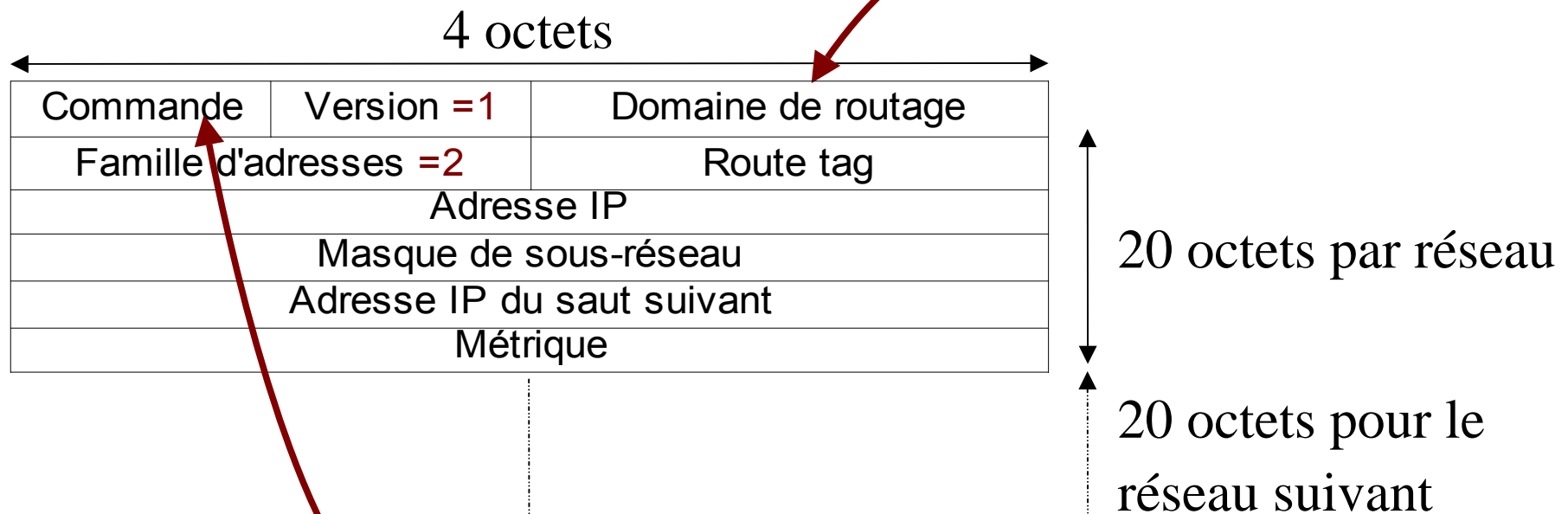


Datagramme RIPv1

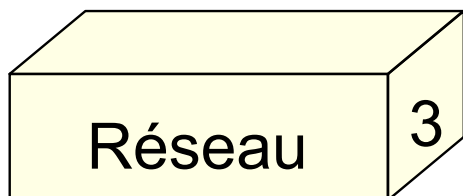


Datagramme RIPv2

Dans le cas où il y a plusieurs process gérant RIP sur la même machine

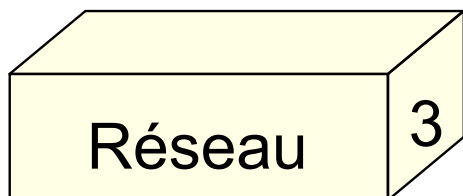


Commande : 1 requête, 2 réponse



Protocole ICMP

- Internet Control Message Protocol, RFC 792,
- Gère les erreurs relatives au protocole IP,
- Peut être employé par la machine ou le routeur à la source du problème.
- Protocole encapsulé dans un datagramme IP
 - champ “type de service” : 0,
 - champ “protocole” : 1

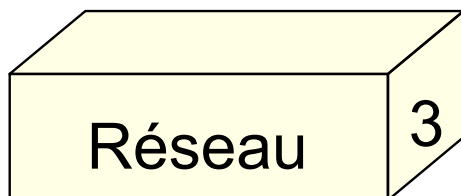


- Composé de 4 blocs :

Type (8 bits)	Code (8 bits)	CRC (16 bits)	Message (longueur variable)
------------------	------------------	------------------	--------------------------------

- Types :

Type	Contenu message	Type	Contenu message
0	Réponse echo (ping)	12	Erreur de paramètre
3	Destination non accessible	13	Demande horodatage
4	Contrôle de flux	14	Réponse horodatage
5	Redirection	15	Demande d'information
8	Echo (ping)	16	Réponse à 15
9	Avertissement Routeur	17	Demande de masque d'adresse
10	Sollicitation routeur	18	Réponse à 17
11	Durée de vie écoulée		



Exemple : Ping

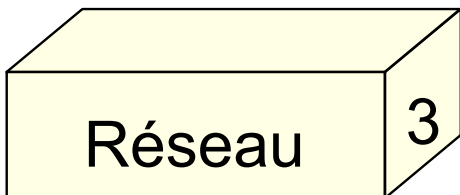
- Ping est un petit utilitaire permettant de contrôler si une machine est joignable sur un réseau,
- Format du message :

8/0 (8 bits)	0/3 (8 bits)	CRC (16 bits)	Application ID 16 bits	Numéro séquence 16 bits
-----------------	-----------------	------------------	---------------------------	----------------------------

- Sortie de la commande 'ping' :

```

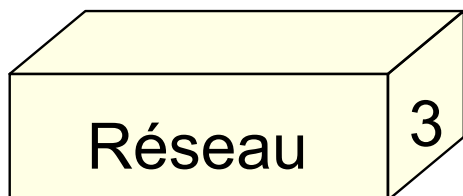
PING 192.168.42.22 (192.168.42.22) 56(84) bytes of data.
64 bytes from 192.168.42.22: icmp_seq=1 ttl=64 time=0.307 ms
64 bytes from 192.168.42.22: icmp_seq=2 ttl=64 time=0.304 ms
64 bytes from 192.168.42.22: icmp_seq=3 ttl=64 time=0.174 ms
64 bytes from 192.168.42.22: icmp_seq=4 ttl=64 time=0.169 ms
64 bytes from 192.168.42.22: icmp_seq=5 ttl=64 time=0.132 ms
  
```



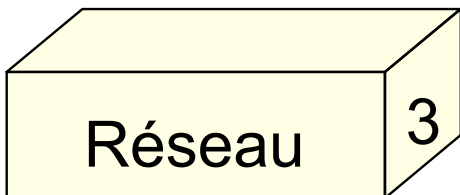
```

--- 192.168.42.22 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3996ms
rtt min/avg/max/mdev = 0.132/0.217/0.307/0.074 ms
  
```

- Les protocoles IP de niveau 3 étudiés :
 - IP : adressage et fragmentation des paquets,
 - ARP: retrouve l'adresse physique à partir de l'adresse logique,
 - RARP : la conversion inverse,
 - **ICMP** : gestion d'erreurs,
 - **RIP** : routage des paquets.



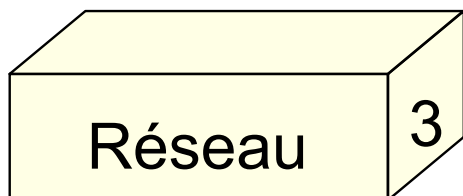
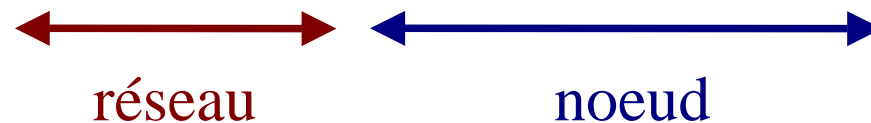
- IPX : Inter-network Packet eXchange,
- Développé par Novell au départ,
- Famille de protocoles plus simples que IP,
- Au niveau 3 : IPX et RIP (différent de celui d'IP)
 - Peut circuler sur 4 type de trames Ethernet ≠,
 - Adressage sur 80 bits, (autoconfigurable),
- Utilisé pour des réseaux de faible importance,
 - Protocole “bavard”,



- Adresse en deux parties :
 - Adresse de réseau : 32 bits
 - Attribué par l'administrateur / auto-attribué (aléatoire),
 - Adresse de noeud : 48 bits
 - Généralement l'adresse MAC,
 - Evite l'utilisation du protocole ARP.
- => Plusieurs réseaux logiques peuvent se partager une seule interface !

- Exemple d'adresse :

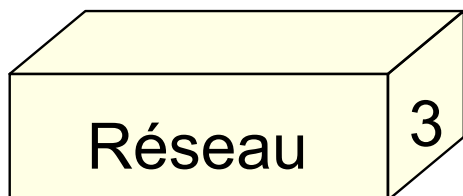
00000051:0000F3C4F69C



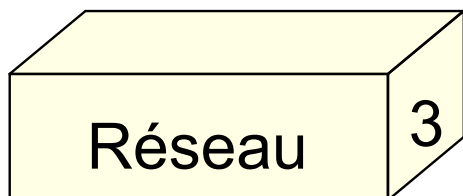
- En tête de 28 octets,
- Fragmentation des paquets non autorisée.

Champs	bits
CRC	16
Longueur du paquet	16
Contrôle de transport	8
Type de packet	8
Réseau de destination	32
Noeud de destination	48
Socket de destination	16
Réseau source	32
Noeud source	48
Socket source	16
Données	...

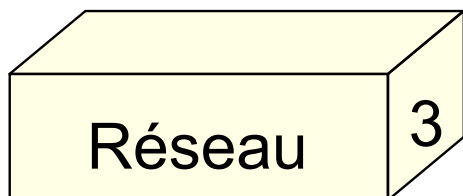
← Nombre de sauts: si > 16 destruction
 ← 5 : protocole SPX
 ← 17 : protocole NCP
 ← Socket: forme de multiplexage



- Protocole RIP de Novell :
 - basé sur des vecteurs de distance :
 - Tops d'horloge (mesure de débit),
 - Nombre de sauts,
 - peu adapté aux réseaux de grande taille,
- Table de routage différente de celle d'IP
 - une table par protocole IPX activé,
 - transmission des tables entre routeurs (chaque 60s),
 - stratégie '*split horizon*' + *classement chronologique*

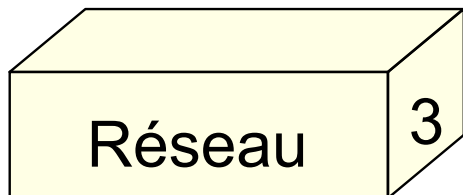


- NetBEUI est sur-couche de NetBIOS,
 - NetBIOS => Network Basic Input/Output System,
 - NetBEUI => NetBIOS Extended User Interface,
- Paradoxalement:
 - NetBEUI en couche 3,
 - NetBIOS en couche 4,
- NetBEUI formalise les trames NetBIOS,
 - 22 types de trames différentes.

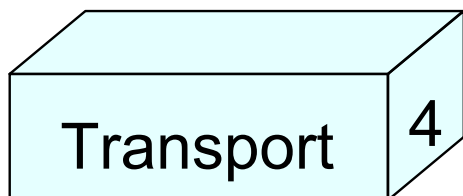


Protocole NetBEUI

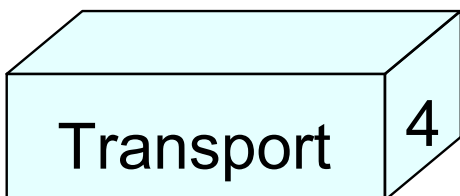
- Inventé par IBM, repris par Microsoft,
- Utilisé pour de petits réseaux,
- Réseau non routable,
- Chaque machine a un nom NetBIOS (max 16),
- NetBEUI utilise : nom NetBIOS + adresse MAC,
- Protocole très bavard :
 - utilisation massive de diffusions.



- Fonctions de la couche transport :
 - Division des messages longs en paquets,
 - Contrôle de la taille des paquets,
 - Regroupement des messages courts en 1 paquet,
 - Rassembler les paquets en 1 message,
 - Extraction et reconstitution du message d'origine,
 - Envoi et réception d'un accusé de réception,
 - Contrôle de flux et correction des erreurs de reconstitution.

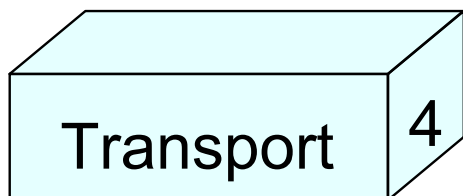


- Suite de protocoles IP :
 - TCP, Transmission Control Protocol,
 - UDP, User Datagram Protocol
- Suite de protocoles IPX :
 - SPX, NCP, SAP, (non détaillés dans ce cours)
- Protocole NetBIOS
 - (non détaillé dans ce cours)

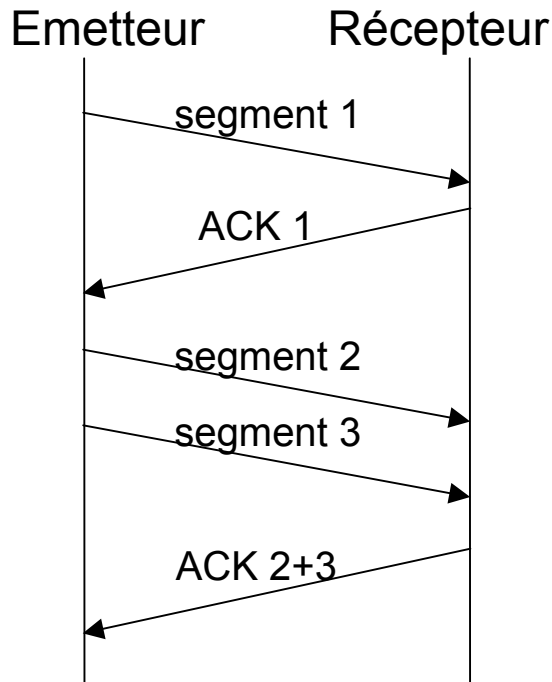


Protocole TCP

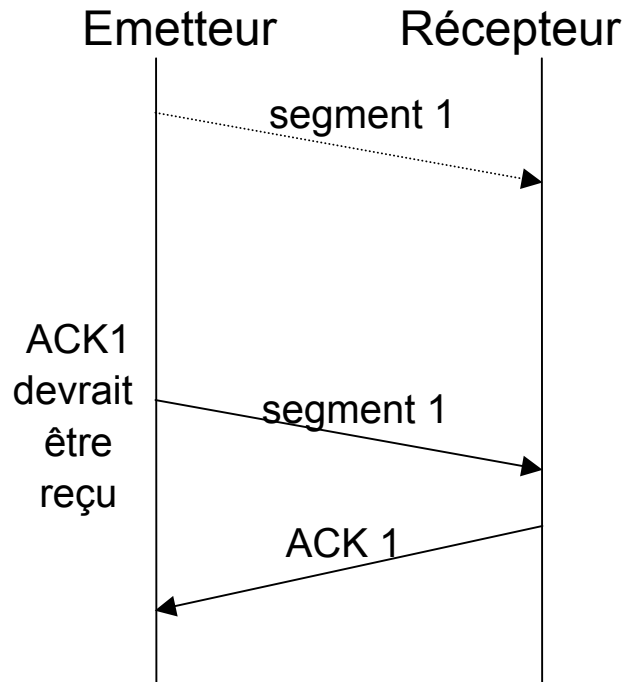
- TCP : Transfer Control Protocol – RFC 793,
- TCP fournit un service sécurisé de remise des paquets,
- TCP fournit un protocole fiable, orienté connexion encapsulé dans IP,
- TCP effectue des vérifications sur les paquets,
- TCP exige un accusé de réception des données,



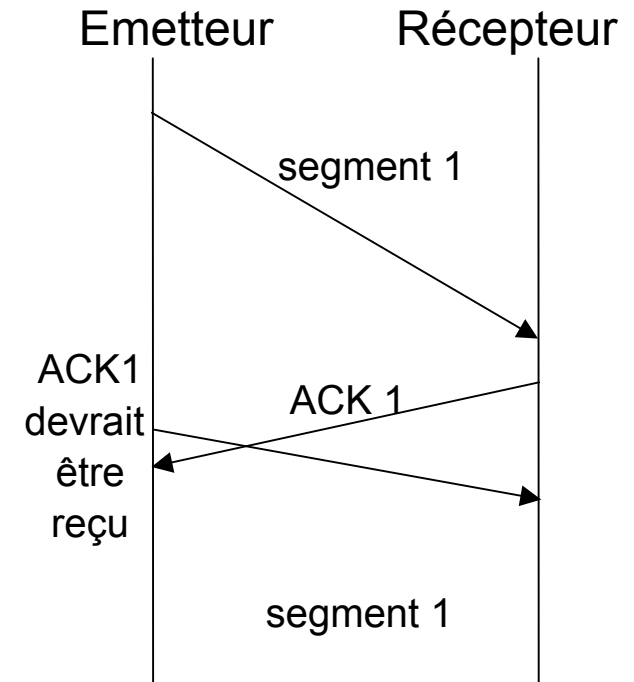
Echanges TCP



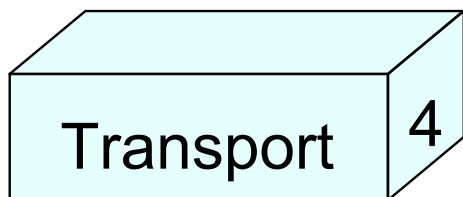
Transmission sans problème



Cas d'un paquet perdu



Cas d'un paquet dupliqué



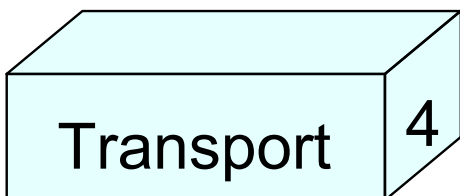
En-tête TCP

32 bits

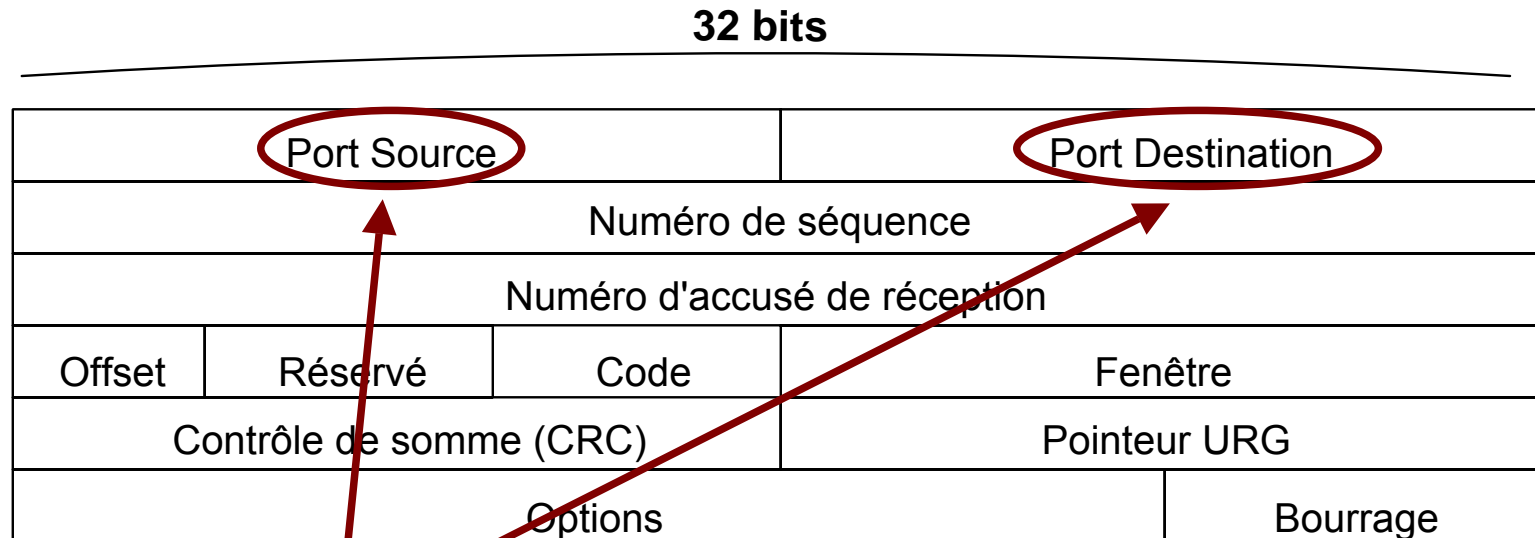
Port Source		Port Destination	
Numéro de séquence			
Numéro d'accusé de réception			
Offset	Réservé	Code	Fenêtre
Contrôle de somme (CRC)		Pointeur URG	
Options			Bourrage

Données

- En-tête :
 - 20 octets au minimum,
 - aligné sur 32 bits.



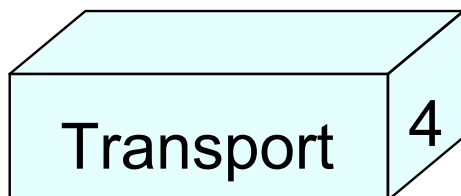
En-tête TCP



N°port :

permet une communication simultanée de plusieurs applications différentes entre 2 même machines.

ex : FTP (21), SSH (22), telnet (23), HTTP (80)



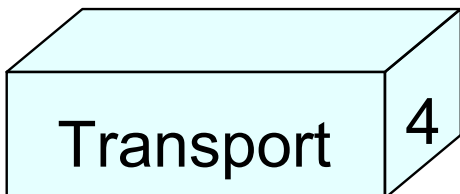
En-tête TCP

32 bits

Port Source			Port Destination		
Numéro de séquence					
Numéro d'accusé de réception					
Offset	Réservé	Code	Fenêtre		
Contrôle de somme (CRC)			Pointeur URG		
Options				Bourrage	

N° séquence :

position des données à transmettre par rapport au segment original. Au démarrage, le n° de segment est tiré aléatoirement.



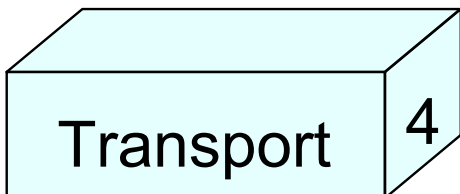
En-tête TCP

32 bits

Port Source		Port Destination	
Numéro de séquence			
Numéro d'accusé de réception			
Offset	Réservé	Code	Fenêtre
Contrôle de somme (CRC)		Pointeur URG	
Options			Bourrage

N°d'accusé de réception :

numéro qui identifie la position du dernier octet reçu (accompagné du drapeau ACK).

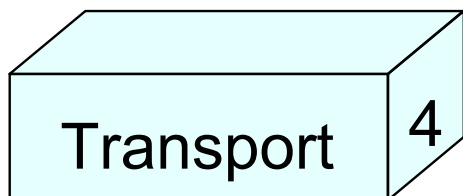


32 bits

Port Source		Port Destination	
Numéro de séquence			
Numéro d'accusé de réception			
Offset	Réservé	Code	Fenêtre
Contrôle de somme (CRC)		Pointeur URG	
Options			Bourrage

Offset :

- codé sur 4 bits,
- donne la taille de l'en-tête en mots,
- 5 en-tête normal,
- 6 ou + en tête avec options.



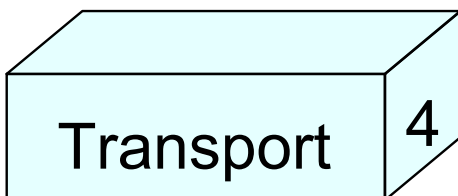
En-tête TCP

32 bits

Port Source		Port Destination	
Numéro de séquence			
Numéro d'accusé de réception			
Offset	Réservé	Code	Fenêtre
Contrôle de somme (CRC)		Pointeur URG	
Options			Bourrage

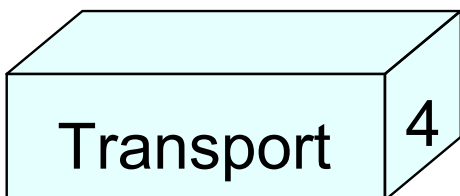
Code : (6 bits)

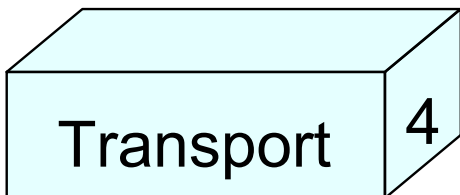
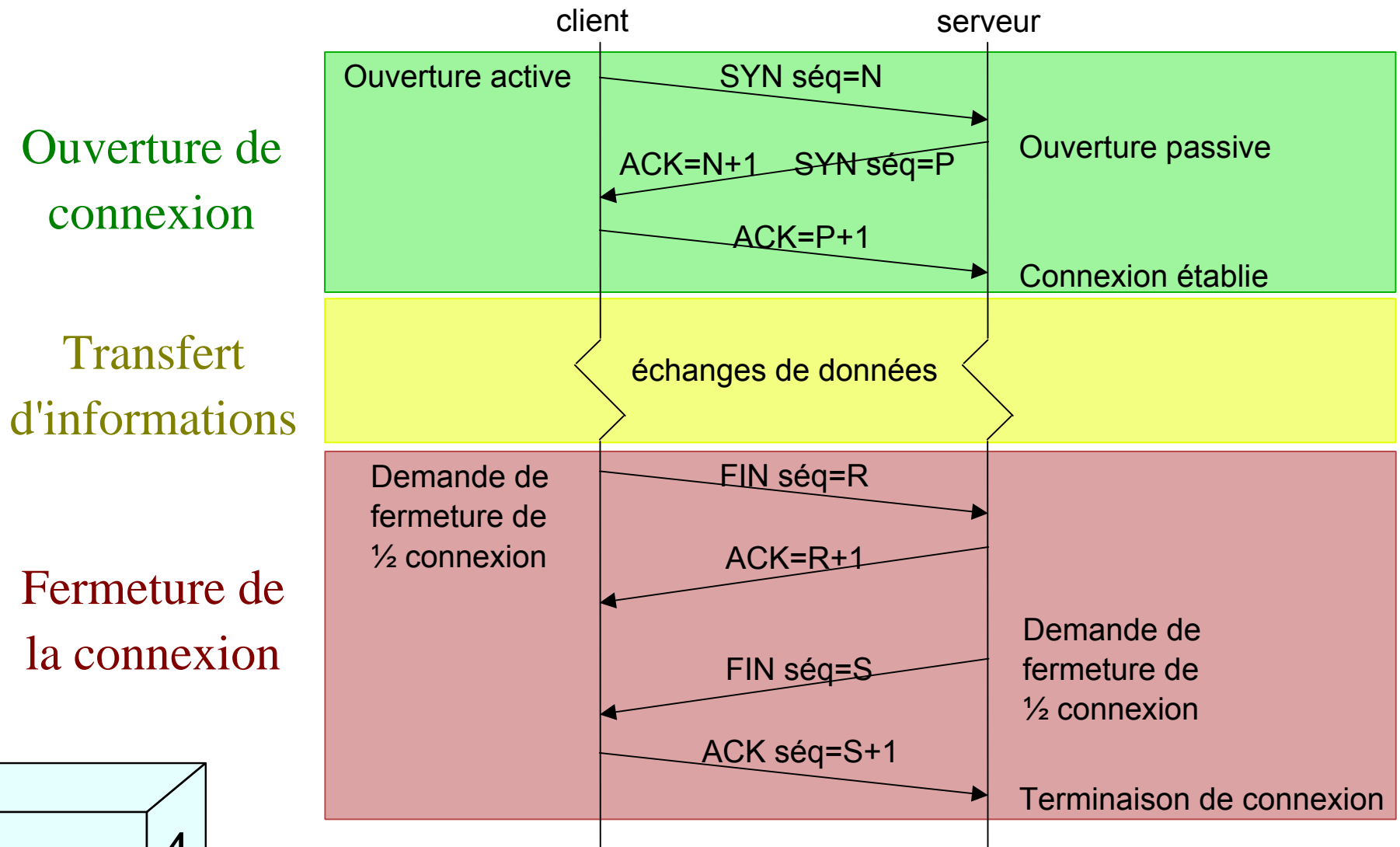
- influe sur le comportement de TCP
URG, ACK, PSH, RST, SYN, FIN



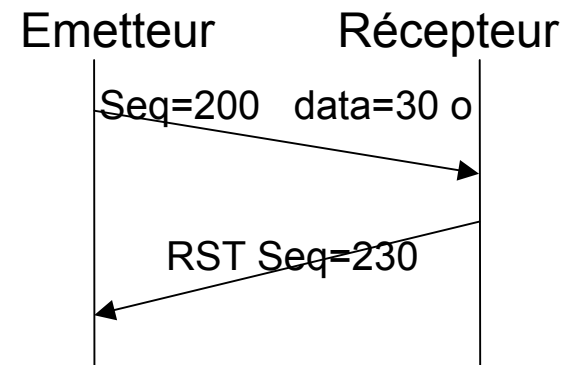
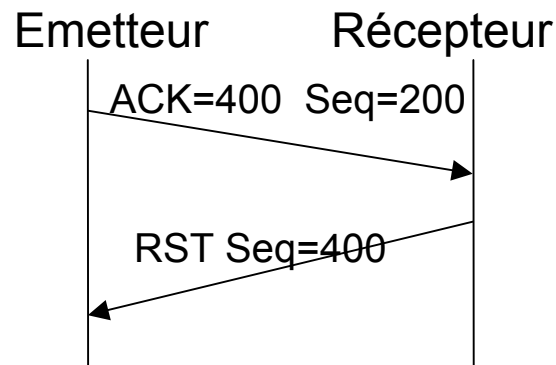
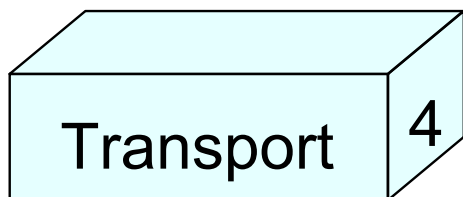
Code TCP

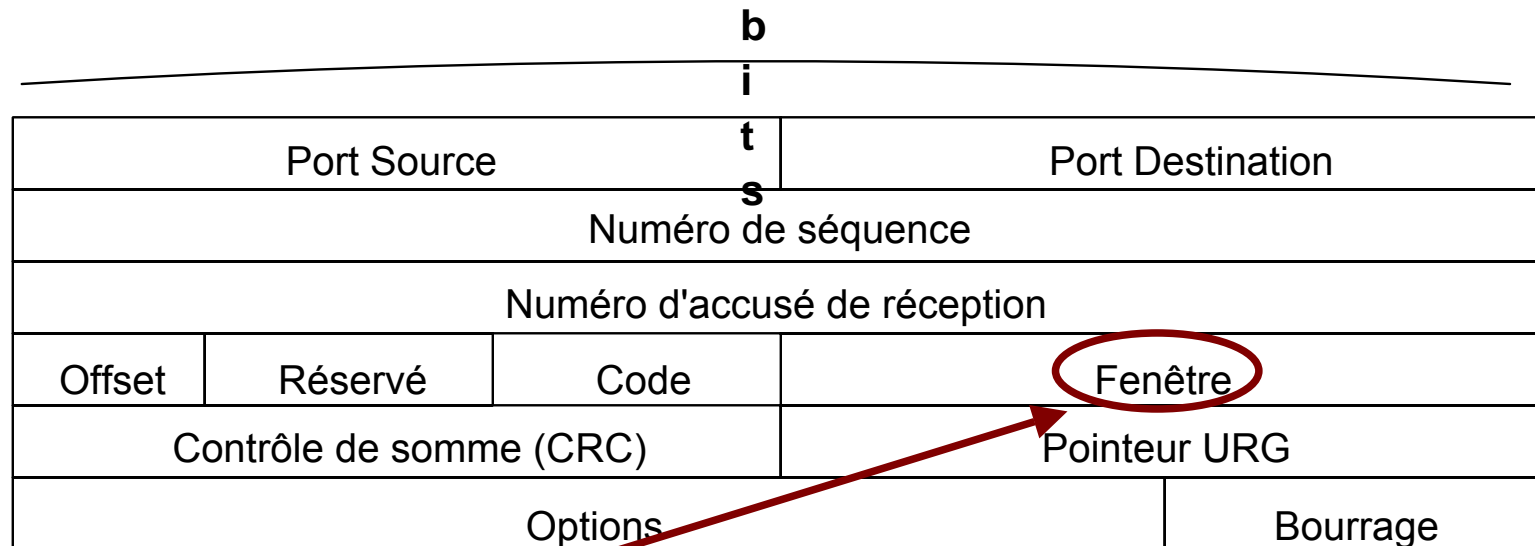
- URG : le champ “Pointeur URG” doit être exploité,
- ACK : le champ “accusé de réception” doit être exploité,
- PSH : toutes les données doivent être transmises à l'application sur le champ (PSH = PUSH),
- RST : réinitialisation de la connexion,
- SYN : le champ “N° de séquence” contient la valeur de début de connexion,
- FIN : l'émetteur du segment a fini d'émettre.





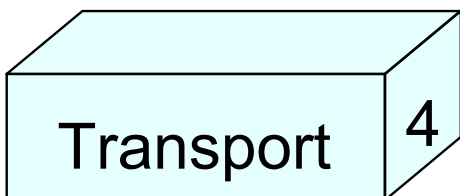
- Mécanisme employant le code RST,
- Sert à couper la connexion au plus vite,
- Type d'arrêt généré par le protocole TCP lui-même quand l'application s'est arrêtée de manière brutale.
- 2 cas possibles :





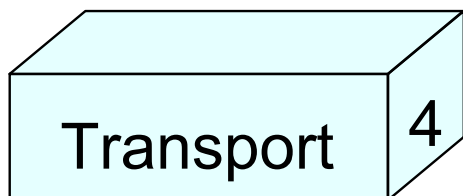
Fenêtre : (16 bits)

- nb d'octets à partir de la position marquée dans l'accusé de réception que le récepteur est capable de recevoir,
- les paquets après N° de séquence + fenêtre sont mis en attente.



Fenêtre glissante

- C'est un système de contrôle de bout en bout,
- Permet de réguler le trafic,
- La fenêtre peut-être de taille variable,
- Améliore l'état de la bande passante du système,
- Permet de ne renvoyer qu'un ACK pour plusieurs messages envoyés précédemment,
- Pour un flot de A vers B, c'est B qui régule la taille de la fenêtre.

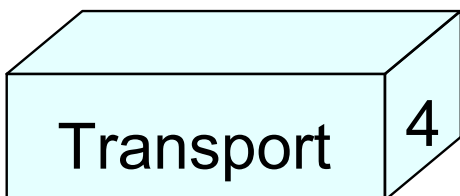


32 bits

Port Source		Port Destination	
Numéro de séquence			
Numéro d'accusé de réception			
Offset	Réservé	Code	Fenêtre
Contrôle de somme (CRC)		Pointeur URG	
Options			Bourrage

Pointeur URG : (16 bits)

- communique la position d'une donnée urgente en donnant son décalage par rapport au n° de séquence,
- dès que la donnée est reçue, elle doit être transférée à l'application.

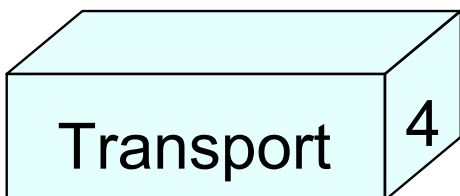


32 bits

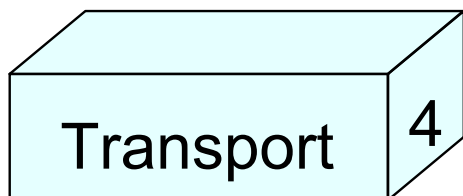
Port Source		Port Destination	
Numéro de séquence			
Numéro d'accusé de réception			
Offset	Réservé	Code	Fenêtre
Contrôle de somme (CRC)		Pointeur URG	
Options			Bourrage

Options : 2 formats :

- options mono-octet,
- octet de type d'option, octet de longueur d'option, octets de valeur d'option.

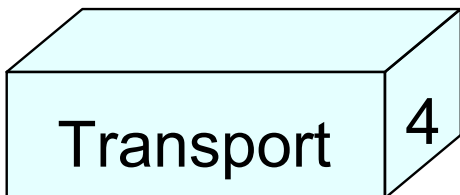


- **mss** : taille maximale du segment des données applicatives que l'émetteur accepte de recevoir, elle est envoyée lors de l'établissement de la connexion (Ethernet ~ 1460 octets),
- **timestamp** : pour calculer la durée d'aller-retour,
- **wscale** : Facteur d'échelle de la fenêtre "shift", dans ce cas la taille est *fenêtre x 2 x shift*,
- **nop** : *ne fait rien, sert au bourrage.*

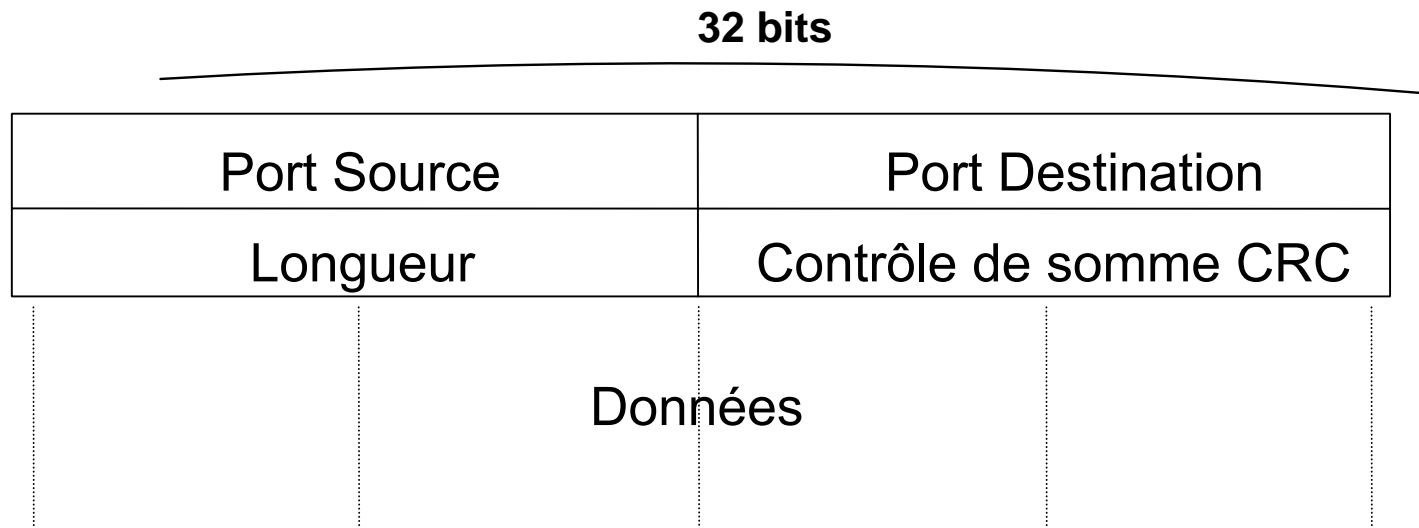


Protocole UDP

- UDP: User Datagram Protocol – RFC 768,
- UDP ne vérifie pas que le destinataire a reçu le message,
- UDP ne réordonne pas les paquets,
- UDP ne contrôle pas les flux,
- UDP est un mode de transport non connecté,
- UDP rajoute à IP la notion de ports applicatifs,



En-tête UDP

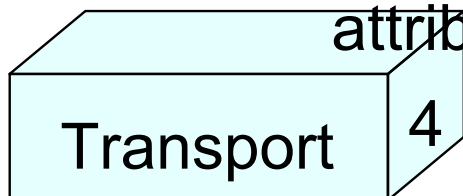


N°port :

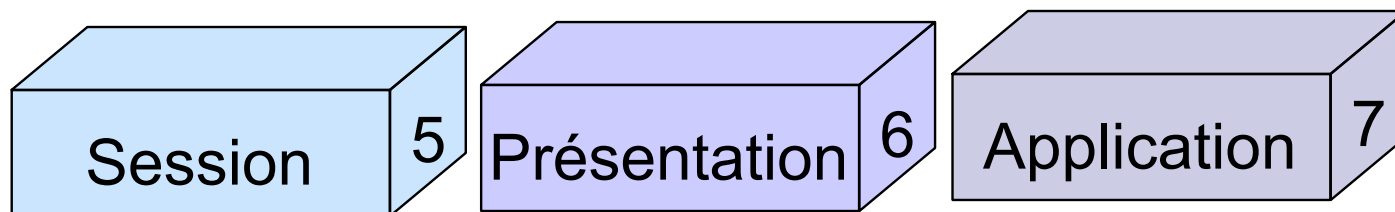
même fonctionnalité que TCP,

peut être partagé avec TCP pour le même type d'application,

attribué par l'IANA pour les protocoles très courants.



- La couche **session** gère la connexion entre deux ordinateurs du réseau,
 - La couche **presentation** gère le format des données échangées entre 2 machines,
 - La couche **application** joue le rôle d'une interface d'accès des applications au réseau.
- => Dans la pratique, ces 3 couches sont confondues !!!**
- => On peut les regrouper en une couche application.**



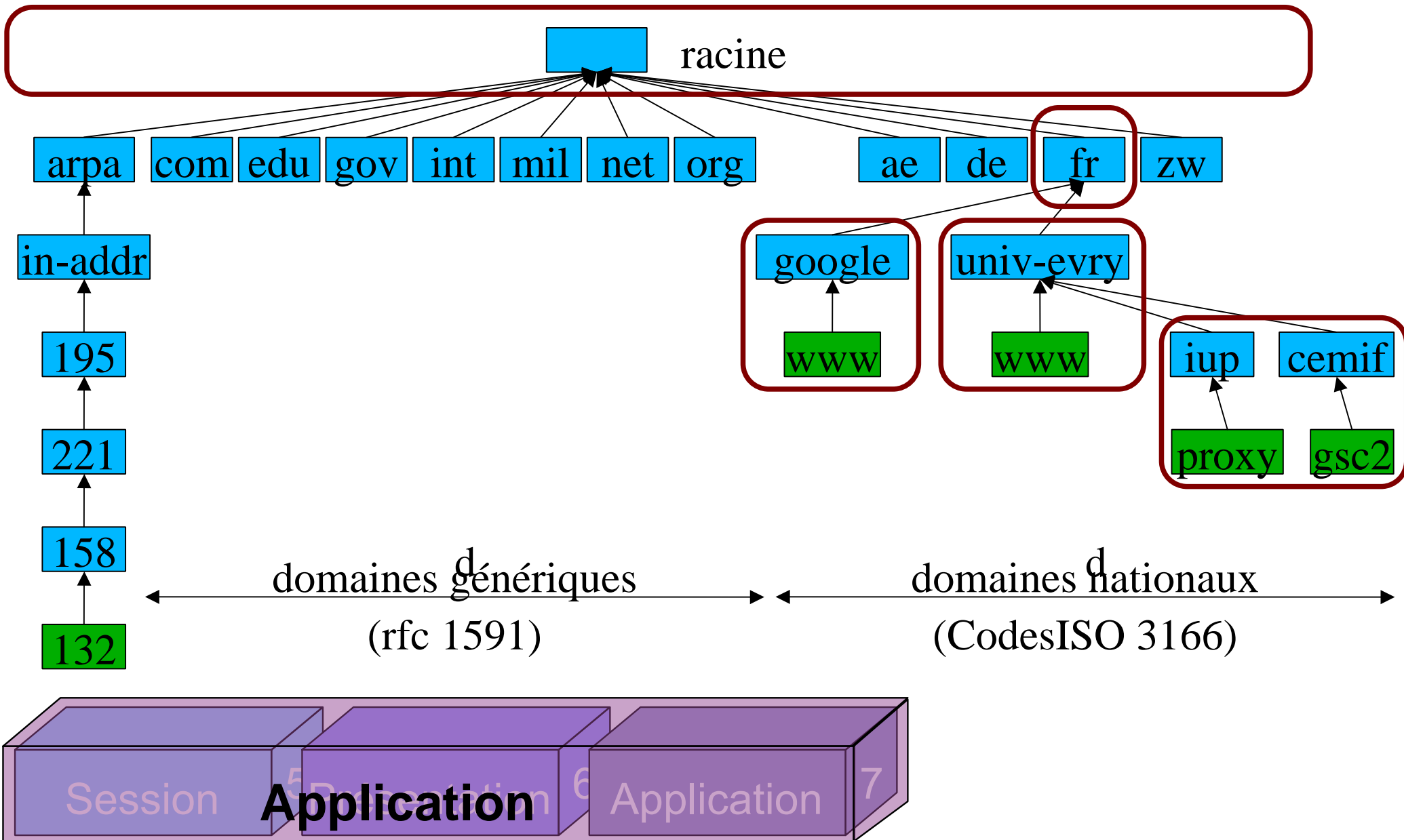
- Raison : les adresses IP sont difficiles à manipuler et à retenir,
- Les noms de domaines sont des noms plus parlants : ex : www.google.fr ,
- Ces noms sont aussi appelés les DNS (Domain Name System), 1987 - RFC 1034 et 1035,
- La base de données de ces noms est distribuée.



- Systeme hiérarchisé sous forme d'arbre,
- Chaque nom porte un nom, la racine n'en a pas,
- Les machines ou feuilles sont nommées à l'aide du chemin parcouru dans l'arbre,
- Le séparateur entre chaque noeud est le '.',
- Tout noeud est un domaine,
- Les serveurs de noms traitent des zones.



Exemple de hiérarchie



- Le '.' est le séparateur,
- 63 caractères max pour un noeud, souvent 12 max par habitude,
- majuscules et minuscules indifférenciées,
- les chiffres, '-' et '_' sont autorisés,
- les espaces, tabulations sont interdits,
- le nom complet fait 255 caractères max.



- Dans le cas des machines Unix :

- fichier : /etc/resolv.conf , exemple :

```
domain cemif.univ-evry.fr
search cemif.univ-evry.fr. , univ-evry.fr.
nameserver 195.221.158.231
nameserver 194.199.90.1
```

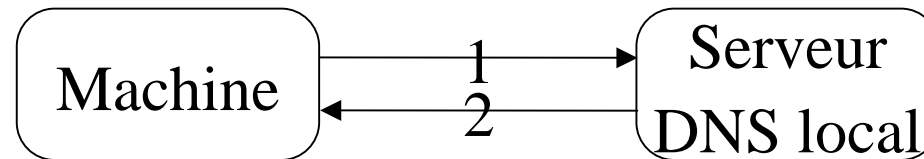
- 'domain', domaine local,

- 'search', suffixe à mettre à un nom de machine,

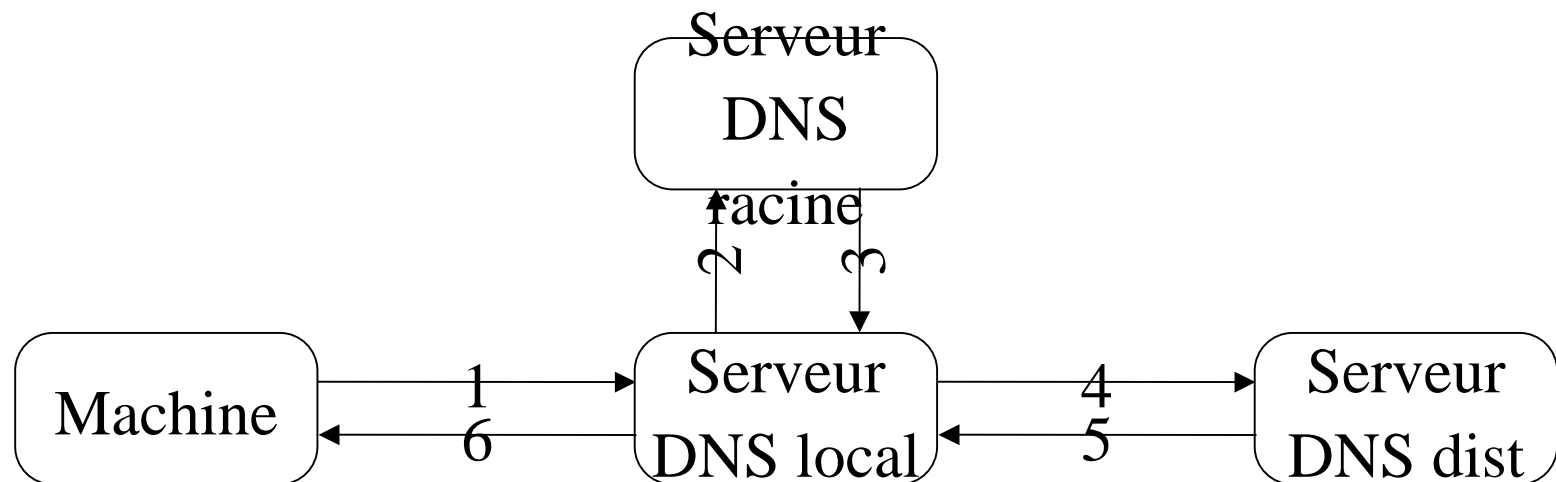
- 'nameserver', adresse de serveur DNS,



- Interrogation locale :



- Interrogation distante (mode récursif):



- Transport :
 - UDP (port 53), 520 octets max, adapté pour des requêtes standards, pas des transferts de zone à zone,
 - TCP (port 53), le datagramme inclus alors une donnée de type 'longueur'.
- Un format pour les datagrammes des requêtes,
- Un format pour les données dans la base.



- En-tête de 12 octets :

Id	Flags1	Flags2	QDcount	ANcount	NScount	ARcount
16 b	8 b	8 b	16 b	16 b	16 b	16 b

- Il est suivi des enregistrements de ressource (RR) qui sont de taille variable :

Nom	Type	Classe	TTL	Longueur	Données
variable	16 b	16 b	32 b	16 b	variable

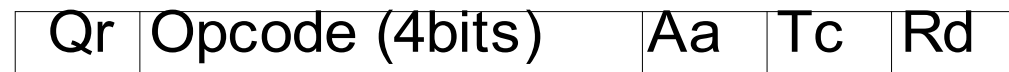


En-tête DNS



Identifiant :

- requête : aléatoire,
- réponse : dupliqué



demande de
récursivité

message tronqué ?

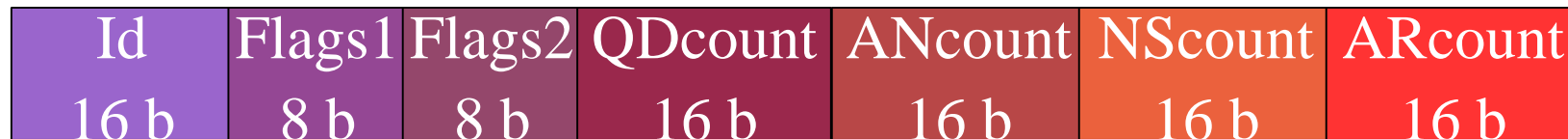
réponse d'une
entité autoritaire

0: requête
1: réponse

0: requête standard
1: requête inverse
2: statut requête serveur
3-15 : réservé



En-tête DNS



récurtivité
autorisée

réservé (0)

type de réponse :

0: pas d'erreur

2: pb serveur

4: non implémenté

6-15: réservé

1: erreur de format requête

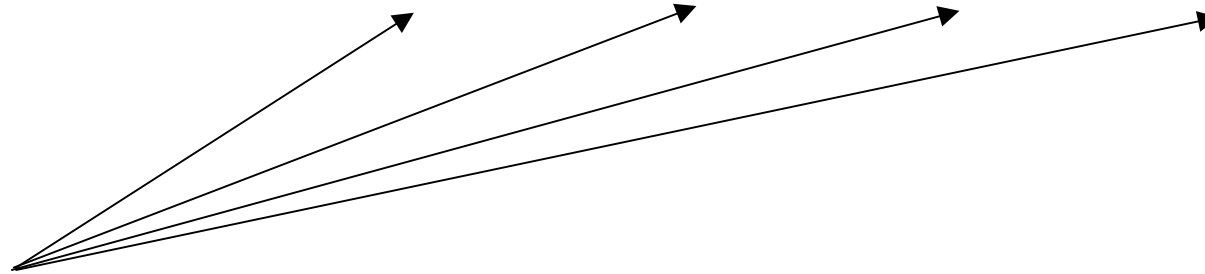
3: nom inexistant

5: refus



En-tête DNS

Id	Flags1	Flags2	QDcount	ANcount	NScount	ARcount
16 b	8 b	8 b	16 b	16 b	16 b	16 b



nb d'entrées dans les sections :

- question,
- réponse,
- autorité,
- additionnel.

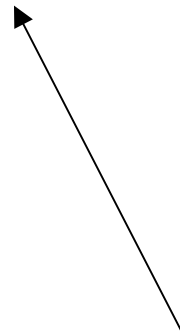
Requête de nom ou d'adresse
IP à partir d'un nom :

Unix : nslookup ou host
Windows : nslookup



Nom	Type	Classe	TTL	Longueur	Données
variable	16 b	16 b	32 b	16 b	variable

Entrée	Valeur	Désignation
A	1	adresse de l'hôte
NS	2	nom du serveur DNS
CNAME	5	nom canonique
SOA	6	information de zone
WKS	11	services internet
PTR	12	résolution inverse
HINFO	13	description machine
MINFO	14	groupe de boîte à lettre
MX	15	serveur de messagerie
TXT	16	chaîne de caractère



Nom	Type	Classe	TTL	Longueur	Données
variable	16 b	16 b	32 b	16 b	variable

01: Internet (In),
04: Hesiod (Hs)

Durée de vie de l'entrée
en secondes

Longueur en octets
des données suivantes

Type	Données
A	adresse IP (In)
NS	nom d'hôte
CNAME	nom de domaine
SOA	plusieurs champs
PTR	adresse IP sous forme de nom



- FTP : File Transfer Protocol – RFC 959, 1985,
 - port 21 pour les commandes,
 - protocole de transfert de fichiers entre deux machines,
 - sous sa forme la plus simple, envoie des commandes à la machine distante après s'être identifié:
 - cd : pour changer de répertoire,
 - ls : pour voir le contenu d'un répertoire,
 - get : pour télécharger un fichier,
 - put : pour déposer un fichier,
 - mget/mput : même chose que get/put avec des '*'



- Telnet - RFC 854, 1983, port 23,
 - permet de se connecter à une machine distante,
 - après authentification, on peut faire exécuter à la machine distante diverses commandes,
 => inconvénient : tout passe 'en clair' sur le réseau.
- SSH : Secure Shell, 1995, port 22,
 - reprend les fonctionnalités de Telnet et FTP,
 - communications cryptées.



- SMTP: Simple Mail Transfer Protocol
 - RFC 821, 1982, port 25,
 - Protocole de transfert d'e-mail (ne permet pas de les télécharger),
 - Possibilité d'avoir des relais.
- POP3 : Post Office Protocol
 - RFC 1939, 1994, port 110,
 - Protocole permettant de télécharger les mails.



- HTTP : HyperText Transfer Protocol
 - HTTP/1.0, RFC 1945, 1996,
 - HTTP/1.1, RFC 2616, 1999,
 - Un des protocoles les plus répandus,
 - C'est celui des serveurs et des navigateurs WEB !!!
 - Permet d'échanger des informations multi-média à travers le monde.



- 2 grandes familles d'API similaires :
 - API Unix : implémenté dans le noyau,
 - Fichiers d'en-tête nécessaires :
 - #include <sys/types.h>
 - #include <sys/socket.h>
 - API Windows : implémenté dans la librairie Winsock,
 - Fichiers d'en-tête nécessaires :
 - #include <windows.h>
 - #include <winsock.h>
- Fera l'objet d'une séance de TP !!!



- Englobe les couches 5,6 et 7 du modèle OSI,
- Dans celle-ci on retrouve, entre-autres,
 - la gestion des noms de machines (DNS),
 - divers autres protocoles :
 - ftp, telnet, ssh, pop3, smtp, http, etc ...



Conclusion

- En résumé, que se passe t'il si je fais une recherche sur le site : <http://www.google.fr/> ?
- Ceci ne constitue qu'une introduction sommaire aux réseaux, de nombreux domaines n'ont pas été explorés.

Des Questions ?