

**Introduction à la
problématique des
Réseaux avec QoS
- Liaison d'Acheminement et
Internet Classique vis-à-vis de
la QoS -**

Ingénierie des Réseaux d'Entreprise (Cycle C),
Compléments Réseaux de Transport et Application (Cycle B)

Janvier 2002

Eric Gressier-Soudan

Liaisons d'Acheminement et QOS

Protocoles de Liaison - LAN

Token Ring, FDDI, Ethernet 10M-100M-1G
possibilité de Multicast/Broadcast natif

- Token ring : pb temps d'attente du jeton non borné (si panne du moniteur), il faut utiliser le mécanisme de priorité mais pas d'équité, débit faible (16Mb/s)
- FDDI : 100Mb/s, pb du temps d'attente du jeton mais borné par le TTRT, apte à transmettre du multimédia pourvu que le réseau ne soit pas trop long, ni trop de stations (gigue de traversée) ... réseau d'artère qui tombe en désuétude, sinon version FDDI-2 pour écouler du trafic isochrone ... mais pas un succès commercial
- Ethernet : 100Mb/s et 1Gb/s, (10Gb/s maintenant en MAM), non déterministe, toutefois avec 802.1Q/p, possibilité de rendre certains trafics prioritaires dans la traversée des commutateurs (switch)! Pas de garantie sur les délais de transfert. Logique de type "best effort".
Attention, Ethernet commuté Full-Duplex évite les collisions, permet l'émission et la réception simultanée mais le commutateur introduit de la gigue.
Attention aux effets du Spanning-Tree (profondeur) sur un réseau de switch.

802.1Q/p

Norme associée aux VLAN : extension de la trame Ethernet : passe de 1518 o à 1522 o

4 octets ajoutés devant le champ type (VLAN tag):

3 bits 1 b 12 bits
(TR)

Tag Protocol Identifier (8100 pour Ethernet)	USER PRIORITY	CFI	VLAN ID
	TAG Control Information		

2 Octets

2 Octets

8 niveaux de priorité, qui permettent à un commutateur d'écouler un trafic prioritairement à un autre.

peut servir à mapper le champ priorité d'un datagramme IP (voir TOS)

attention à la gestion de la priorité d'un fournisseur de commutateur à l'autre

Protocoles de Liaison – Accès fournisseur

- **Modem** : 56Kb/s, asymétrique entre descente et remontée, faible débit, audio ... pas de QoS
- **ISDN** : 2*64Kb/s, voix numérisée, visioconférence vidéo en compressé H261. QoS sous la forme de débit garanti.
- **ADSL** : plus généralement xDSL, débit asymétrique garanti à un usager tel que : haut débit en descente, débit raisonnable en montée; en fonction de la nature des installations et de l'éloignement du DSLAM (Digital Subscriber Line Multiplexer), dans la pratique les débits peuvent être plus faibles que ceux annoncés. Voir avec RADSL (Rate Adaptative DSL) et VDSL (Very High DSL) des versions avec des garanties de débit par application.
- **Câble** : transmission numérique, la bande passante totale d'un nœud de réseau (qui dessert entre 500 et 1500 abonnés) est partagée entre tous les abonnés. Le standard PacketCable/DOCSIS (IP sur câble) permet la réservation de bande passante, vise à limiter le délai de transmission à 20ms, la couche d'accès supporte des fonctions de gestion de QoS.
- **Réseau Electrique** : technologie Digital Power Line, développée initialement en GB puis supportée par Nortel offre un débit de 1Mb/s.

Protocoles de Liaison – Réseaux Air

- **LAN sans fil:** LMDS/MMDS, DECT, Wireless Ethernet (famille 802.11), Bluetooth, Apple Airport, approche prometteuse, mais les aspects QoS sont encore à approfondir.

- **Mobile sans fil :** GSM très faible débit, GPRS 100 à 384 Kb/s, UMTS (2Mb/s terminaux faiblement mobiles) seul support envisagé pour le multimedia mobile personnel, seul support pour lequel on parle de QoS.

- **Satellites :**
 - Géostationnaire (36 000 km): 2Mb/s en point à point jusqu'à 24Mb/s, grands délais de transfert (latence), compter 500 ms en A/R entre la terre et le transpondeur spatial.
 - Orbite basse (LEO – Low Earth Orbit - 1500km): 64 Mb/s en descente et 2Mb/s à la montée, nécessitent une communication entre les satellites.

- **Avion ou Ballon stratosphérique :** 22km, 64kb/s à 2Mb/s symétrique, solution plus flexible et moins onéreuse

Protocoles de Liaison - WAN

- **X25** : faible débit (64Kb/s à 2Mb/s), protocole de contrôle de flux et de contrôle d'erreur entre commutateurs (très fiable), pas de support explicite de la QoS
- **Frame Relay** : réseau physique de 1,5 à 45Mb/s sous-jacent (52Mb/s avec la proposition HSSI), commutateurs fonctionnent en mode "forwarding", pas de contrôle de flux ni d'erreur entre commutateurs (mieux que X25 pour le multimédia), notion de bande passante garantie (CIR, Committed Information Rate) et de débit maximum autorisé (EIR, Excess Information Rate), ébauche de gestion de ressources vis-à-vis de la QoS par le mécanisme d'indication de congestion (bits BECN et FECN¹) et le bit DE², possibilité d'utiliser le champ TOS d'un datagramme IP (délai, débit, fiabilité, coût) pour définir le bit DE
- **Ethernet 10Gb/s**

¹ BECN = Backward Explicit Congestion Notification, Forward Explicit congestion Notification

² DE = Discard Eligible

ATM

Réseau à Commutation de Cellules qui a pour objectif de multiplexer différents flots de données en un seul lien qui utilise une technologie de type TDM ou MRF (Multiplexage à Répartition dans le Temps) comme SONET, SDH, PDH.

Couche d'Adaptation -AAL	S-Couche de convergence
	S-Couche SAR
	Couche ATM
	Couche Physique

Réseau Haut Débit : 155Mb/s (lien OC3), 622Mb/s (OC12) voire 2,48Gb/s (OC48) sur une artère.

En fait la partie réservée aux données applicatives dépend de l'AAL.

Classes de Services et AALs

Les Classes de services considérées dépendent de l'intervalle de temps séparant les cellules, et de la tolérance à la gigue :

Classe	Description	Exemple
CBR	Débit constant garanti	Audio/vidéo non compressée
rt-VBR	Débit variable : trafic temps réel	Audio/vidéo compressée
nrt-VBR	Débit variable : trafic non temps réel	Transactionnel
ABR	Débit Disponible	Interconnexion de réseaux locaux
GFR ³	Trafic non temps réel équivalent au Frame Relay CIR	Trafic IP sans contrainte de synchronisation, ni garanties, ni délai, ni gigue
UBR	Débit non défini	Données info, support d'IP

Différentes AAL sont proposées :

AAL 1 : pour transmettre les données temps réel à débit constant suivant un mode connecté orienté flot de bits, pas de détection d'erreur mais indication d'erreur (doute émis sur sa nécessité)

AAL 2 : transmission de données temps réel à débit variable, avec préservation des frontières des messages, mode connecté (obsolète)

AAL 3/4 : mode flot ou message, supporte le multiplexage sur un VC (en cours d'abandon)

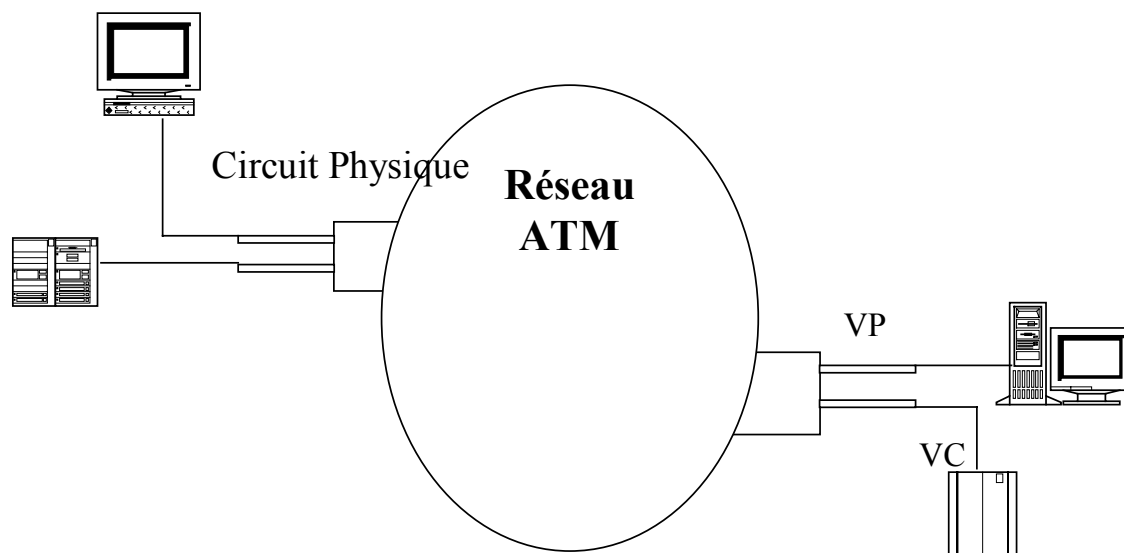
AAL 5 : supporte le point à point et le point à multipoints, connecté ou non, mode message ou mode flot, message de taille max en mode message de 64Ko, pas de contrôle d'erreur de bout en bout (seule AAL qui semble avoir de l'avenir)

AAL 0 : accès direct aux cellules ATM, on peut écrire sa propre AAL

³ GFR = Guaranteed Frame Rate, service non temps réel, développé récemment, MFS (Maximum Frame Size) et MCR sont spécifiés.

Réseau ATM en mode connecté

Voie virtuelle (VC) et Chemin virtuel (VP)



Cellule ATM : 5 octets d'entête + 48 octets de données soit 53 octets

La commutation opère à partir des identificateurs de VP et de VC : brassage de VP, et commutation de VC.

Fonctions associées à la gestion de trafic :

- Contrôle d'admission,
- Surveillance du trafic utilisateur,
- Contrôle de cellules (lié au bit CLP dans la cellule),
- Cadencement du trafic (GCRA – Generic Cell Rate Algorithm) ...

Modèle de Contrat - QoS ATM

Le réseau ne garantit les contraintes de QoS que si le trafic de la source (utilisateur) respecte le contrat :

Paramètres de Trafic :

- *Peak Cell Rate (PCR)*, cells/s (*débit max*)
- *Substainable Cell Rate (SCR)*, cells/s \leq PCR (*débit moyen*)
- *Maximum Burst Size (MBS)*, cells, (*nombre max de cellules envoyées au débit PCR*)
- *Minimum Cell Rate (MCR)*, cells/s, pour service ABR(*débit minimal garanti*).

Paramètres de QoS demandée :

- *maximum Cell Transfer Delay (maxCTD)*, sec (*délai max*)
- *peak-to-peak Cell Delay Variation (peak-to-peak CDV)*, sec (*gigue max*)
- *Cell Loss Ratio (CLR)*, cells (*taux de perte de cellules max*)

services ATM

Attributs	CBR	rt-VBR	Nrt-VBR	UBR	ABR
Paramètres de Trafic:					
PCR	√	√	√		√
SCR, MBS	n/a	√	√	n/a	n/a
MCR	n/a	n/a	n/a	n/a	√
Paramètres de QoS:					
PpCDV	√	√			
MaxCTD	√	√	√		
CLR	√	√	√	√	

√ = spécifié , n/a = non applicable

Réservation à l'initiative de l'émetteur.

Projection des caractéristiques de flux de messages applicatifs en paramètres de QoS

- Quand tout est constant :
 - Si I intervalle de temps entre deux requêtes successives (secondes), le débit en requêtes par unités de temps est $1/I$
 - Si S est la taille d'une requête (bits), le débit soumis est S/I (bits par seconde)

PCR = $S/(I * 48^4 * 8)$; $1/PCR$ = temps séparant l'arrivée de 2 cellules

- Quand les flux sont irréguliers :
 - Si I_{min} intervalle de temps min entre deux requêtes successives (secondes), le débit en requêtes par unités de temps est $1/I_{min}$
 - Si S_{max} est la taille max d'une requête (bits), le débit soumis est S_{max}/I_{min} (bits par seconde)

$$PCR = S_{max}/(I_{min} * 48 * 8) \text{ Cellules/s}$$

Pour les services à débit variable, on a besoin de S_{moy} et I_{moy} qui permet d'obtenir **SCR** = $S_{moy}/(I_{moy} * 48 * 8)$

N_{raf} Taille max d'une rafale de requêtes de taille S_{max} sur une période d'observation T pendant laquelle on a évalué $S_{max} \geq S_{moy} \geq S_{min}$ et $T \geq I_{moy}$.

$$N_{raf} = \text{partie_ent}[(S_{moy} - S_{min}) * \text{partie_ent}[T/I_{moy}] / (S_{max} - S_{min})] = \mathbf{MBS}$$

⁴ Dans l'absolu, il faudrait tenir compte des données de gestion ajoutées qui sont spécifiques à l'AAL utilisée.

En Synthèse

- Pas véritablement de réseaux de liaison avec QoS à ce jour excepté ATM (et FDDI mais technologie tombée en désuétude), plutôt une bande passante éventuellement garantie (ADSL, RNIS, Frame Relay) et une approche Best Effort du point de vue des utilisateurs de ces liaisons.
- Technologie de commutation mise en œuvre dans les équipements, technologie qu'on retrouve en couche réseau pour IP. Ceci préfigure MPLS.
- Des débits qui augmentent et qui permettent d'envisager le support d'applications multimédia telles que VoIP mais aussi la Vidéo mais toujours d'un point de vue bande passante.

Internet classique et QoS

La gestion de la QoS temporelle dans les protocoles Internet classiques tient d'une politique "au mieux" ou "best effort".

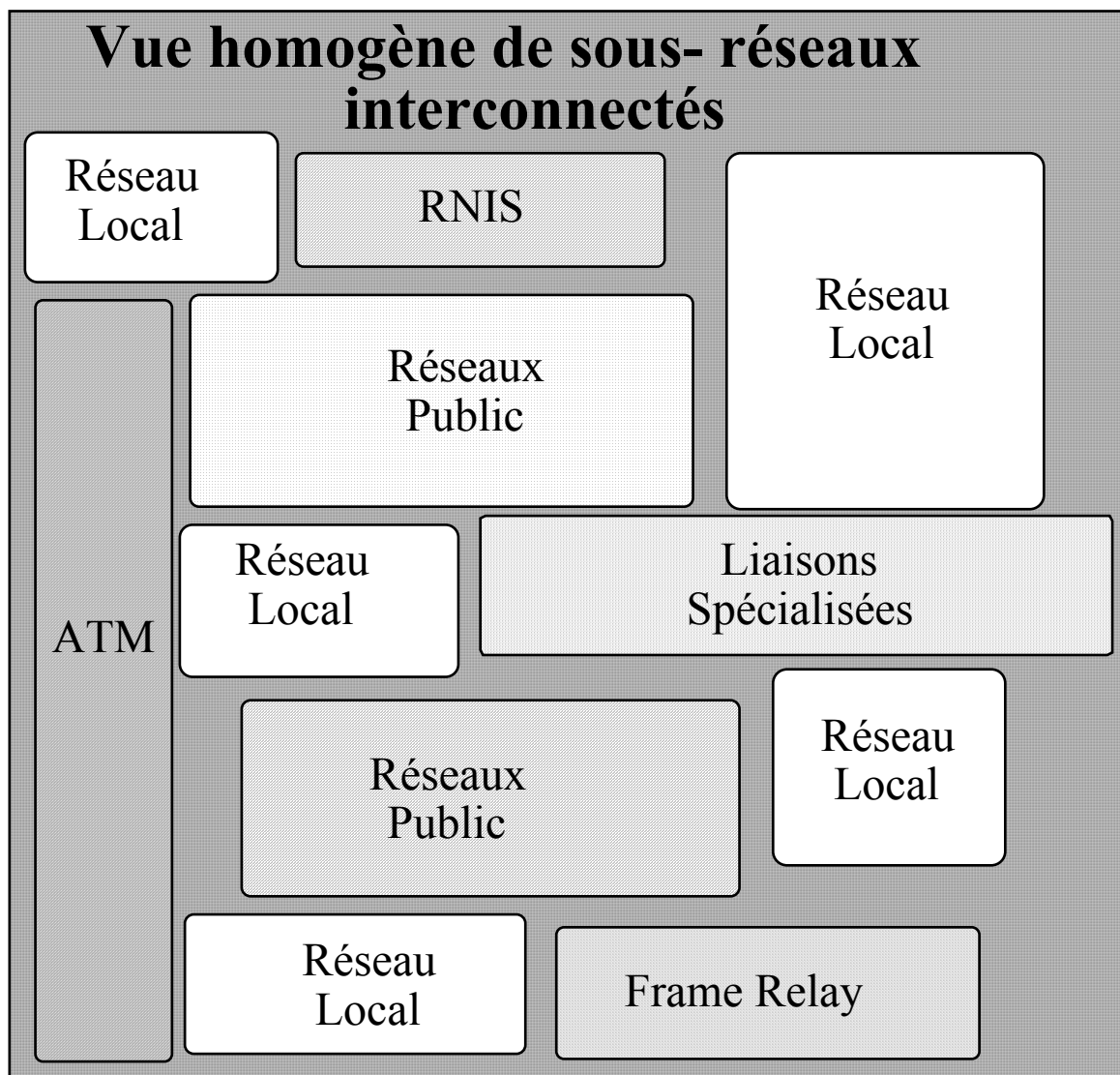
- IP
- TCP
- UDP/RTP

IP

IP: Fédération et Interconnexion de liaisons de données (sous-réseaux)

Hétérogénéité : Fournisseurs de services d'Interconnexion, Diamètre des liaisons de données, Modes d'Adressage

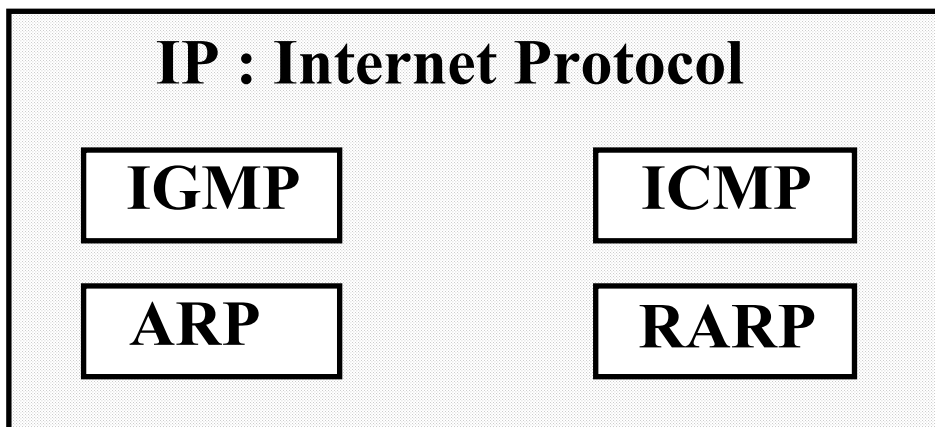
Un seul Réseau de transfert !



- **Homogénéisation** : Adressage et Adaptation de la transmission à la liaison traversée
- **Routage** : intra-domaine et inter-domaine
- **Contrôle de congestion** : Gestion des Ressources du réseau, plutôt guérison, la prévention est faite par TCP

Architecture d'IP

IP V4 :



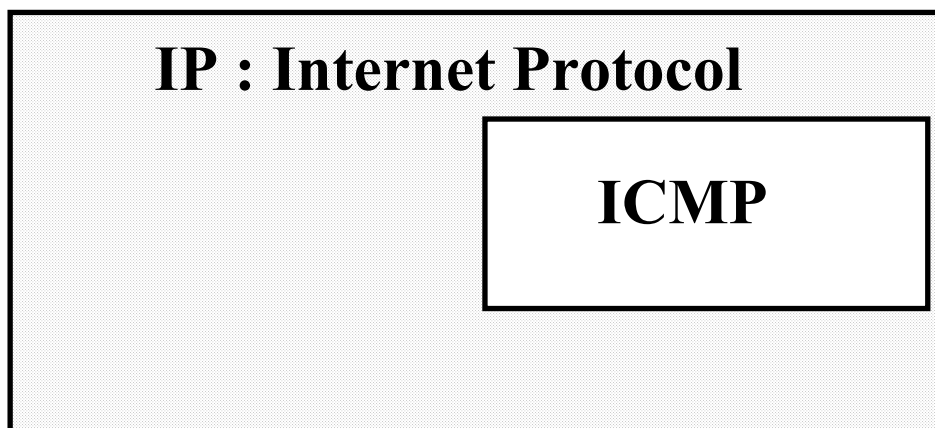
ICMP : Internet Control Message Protocol

ARP : Address Resolution Protocol

RARP : Reverse Address Resolution Protocol

IGMP : Internet Group Management Protocol (Multicast IP)

IP V6 :



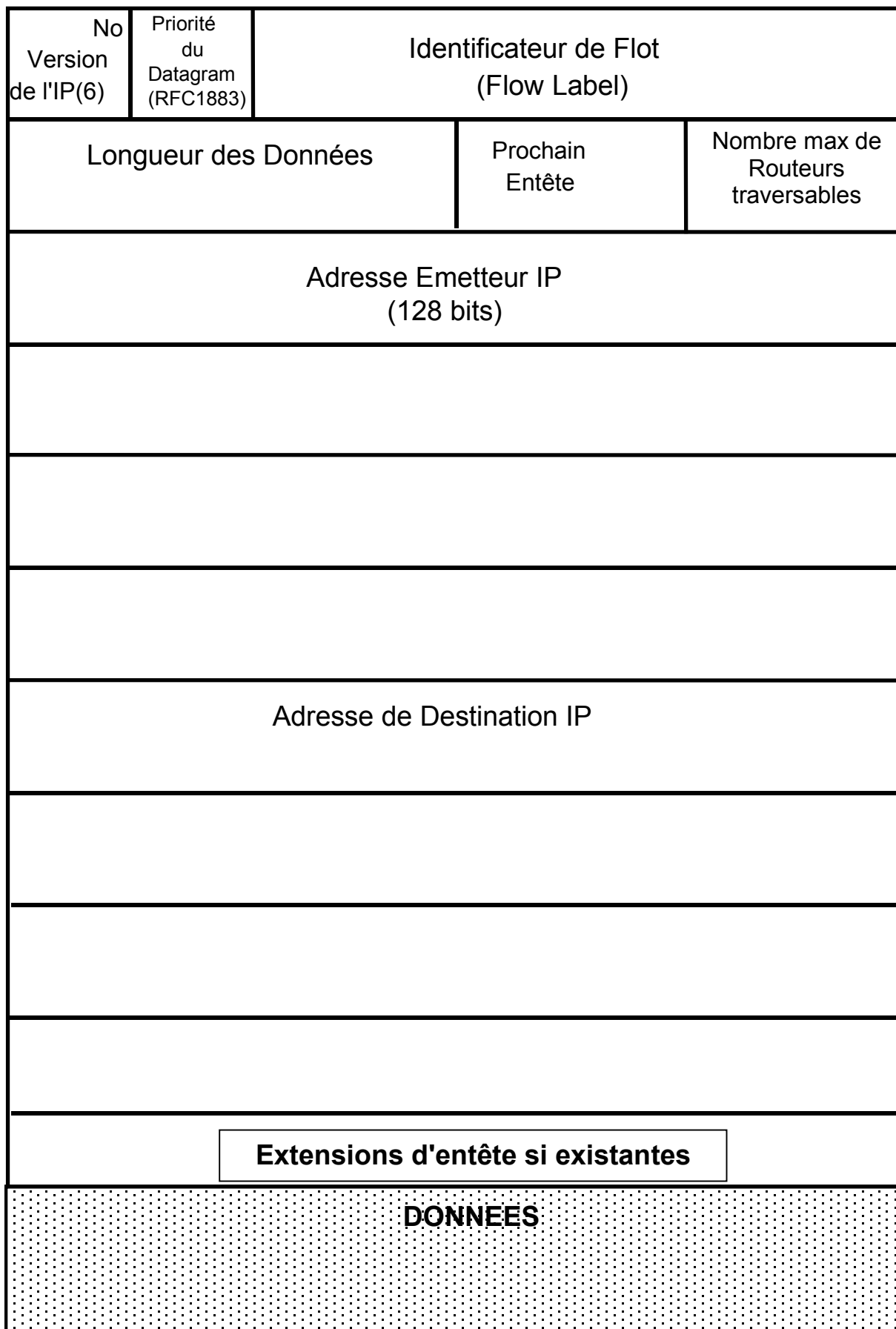
L' Internet Protocol (IP)

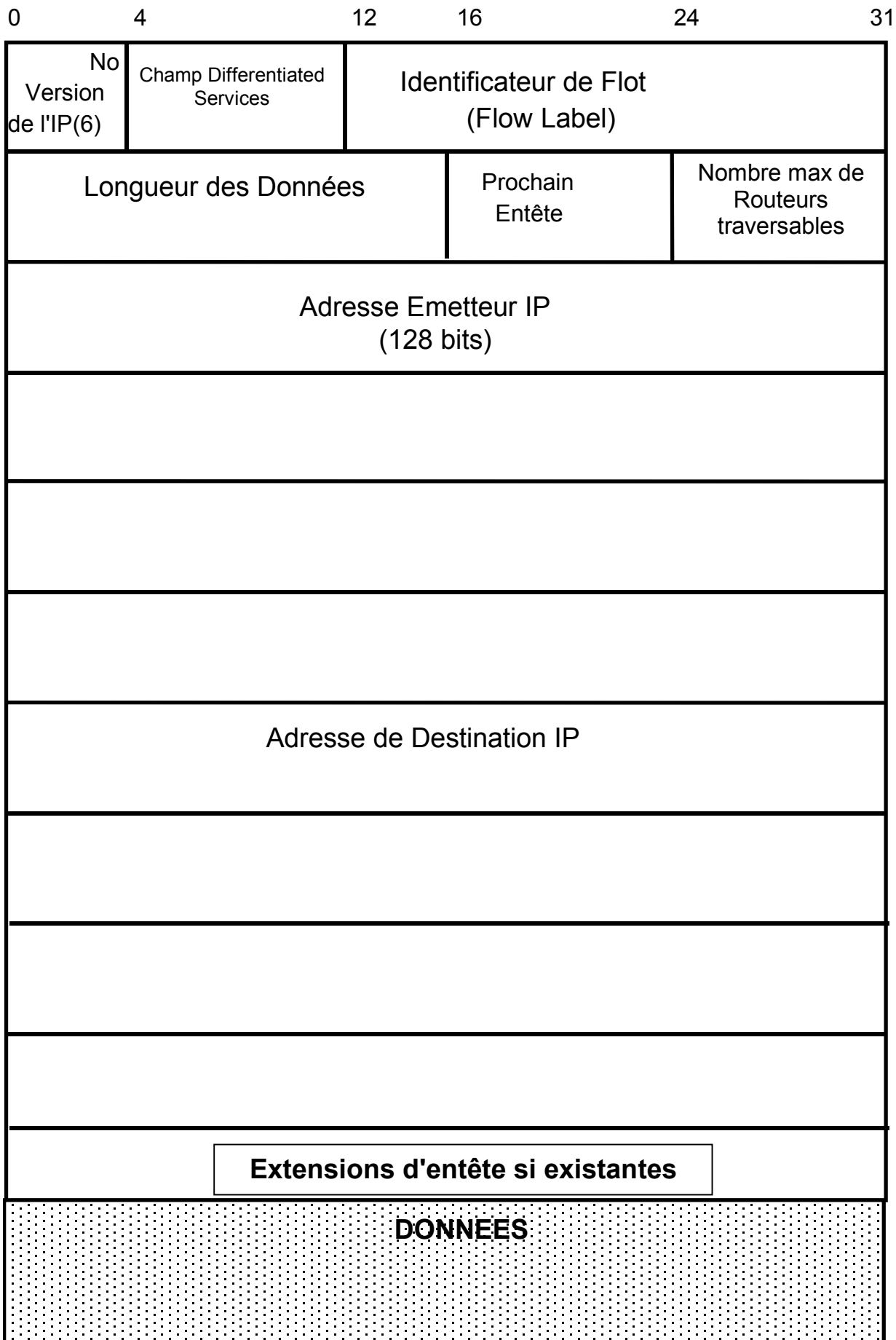
- Communications dans le mode minimal : **DATAGRAM** (mode non connecté, paquets **non Acquittés**)
=> la détection des messages erronés ou perdus et leur ré-émission sont à la charge de l'émetteur des messages (couche Transport).
- Adressage Internet et Routage entre Réseaux
- Conversions d'Adresses (@IP<->08:00:20:06:4b:8e) et adaptation à la liaison traversée
- Fragmentation/Ré-assemblage, Adaptation de la taille des messages soumis par la couche Transport suivant les possibilités offertes par la couche Liaison.
- Encapsulation/Désencapsulation par rapport à la couche Transport

0	4	8	16	19	24	31
No Version de l'IP(4)	Longueur de l'entête (nb de mots de 32 bits)	Façon dont doit être géré le datagram TOS - type of service	Longueur du Datagram, entête comprise (nb d'octets)			
No Id -> unique pour tous les fragments d'un même Datagram			flags (2bits): .fragmenté .dernier	Offset du fragment p/r au Datagram Original (unit en nb de blk de 8 o)		
Temps restant à séjourner dans l'Internet TTL	Protocole de Niveau Supérieur qui utilise IP		Contrôle d'erreurs sur l'entête			
Adresse Emetteur IP						
Adresse de Destination IP						
Options : pour tests ou debug					Padding: Octets à 0 pour que l'entête *32 bits	
DONNEES						

TTL : Time To Live, est exprimé en nombre de machines restant à traverser, décrétementé de 1 par chaque routeur franchi

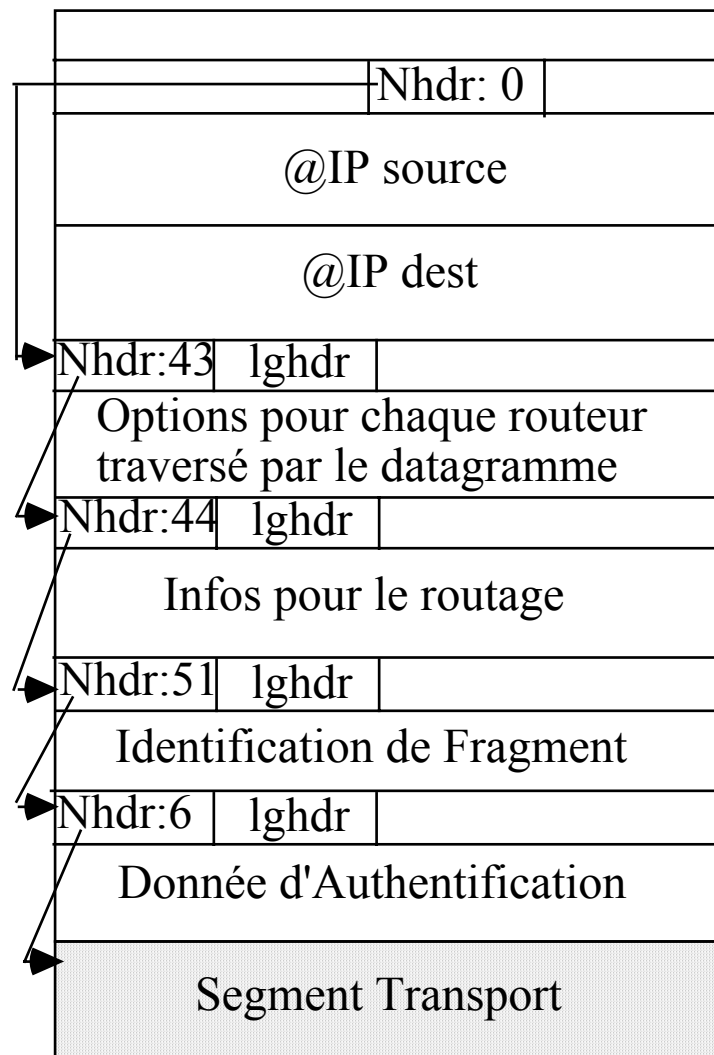
0 4 8 16 24 31





Utilisation du "Prochain Entête" en IPV6

Il est conseillé d'organiser les entêtes d'une certaine façon qui traduit l'ordre des traitements faits par un routeur lors de la commutation d'un datagramme :



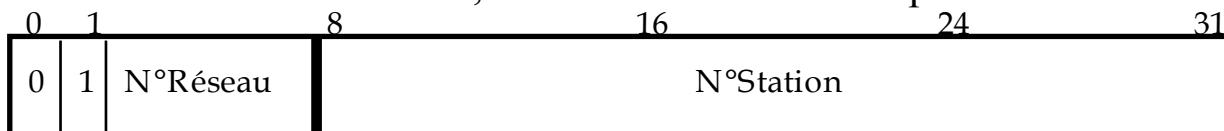
Le champ Nhdr contient le numéro d'extension d'entête du prochain champ (0-options de gestion du TTL, 43-routage...), la dernière extension contient le numéro du protocole transporté dans le champ Nhdr (même fonction que le champ "Protocole de niveau Supérieur du data gramme IPV4).

Adressage Internet IP V4 (32 bits)

Adresses Uniques Universelles :

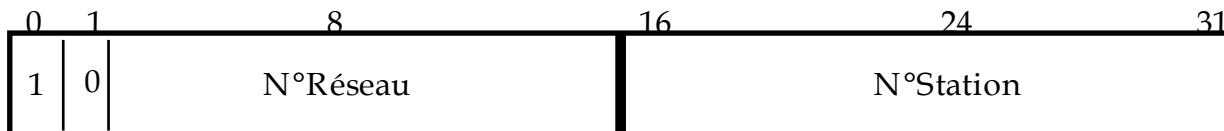
A . B . C . D
(N°Réseau, N°station)

Classe A : Peu de Réseaux, de nombreuses Stations par Réseau



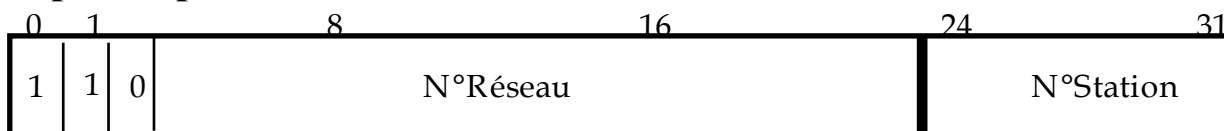
N°de Réseau : 1-126, **127** adresse de **rebouclage en local**

Classe B :



N°de Réseau : 128.1 - 191.254

Classe C : Beaucoup de Réseaux, Peu de Stations par Réseau, **La classe la plus répandue**



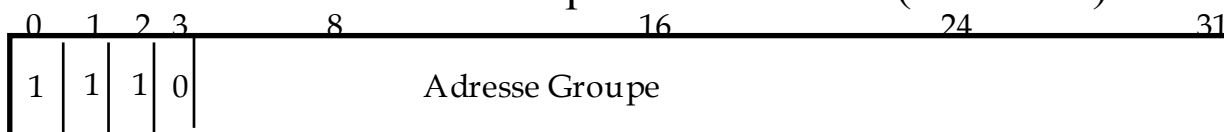
N°de Réseau : 192.0.1 - 223.255.254

N°de Station : 1 – 254

Broadcast : **255** dans le champ **N° de Station**

On fonctionne plutôt en mode CIDR (Classless Inter-Domain Routing) adresse/n n indiquant le nombre de bits réservés à la partie réseau de l'adresse.

Classe D : Adresses de Groupes de Diffusion (Multicast)



N°de Réseau : 225.0.0.0 - 239.255.255.255 (224.0.0.x réservée pour les protocoles de routage)

Adresses privées, Translation et Gestion dynamique

Pour chaque classe mentionnée ci-dessus, il existe des adresses privées complètement gérables par les utilisateurs. En général, on le couple à la translation d'adresse (NAT, Network Address Translation).

Sur un réseau, on peut faire une gestion d'adresse dynamique, en utilisant DHCP, Dynamique Host Configuration Protocol.

Ces mécanismes posent des problèmes pour l'identification des flots dans les routeurs. La gestion des filtres, et en particulier de la conformité des flux sont rendues plus difficile.

Adresses IP v6-IPng(128 bits)

Les nouvelles adresses sont sur 128 bits, la notation est donnée par groupe de 16 bits :

0108:0000:0000:0000:0008:0800:200C:417A

qui peut être simplifiée en :

0108:0:0:0:8:0800:200C:417A

ou encore en :

0108::8:800:200C:417A

(pas plus d'un seul "::" dans une adresse)

Exemples :

adresse privée à un sous-réseau (non routable) :

8 bits	n bits	m bits	p bits
11111110	0	n° ss-réseau	n° station

adresse dépendante d'un fournisseur :

3 bits	n bits	m bits	p bits	125-(n+m+p) bits
010	n° fourn	n° adhérent	n° ss-réseau	n° station

convergence IPv4 - IPng :

à des fins d'expérience, regarder le résultat de la commande `ifconfig -a` sur la machine kirov au cnam (machine Linux) qui a une adresse ipv6 unicast et une adresse ipv6 multicast contruites à l'aide de son adresse ethernet

Un exemple de Ping en IPV6

```
ping6 5F0D:E900:80DF:E000:0001:0060:3E0B:3010
```

```
PING 5F0D:E900:80DF:E000:0001:0060:3E0B:3010: 56 data bytes
```

```
64 bytes from 5f0d:e900:80df:e000:1:60:3e0b:3010: icmp_seq=0 time=43.1 ms
```

```
64 bytes from 5f0d:e900:80df:e000:1:60:3e0b:3010: icmp_seq=1 time=40.0 ms
```

```
64 bytes from 5f0d:e900:80df:e000:1:60:3e0b:3010: icmp_seq=2 time=44.2 ms
```

```
64 bytes from 5f0d:e900:80df:e000:1:60:3e0b:3010: icmp_seq=3 time=43.7 ms
```

```
64 bytes from 5f0d:e900:80df:e000:1:60:3e0b:3010: icmp_seq=4 time=38.9 ms
```

```
64 bytes from 5f0d:e900:80df:e000:1:60:3e0b:3010: icmp_seq=5 time=41.2 ms
```

```
64 bytes from 5f0d:e900:80df:e000:1:60:3e0b:3010: icmp_seq=6 time=39.1 ms
```

```
64 bytes from 5f0d:e900:80df:e000:1:60:3e0b:3010: icmp_seq=7 time=42.1 ms
```

```
- --- 5F0D:E900:80DF:E000:0001:0060:3E0B:3010 ping statistics ---
```

```
9 packets transmitted, 9 packets received, 0% packet loss
```

```
round-trip min/avg/max = 38.9/41.3/44.2 ms
```

source : Stéphane Bortzmeyer lorsqu'il était sysadmin à l'Institut Pasteur

Gestion du Datagramme - TOS (1)

Ce champ n'est pas géré par tous les algorithmes de routage (seulement OSPF et RIP V2).

0	1	2	3	4	5	6	7
Priorité		Type de Service					
		D	T	R			

La priorité influe sur la gestion des files d'attente des datagrammes vers une liaison de données. Celui qui a la préséance la plus élevée est transmis en premier. La priorité de 0 à 7 permet de marquer l'importance du datagramme.

000 : normal

001 : prioritaire

010 : immédiat

011 : urgent

priorités supérieures pour les messages de service réseau

On les retrouve dans le champ DSCP, avec la classe "Class selector".

Gestion du datagramme (2)

Le champ "Type de service" est lié à la métrique utilisée par le routage :

bit **T** : Débit (Througput/Bandwith) -> demande le plus grand débit

bit **R** : Fiabilité (Reliability/Error Rate) -> demande le plus faible taux d'erreur

bit **D** : Délais courts (Delay) -> demande le plus court délai (évite les satellites)

[bit **C** : Coût minimal (Cost) -> demande un coût minimal défini suivant les documents]

Combinaison de bits possible. Ce champ n'est interprété que par certains routeurs qui ont des algorithmes de routage de nouvelle génération supportant plusieurs métriques.

Gestion du datagramme (3)

En IPV6 il existe deux versions du champ priorité.

La version RFC 1883,

La version expérimentale liée à l'approche DiffServ (qui a un effet aussi en IPv4 avec redéfinition du champ TOS), le champ DS sur 8 bits est découpé comme suit :

- 2 derniers bits ignorés
- 6 premiers bits "Differentiated Services Code Point"
 - 3 derniers bits équivalent au TOS d'IPv4, ils définissent une priorité
 - 3 premiers bits indiquent une classe de service

(revu plus précisément dans la partie IP-DiffServ)

D'autres documents donnent un autre découpage... Ça semble s'être stabilisé avec DiffServ et la prise en compte de la gestion de QoS dans le routage.

Identification de Flux de données ou de Canal (spécificité Ipv6)

Un flot d'information entre deux entités est marqué/identifié par le champ "Flow label" et par l'adresse IP de la source. Le "flot label" est spécifié par le programmeur ou l'application.

Ce champ peut servir, au niveau d'un routeur, à optimiser des traitements : un émetteur pour un canal particulier indique toujours les mêmes options et ses datagrammes nécessitent toujours les mêmes traitements.

La marque peut servir de clef dans une table de routage.

Ce champ peut être utilisé pour RSVP et RTP vus plus loin.

IPV4: Fragmentation/Réassemblage des datagrammes

Il y a fragmentation quand un segment (unité de données de la couche Transport Internet) traverse des liaisons dont les sections sont plus petites.

La taille des données sur la liaison (**MTU**) est variable⁵ : 1500o (Ethernet), 1492o (IEEE802.3), 4464o (Token-Ring 4Mb/s), 17914o (Token-Ring 16Mb/s), 4352o (FDDI), 576o (X25), 296o (PPP) ...

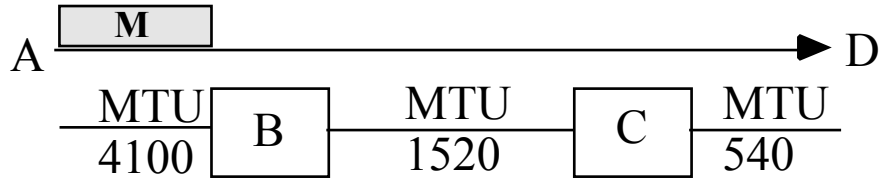
La fragmentation induit du temps de traitement. Du point de vue QoS, elle produit de la gigue. Il faut donc émettre avec la taille du plus petit MTU rencontré sur le chemin de données.

La fragmentation pose un autre problème : elle sépare entête IP et entête Transport (en particulier perte des numéros de port sur les fragments qui suivent), cette information sert à identifier un flot, l'identification sert lors de la mise en place d'architecture de réseaux à QoS (routeurs frontière).

⁵ Source R. Stevens dans TCP/IP Illustrated V1

Exemple de Fragmentation

200 Entête IP + 4000o Données



Un fragment a une taille multiple de 8 octets sauf le dernier.

Envoi par A d'un datagramme de 4020o

A->B:M Lg= 4020, DF=0, MF=0, position=0

Fragmentation sur B (1 datagramme = plusieurs paquets)

B->C: f1 Lg= 1520, DF=0, MF=1, position=0 (paquet 1)
 f2 Lg= 1520, DF=0, MF=1, position=1500 (paquet 2)
 f3 Lg= 1020, DF=0, MF=0, position=3000 (paquet 3)

puis Fragmentation sur C

C->D:f11 Lg= 520, DF=0, MF=1, position=0 (paquet 1)
 f12 Lg= 520, DF=0, MF=1, position=500 (paquet 2)
 f13 Lg= 520, DF=0, MF=1, position=1000 (paquet 3)
 f21 Lg= 520, DF=0, MF=1, position=1500 (paquet 4)
 f22 Lg= 520, DF=0, MF=1, position=2000 (paquet 5)
 f23 Lg= 520, DF=0, MF=1, position=2500 (paquet 6)
 f31 Lg= 520, DF=0, MF=1, position=3000 (paquet 7)
 f32 Lg= 520, DF=0, MF=0, position=3500 (paquet 8)

Assemblage sur le récepteur D. Attention, la fragmentation est pénalisante, elle induit du retard dans la traversée des routeurs, on préfère aujourd'hui se caler sur le plus petit MTU du chemin de données.

Fragmentation à la source en IPV6, la spécification des fragments utilise un entête "Fragment" (44) qui permet de reproduire ce qu'on avait en IPV4.

Traffic Engineering : Small is beautiful ?

Supposons qu'on envoie 8ko à travers un réseau de 4 routeurs reliés chacun par un multiplex E1 (2,048Mb/s).

1ère solution : 2 datagrammes de 4096o
 $(4096+40)*8 / 2,048 \text{ Mb/s} = 19,6 \text{ ms}$ par datagramme
soit 98.36 ms au total ($19,6 * 5$)

2ème solution : 16 datagrammes de 512o
 $(512 + 40)*8 / 2,048 = 2.15 \text{ ms}$ par datagramme
soit 41ms au total ($2,15 * 19$)

Ce résultat va à l'encontre des idées reçues qui considèrent qu'il vaut mieux envoyer de gros datagrammes pour rentabiliser l'effort de gestion protocolaire.

En fait, le raisonnement est faussé car on ne tient pas compte des temps de commutation dans les routeurs traversés, ce qui a une influence malgré les avancées technologiques. Ne pas oublier les délais de propagation, c'est important dans les liaisons satellite et trans-océan.

Le raisonnement n'est pas vrai pour les réseaux très haut débit (débat des jumbo-frames sur Ethernet 10Gb/s).

Encapsulation IP

Le champ "protocole de niveau supérieur" (8 bits) dans l'entête IPV4 indique à quel protocole est destiné le datagramme.

à titre indicatif :

- 1 : ICMP
- 2 : IGMP
- 4 : IP dans IP (encapsulation)
- 6 : TCP (Transmission Control Protocol)
- 8 : EGP (Exterior Gateway [=routeur] Protocol)
- 17 : UDP (User Datagram Protocol)
- 89 : OSPF

En IPV6, c'est le champ NextHeader qui joue ce rôle (celui de la dernière extension ou de l'entête standard si pas d'extension):

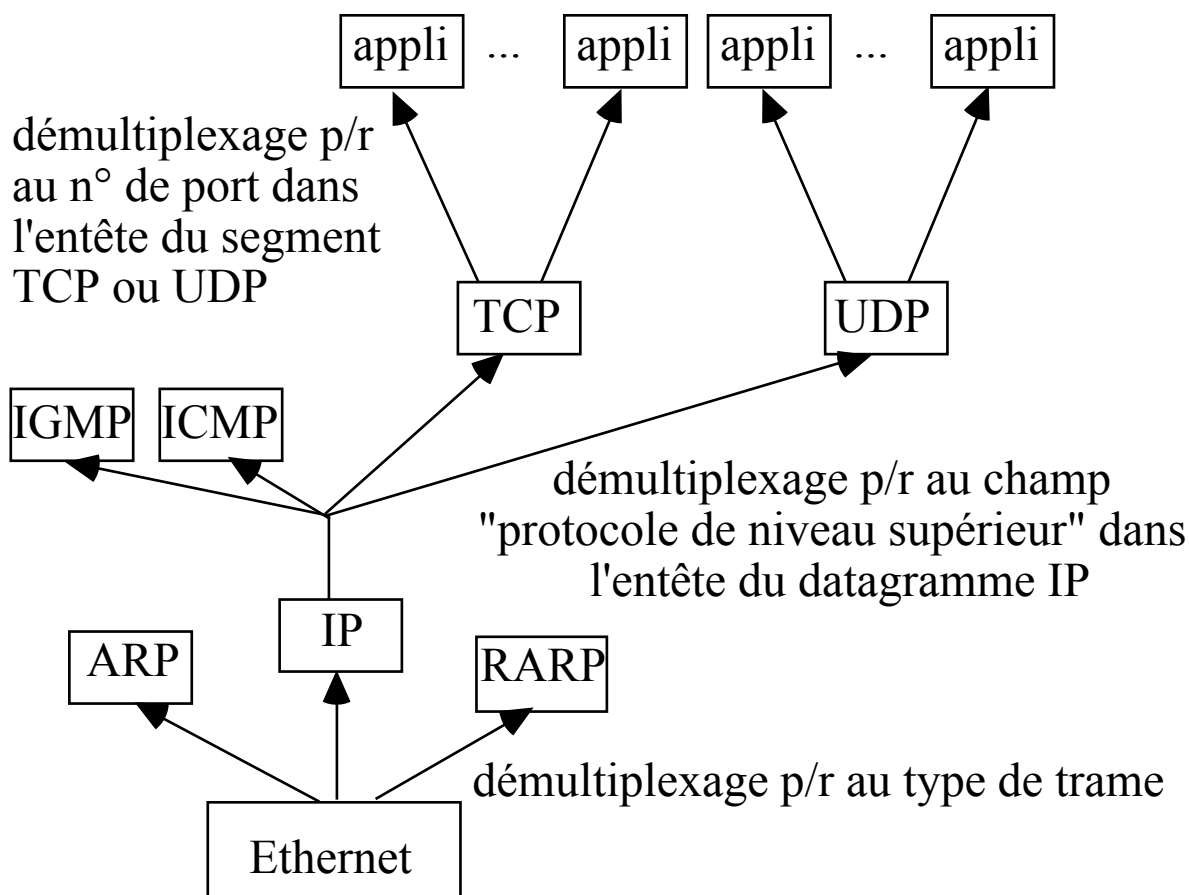
- 4 : IP dans IP (encapsulation)
- 6 : TCP (Transmission Control Protocol)
- 17 : UDP (User Datagram Protocol)
- 46 : Resource Reservation Protocol
- 58 : ICMP
- 59 : No Next Header

Son rôle est en fait beaucoup plus riche:

- 43 : Routing Header
- 44 : Fragment Header

C'est un champ qui intervient pour les tunnels.

De la trame à l'utilisateur en LAN



Implantation de IP V4

La couche IP n'examine pas le datagramme reçu champ par champ. Ca ne gênerait pas pour une station de travail, mais pour un routeur, ça serait inefficace.

L'implantation est optimisée pour les traitements les plus fréquents, en particulier, pour les datagrammes sans options.

Certains routeurs peuvent atteindre un taux de commutation de plusieurs Go/s ...

C'est pour cela que l'utilisation des options tomberait en désuétude. Et, pour faire du routage depuis la source, il faut nécessairement utiliser le champ option ...

IPV6 est affiné pour satisfaire l'objectif de performance! Et pourtant il utilise abondamment les extensions d'entêtes !!!

Protocoles de Routage et QoS

Protocole à vecteur de distance : RIP, RIPv2 seul habilité à prendre en compte plusieurs métriques. Problème de convergence du routage (trop long, crée des boucles), volume des informations transmises périodiquement (toutes les informations de routage du nœud). Reste un protocole limité à un campus.

Protocole à Etat des Liens : OSPF, supporte facilement la gestion plusieurs métriques, s'adapte au multicast et à la QoS ! Fonctionne sur la base de zones (Area), 1 zone = 1 campus. Plus rapide dans le calcul du routage, le nœud calcule lui-même puisqu'il a l'état de tous les liens.

Protocole inter-zone tel que BGP ?

Pour plus d'informations voir le cours de JP. Arnaud sur le routge.

Protocoles de Transport TCP/UDP-RTP, autres approches

Relations Applications/Transport

- UDP :
 - Client/Serveur en LAN
 - Multimedia en LAN /WAN
 - Multicast
 - TFTP, RTP, NFS, OSPF, RIP, SNMP, VoIP
 - ...

- TCP :
 - Transfert de données fiable (fichiers, terminal virtuel ...)
 - Client/Serveur en WAN
 - Unicast
 - DNS, Telnet, FTP, HTTP, SMTP, NNTP, NFS, BGP, LDAP ...

95% du flux Internet est de type TCP.

Transmission Control Protocol - TCP

- Orienté Flot d'octets

ne préserve pas la notion d'enregistrement (du point de vue de l'utilisateur au niveau API socket),
séquencement des octets garanti.

- Mécanisme de Circuit Virtuel : notion de connexion, en Full-Duplex

Acquits Positifs avec Retransmissions en cas d'erreurs,
Contrôle de flux
Pas de duplications des messages possibles,
Données urgentes,
Informé des ruptures de connexions (le moment dépend de la configuration choisie).

- Contrôle d'Erreurs,

Pas adapté au multimédia, aux réseaux haut débit actuels, aux réseaux de mobiles.

Protocole qui a été modifié pour améliorer ses performances et participer au **contrôle de congestion global de l'Internet.**

Diamètre d'une connexion TCP

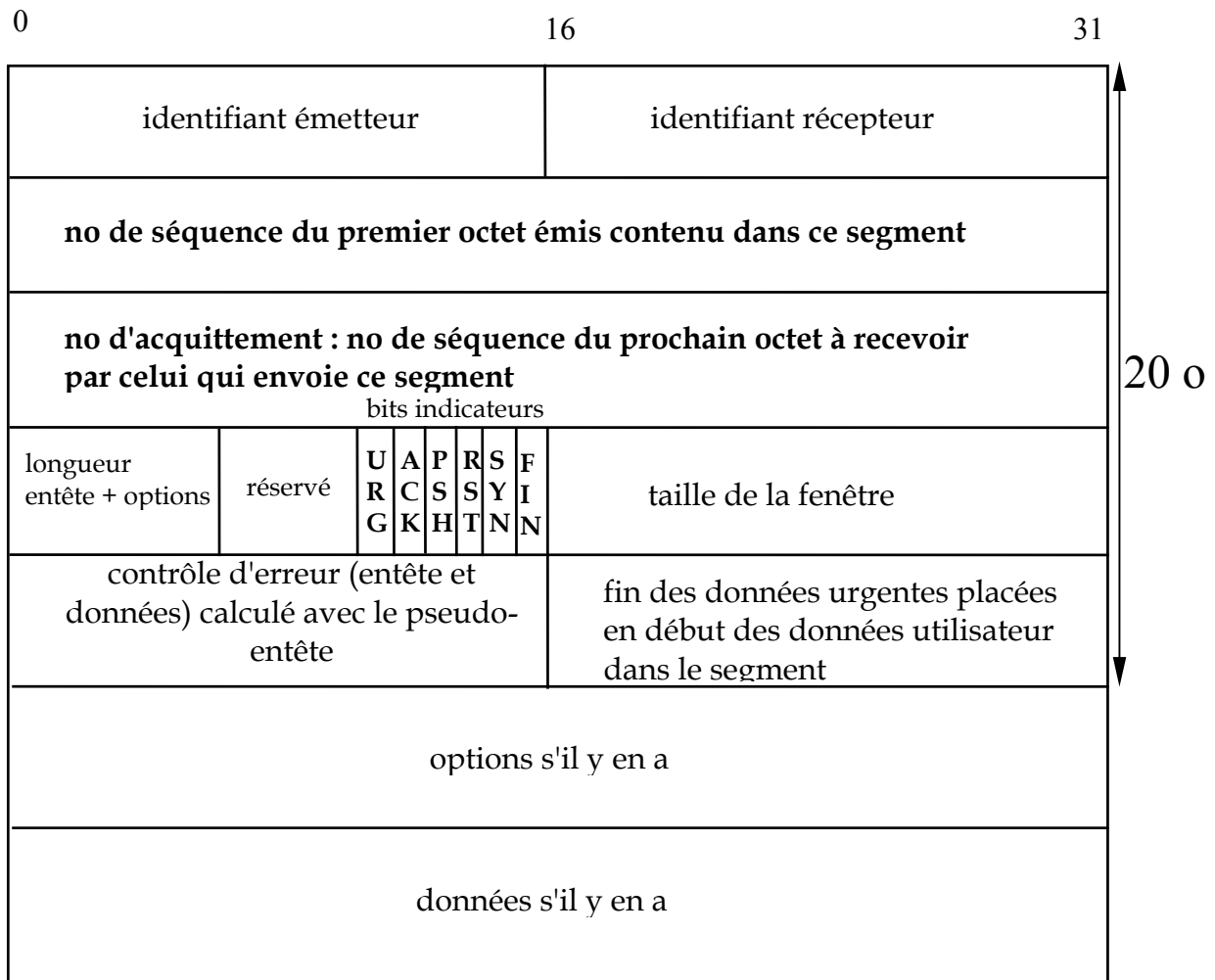
Connaître le diamètre permet d'éviter la fragmentation en cours de route.

Le diamètre de la connexion correspond au MTU du plus petit lien rencontré sur une connexion. Quand une connexion TCP est ouverte, TCP utilise le paramètre MSS fourni par l'autre entité ou le MTU de l'interface de sortie.

Les datagrammes sur cette connexion ont le bit DF à 1 (Don't fragment). Un routeur qui doit fragmenter, élimine le datagramme et génère un message ICMP "can't fragment". Suivant la version d'ICMP, la taille du MTU avec le prochain noeud peut être indiquée.

Les prochains datagrammes envoyés sont plus petits. Toutefois, comme les routes sont multiples, et qu'elles changent, TCP tente périodiquement (toutes les 10 mn recommandé) d'augmenter la taille des segments.

Segment TCP



Les bits indicateurs, s'ils sont positionnés, informent sur la nature du segment :

. **"SYN"** initialisation d'une connexion, dans ce cas le numéro de séquence porté indique le numéro du premier octet du flot de données, un segment contenant un SYN consomme un octet dans le flot d'octets de données, le numéro *i* du premier numéro de séquence est déterminé aléatoirement

. **"ACK"** acquittement des octets -> numéro envoyé – 1, acquit positif

. **"RST"** réinitialisation de connexion

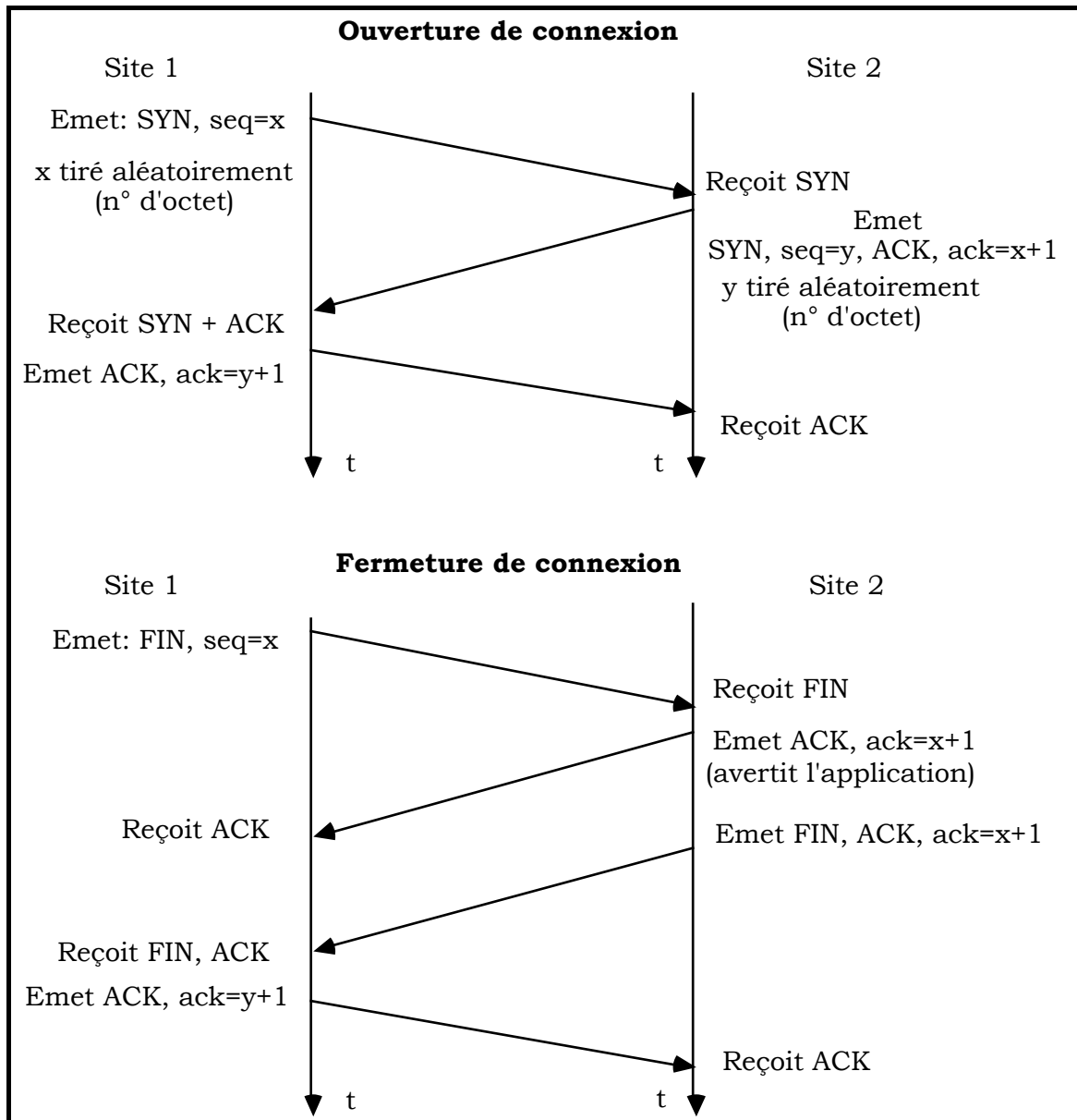
. **"URG"** données urgentes contenues dans le segment

. **"PSH"** délivrer les données au plus tôt au récepteur dès qu'elles sont correctement reçues

. **"FIN"** plus aucune donnée ne sera envoyée par celui qui a fait FIN.

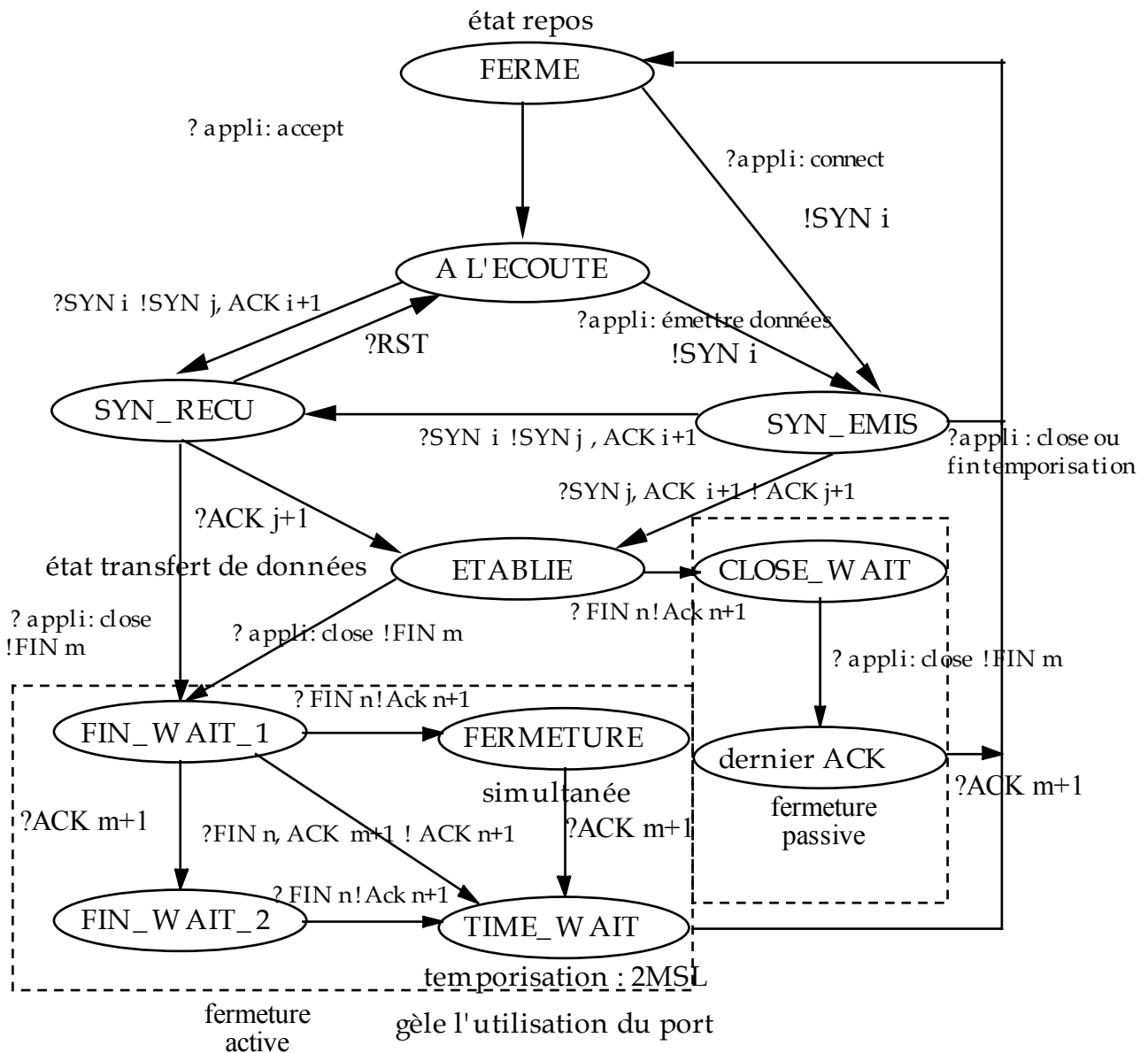
Champ option : type (1 octet), longueur totale du champ option(1 octet), infos associées à l'option (variable)

Ouverture et Fermeture de Connexion



Extrêmité de connexion : @IP + n°port, quand on ajoute le numéro du premier octet associé à chaque extrêmité d'une connexion, on définit une "instance de connexion".

Automate Protocolaire



MSL : Maximum Segment Lifetime (30s, 1mn, 2 mn)
généralement, le client fait la fermeture active, et le serveur la fermeture passive, la connexion peut rester bloquée longtemps (max 4mn), c'est le client qui gèle l'instance de connexion car son port n'est pas réutilisable pendant 2MSL

TIME_WAIT -> pb lors de réutilisation de port si un client enchaîne ouverture/fermeture de connexion sur un même port.
Option pour court-circuiter ce mécanisme : SO_REUSEADDR

Contrôle de flux TCP

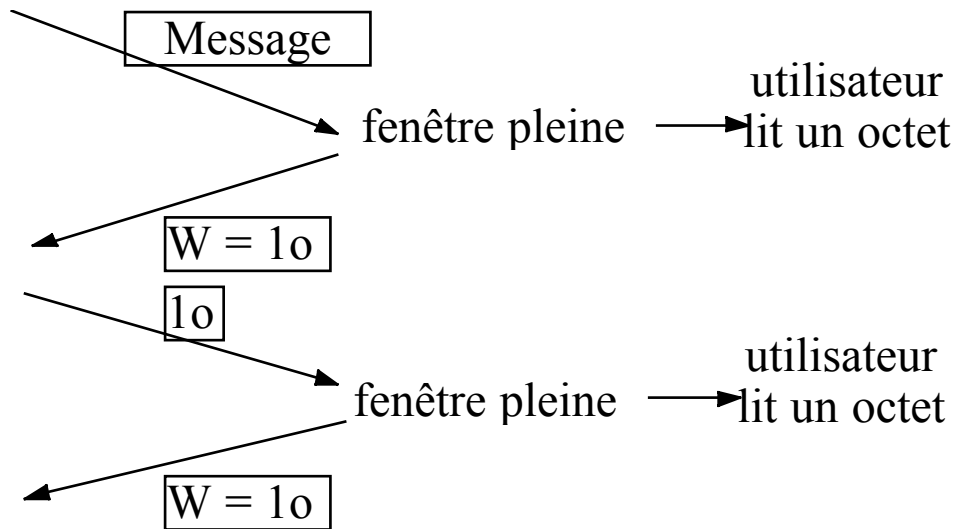
Mécanisme de fenêtre d'octets : chaque extrémité indique une taille de fenêtre, champ sur 16 bits (soit 65 535 o).

Le récepteur envoie son crédit en fonction des octets de données que retire son utilisateur.

La valeur de la taille de la fenêtre est indiquée dans chaque segment (Principe de crédit plutôt que de fenêtre. C'est pour mieux résister aux pertes de segments, en effet, si un émetteur ne reçoit pas son crédit, il ne reste bloqué que jusqu'au message suivant). Il existe un timer, "persist timer" qui oblige l'émetteur à demander le crédit périodiquement s'il n'y a pas d'échange.

Silly Window Syndrome

Problème qui se produit quand on transmet de grands blocs côté émetteur, et que le récepteur ne lit ses données que par très petits bouts.



Solution :

Le récepteur ne donne que des tailles de fenêtre dont la valeur est au moins supérieure à la moitié de la taille maximum de la fenêtre.

L'émetteur respecte les conditions suivantes pour envoyer ses segments :

- a) segment de longueur max autorisé (taille max fenêtre)
- b) segment de taille $>$ moitié de la taille max de la fenêtre
- c) pas obligatoire d'attendre un ack en réponse à l'envoi d'un segment pour transmettre un nouveau segment

Dimensionnement de la fenêtre

Pb encore plus important avec les réseaux à haut débit.

capacité d'une connexion (*bit* ou en *octet* si on / par 8)
= débit nominal (b/s) * délai de propagation⁶ A/R

Type de Réseau	nominal b/s	nominal o/s	RTT (ms)	capacité (o)
Ethernet 10Mb/s	10Mb/s	1,25Mo/s	3ms	3,750ko
Ethernet 100Mb/S	100Mb/s	12,5Mo/s	3ms	37,5ko
Ethernet 1Gb/s	1Gb/s	125Mo/s	3ms	375ko
multiplex T1 cont	1,544Mb/s	0,193Mo/s	60ms	11,58ko
multiplex T1 sat	1,544Mb/S	0,193Mo/s	500ms	95,5ko
multiplex T3 cont	44,736Mb/s	5,592Mo/s	60ms	335,52ko

La capacité d'une connexion est à comparer avec la taille de la fenêtre.

La taille de la fenêtre (65,5ko) est trop petite pour certains réseaux. La taille de la fenêtre peut être augmentée avec l'option "window scale" qui permet de définir une taille de fenêtre sur 32 bits (spec d'option : type = 3, lg = 3, valeur).

Le champ option contient un facteur multiplicatif (valeur). Une valeur de 2 dans le champ option "window scale" donne une valeur de $65535 * 2^2$, soit 256 Ko. Cette option s'utilise à l'ouverture de cnx, nécessairement par les deux entités (TCP est full-duplex, les entités sont toutes les deux émettrices).

⁶ au niveau transport, ne pas confondre avec le niveau physique, ce délai de propagation A/R est spécifique à chaque connexion de Transport, il fait l'objet d'une évaluation périodique par la couche Transport

Fast Recovery and Fast Retransmit

TCP offre un transfert fiable au-dessus d'un réseau à datagrammes donc potentiellement non fiable et pouvant subir des erreurs de transmission et des congestions (plus fréquent). Deux techniques sont utilisées pour les retransmissions :

- Go Back N

A chaque segment émis est associé une temporisation, si elle arrive à échéance, on retransmet tous les segments émis depuis le segment dont la temporisation a expiré. L'évaluation de la valeur de cette temporisation est importante, elle est fondée sur le délai de propagation A/R

- Fast Retransmit and Fast Recovery

Dès qu'un récepteur reçoit un segment hors séquence, il envoie un ACK avec le no du prochain octet à recevoir, et sauve le segment reçu en attendant de combler le trou dans le flot de données avec le segment manquant.

Un émetteur qui reçoit plusieurs fois le même ACK (duplicated ACK) suspecte qu'il y a eu une perte de segment. Au bout de 3 ACKs identiques, il ré-émet le segment manquant. Ce phénomène est aussi un indicateur de congestion.

Vieux Paquets et Epuisement de l'espace de n°

Combien de temps pour épuiser l'espace des n° de séquence TCP (espace de numérotation de 4 Go):

Type de Réseau	nominal b/s	nominal o/s	délai d'épuisement
Ethernet 10Mb/s	10Mb/s	1,25Mo/s	53mn
Ethernet 100Mb/S	100Mb/s	12,5Mo/s	5mn20s
Ethernet 1Gb/s	1Gb/s	125Mo/s	32s
Multiplex T1 cont	1,544Mb/s	0,193Mo/s	46mn
Multiplex T3 cont	44,736Mb/s	5,592Mo/s	12mn48s
ATM	155,52Mb/s	19,44Mo/s	3mn40s

Quelle valeur de MSL faut-il pour qu'un vieux paquet ne soit pas accepté comme un paquet correct du flot d'une cx, 30s, 1mn, **2mn**?

Slow Start – Congestion Avoidance

Mécanisme qui participe au contrôle de congestion de l'Internet. Le contrôle de congestion est géré à deux niveaux réseau et transport, mais plutôt au niveau Transport. Principe au niveau automate de Transport : Quand l'émetteur détecte une congestion, il diminue le débit des données qu'il soumet.

Détection ? Quand une temporisation arrive à échéance pour un segment envoyé (pas d'ACK ou ACK en retard p/r à la tempo d'où l'importance du réglage des temporisations qui par ailleurs évoluent dynamiquement), en effet une erreur de transmission est maintenant trop peu fréquente, ou situation de duplicated ack.

Que fait-il ? Il maintient une **fenêtre de congestion**. En fait, la taille des données émises est le minimum de la taille courante de la fenêtre de congestion, et du crédit alloué par le récepteur (combinaison capacité réseau et capacité récepteur).

Quand il reçoit un ACK avant expiration de tempo associée à son 1er segment émis, il augmente la taille de sa fenêtre de congestion de 1 segment⁷. Il peut envoyer 2 segments à chaque segment acquitté. L'augmentation exponentielle de la taille de la fenêtre ! Attention, il n'envoie jamais plus que la taille du crédit spécifié par le récepteur.

A un certain moment, la capacité maximale du réseau est atteinte, ce qui se traduit par "un segment est perdu"... l'ACK ne revient pas (fin de temporisation)! Il y a congestion !! Idem si ACK dupliqué ! La fenêtre de congestion est trop large. Idem à la réception d'un message ICMP SOURCE_QUENCH

Un deuxième mécanisme entre en jeu, le contrôle de congestion. Le seuil de congestion est initialisé à la moitié de la fenêtre de congestion au moment où la congestion a été détectée, et la fenêtre de congestion est ré-initialisée à 1 segment.

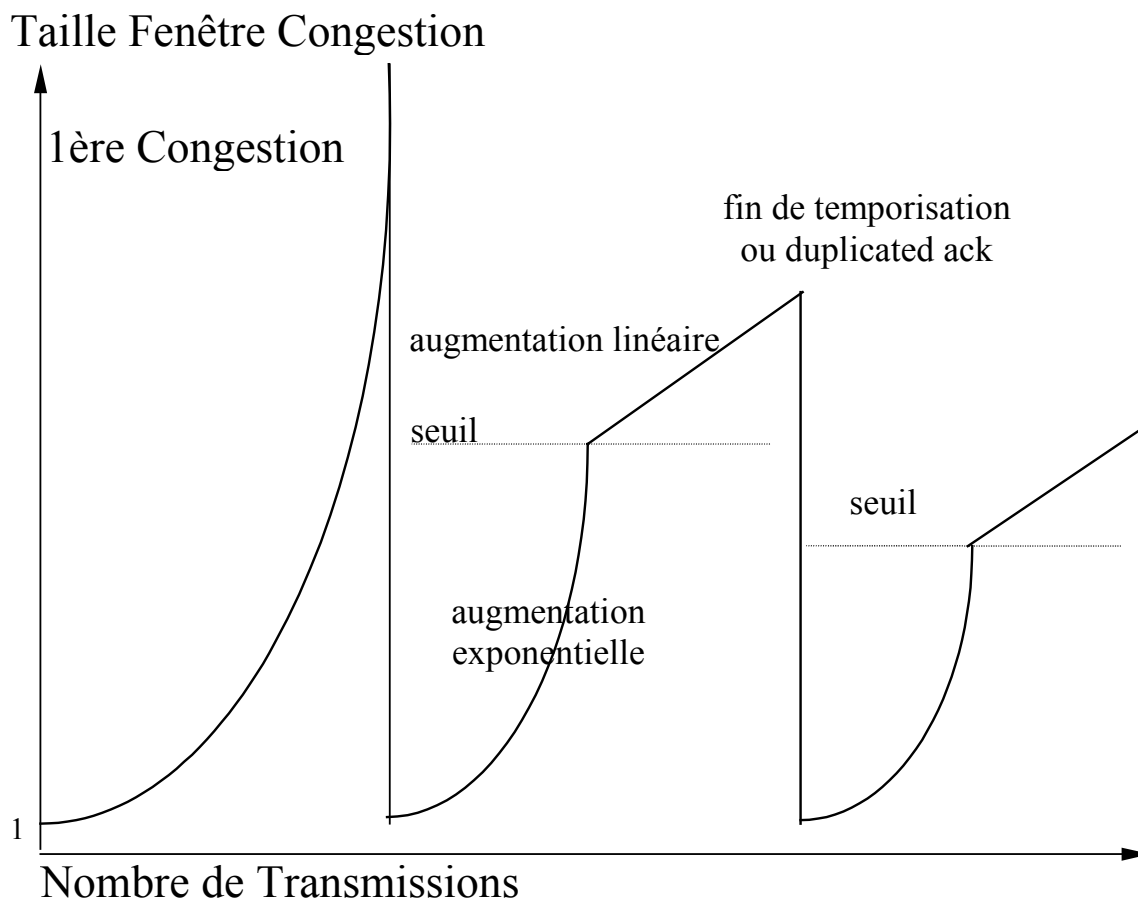
Maintenant, on recommence comme à la phase précédente, quand la fenêtre de congestion atteint le seuil de congestion, l'augmentation du débit ne progresse plus que linéairement (un segment de plus par temps de propagation A/R).

La détection d'une nouvelle congestion divise à nouveau la fenêtre de congestion de moitié.

⁷ dans la réalité, TCP raisonne en nombre d'octets et pas en nombre de segments

Evolution de la taille de fenêtre de congestion

Ce mécanisme garantit la stabilité de l'Internet.



Le volume pouvant être émis est toujours le minimum de la taille de la fenêtre de congestion et du crédit.

Remarque : Tant qu'il n'y a pas de congestion la fenêtre de congestion augmente.

Calcul du Délai de Propagation A/R et de la temporisation de retransmission

Le délai de propagation A/R est la base de paramétrage du délai de temporisation pour retransmission. Il est estimé en permanence.

La difficulté tient à ce que le RTT évolue au plus près de la charge réelle du réseau : une temporisation trop courte déclenche une retransmission sans raison, ce qui charge le réseau puis déclenche le slow-start de façon prématurée et sous-dimensionne le seuil de congestion.

Mesure du RTT : temps séparant l'émission d'un segment de la réception de son acquit par le récepteur (segment non retransmit).

Si une mesure de RTT est en cours pour un segment, il n'y a pas de nouvelle mesure prise pour un nouveau segment émis.

Le RTT est mesuré en tick d'horloge TCP, 1 tick vaut généralement 500ms dans une implantation standard.

1^{ère} proposition :

$R = a * R + (1-a) * M$, "Retransmit Timeout" $RTO = R * b$

R ancienne estimation du RTT, M dernier temps mesuré pour un RTT
a vaut généralement 0,9 et b vaut 2

Ce résultat ne prend pas en compte les grandes variations de RTT

2^{ème} proposition : (Jacobson 1988)

$Err = M - A$

$A = A + g * Err$, A est une estimation du RTT moyen où $g = 1/8$

$D = D + h * (|Err| - D)$, D est une estimation de la variation moyenne où $h = 0,25$

$RTO = A + 4D$

Ce résultat est meilleur.

Débit d'une cnx TCP – modèle analytique

D'après Madhavi et Floyd (1997), la bande passante d'une connexion TCP est :

$$B = 1,22 * MTU / (RTT * \sqrt{\text{taux de perte}})$$

On suppose que le MTU est celui de la connexion et que les datagrammes sont de même longueur et de taille MTU.

Ce résultat provient d'une analyse du mécanisme d'évitement de congestion en régime permanent pour une connexion.

User Datagram Protocol - UDP

Protocole de Transport :

- sans connexion,
- sans acquits,
- ne conserve pas l'ordre des messages,
- sans contrôle de flux,
- préserve la notion d'enregistrement.

=> possibilités de :

- pertes de messages,
- duplication des messages (si ré-émission),
- déséquencement,
- émetteur trop rapide p/r au récepteur

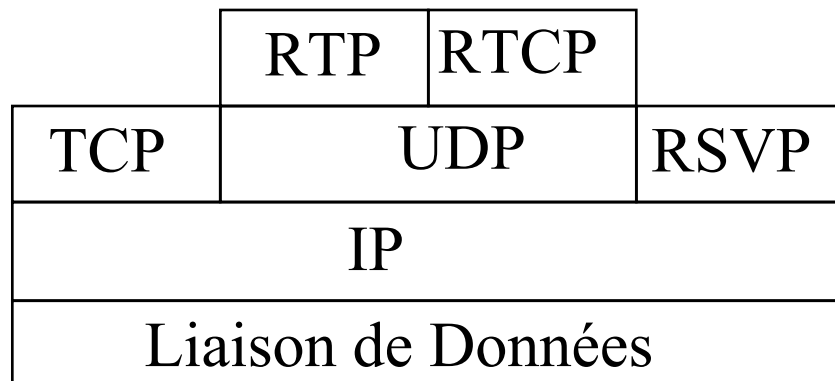
Protocole avec peu de traitement,

Protocole qui n'a pas été modifié depuis sa conception (1980).

Entête : 8 octets (2o port source, 2o port destination, 2o longueur du datagramme exprimée en octets, 2o CRC sur la totalité du datagramme avec un calcul utilisant une pseudo-entête)

Contrôle d'erreur activable ! Généralement activé même en LAN aujourd'hui.

Pile de protocoles Temps Réel IP à l'origine



Aucune hypothèse sur la couche Liaison de Données, excepté le fait que certaines liaisons peuvent ne pas être satisfaisantes.

RTP : Real time Transport Protocol

RTCP : Real Time Control Protocol

RSVP : Resource Reservation Protocol

applications visées: audio-conférence, visio-conférence donc de type **isochrone**

RTP/RTCP ont une évolution complètement indépendante de
RSVP maintenant

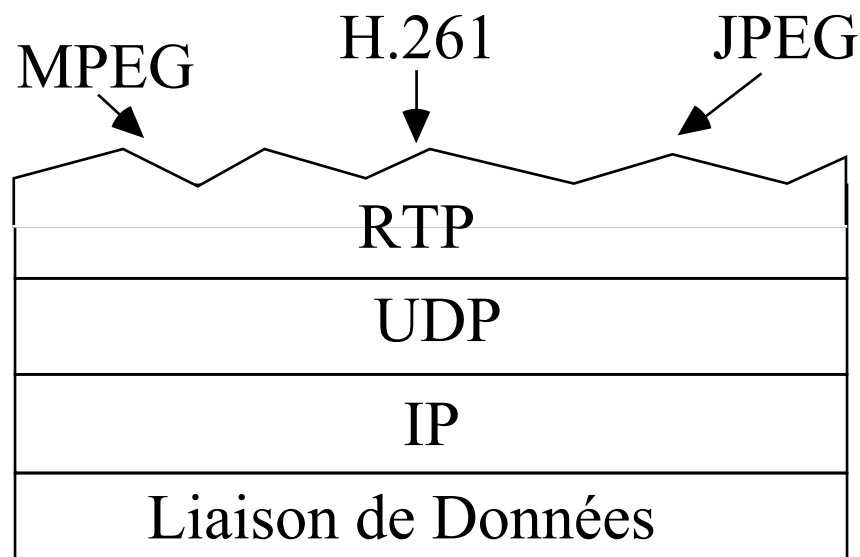
RTP utilise le port 5004

Hypothèses de Conception de RTP

Les flux de données vidéo, audio, tolèrent des pertes de messages mais pas des discontinuités trop grandes dans leur cadencement.

RTP se **combine** avec des protocoles de plus haut niveau spécialisés pour un type de média.

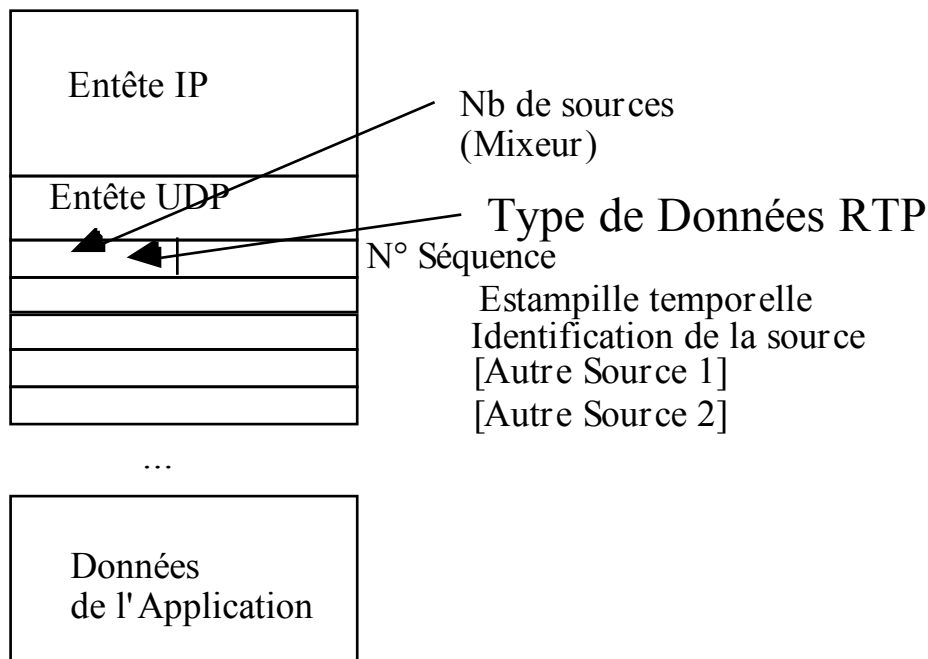
Exemple d'utilisation de RTP avec des techniques de compression vidéo :



A partir du flux de données, le récepteur doit pouvoir re-synchroniser les informations pour les restituer. La source doit pouvoir mettre les estampilles temporelles nécessaires au destinataire des messages RTP.

L'objectif de conférence implique des flux de données en diffusion (multicast).

Message RTP



Le **numéro de séquence** permet de délivrer les datagrammes dans l'ordre de la source, la **date** permet de reconstituer la base de temps du flux généré par la source, d'où le re-cadencement du flot en réception.

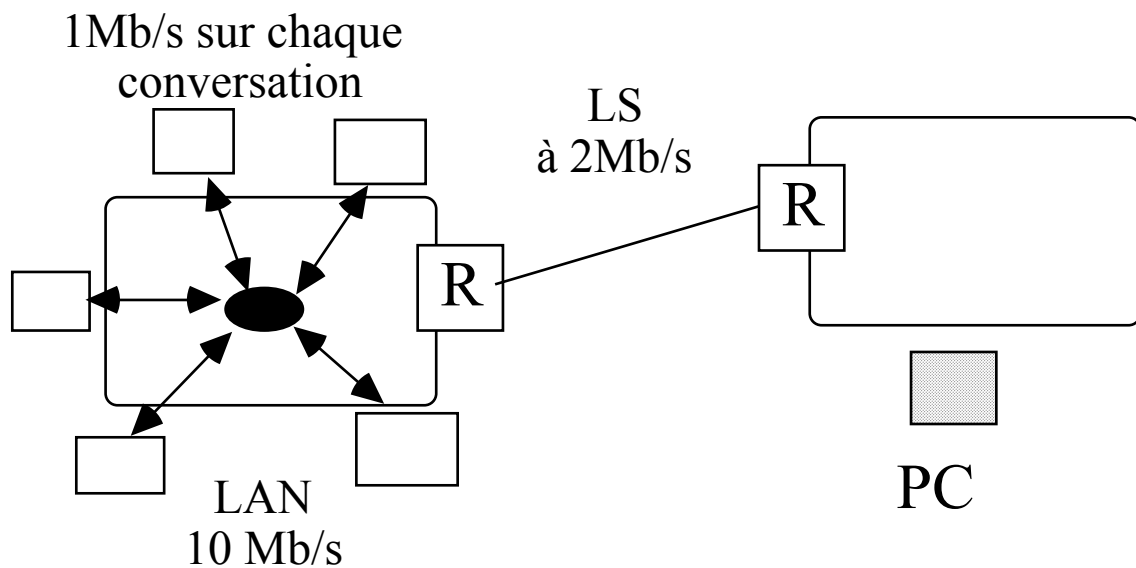
Le type de données (7 bits) permet d'identifier le contenu du message : 4 par exemple pour de l'audio encodée G.723 (voix paquetisée à 5.3 et 6.3 kb/s pour VoIP et VoFR).

La source est le premier émetteur du message, il détermine le numéro de séquence, l'estampille temporelle (date). Les translateurs préservent l'identification de la source tandis que les mixeurs la modifient, voir ci-après.

Remarque : entête standard UDP+RTP = 20 octets

Pb à résoudre

Vidéoconférence entre des stations sur un réseau local, un PC sur un autre réseau distant veut rejoindre la conférence :

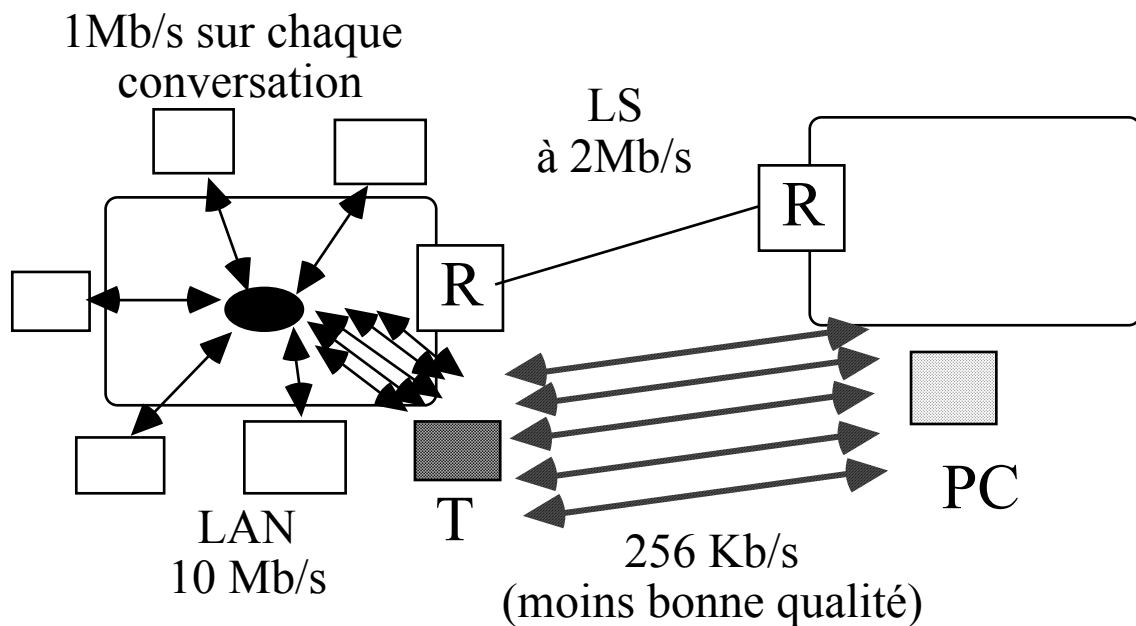


Trafic total sur le LAN de 5Mb/s ne peut pas passer sur la LS.

Solution : Utilisation de Translateurs et de Mixeurs

Translateurs

Le translateur est une sorte de convertisseur capable de modifier un flot de données (isochrone) en un flot de moins bonne qualité :



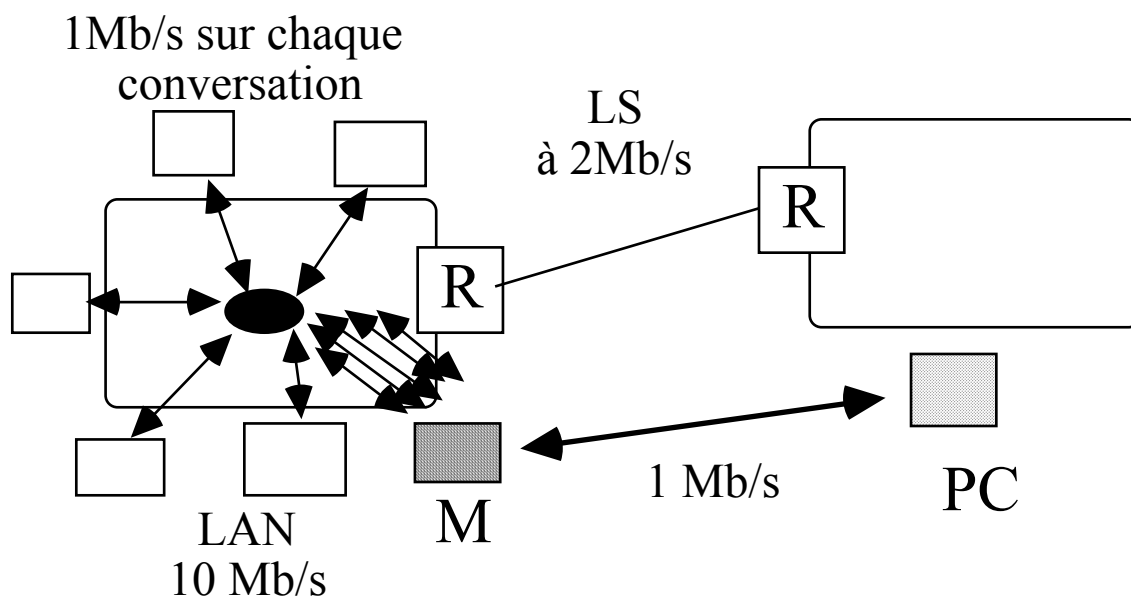
En entrée, le Translateur accepte les flots de 1 Mb/s, et les convertit en flots de 256 Kb/s

La vidéoconférence peut maintenant atteindre le PC.

Le mécanisme de translateur peut servir à traverser les pare-feux (un de chaque côté d'un pare-feu).

Mixeurs

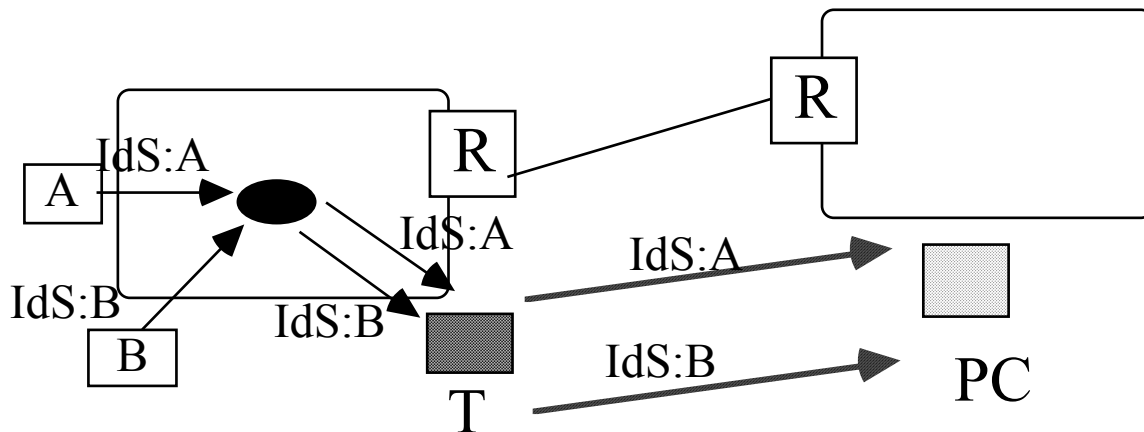
Les Mixeurs ont un objectif équivalent à celui des Translateurs sauf qu'ils combinent les flots.



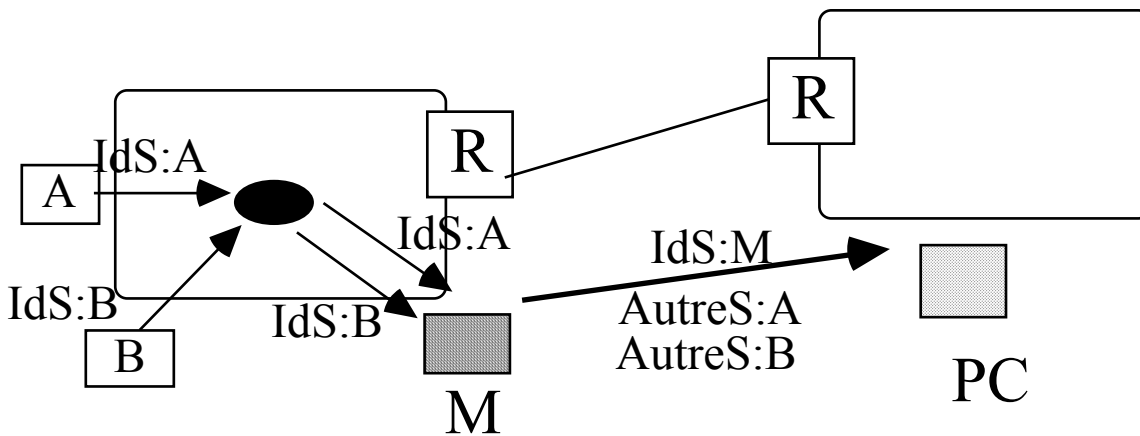
Cette technique est plus adaptée à des flots de données audio.

Identification de sources

Translateurs (la source n'est pas modifiée)



Mixeurs



Protocole RTCP

Accompagne le protocole RTP, correspond au port 5005.

RTP => flot de données

RTCP => flot de contrôle (compte-rendu retour vers la source en particulier)

Permet d'échanger des "rapports d'activité", 5 types de messages :

- **Rapport Emetteur** : l'émetteur envoie périodiquement aux récepteurs ce qu'ils auraient du recevoir, un émetteur peut être l'initiateur de la conférence, mais aussi un participant : estampilles temporelles (temps absolu émetteur, date RTP), nb de messages RTP, nb d'octets de données transmis, délai depuis le dernier rapport émetteur, délai écoulé depuis le dernier rapport récepteur...

- **Rapport Récepteur** : pendant du rapport émetteur pour un site qui ne fait que recevoir, ceci permet de contrôler la qualité du flux reçu

- **Description de la Source**

- **Fin de participation**

- **Type spécifique**, dépend de l'application

RTCP pourrait servir à faire de la réservation de ressource (proposition YESSIR par H. Schulzrinne).

Alternatives et Etudes

- FEC incorporé à RTP (Forward Error Correction, les messages sont dupliqués dans le flot émis, les répliques peuvent être compressées)
- TCP Vegas (TCP avec un contrôle du débit de la source en fonction de la bande passante estimée disponible)
- TCP friendly (UDP + compléments qui se comporte comme TCP pour résoudre/réduire la congestion du réseau)
- TCP SACK

Autres approches

XTP – Xpress Transport Protocol :

Hypothèses de conception : taux d'erreur sur les liens de transmission faible, intégration en une seule couche des couches réseau et transport, intégrable dans du silicium, destiné aux communications temps réel.

Types de services supportés :

- Connexion
- Transaction
- Datagramme
- Datagramme avec acquit
- Flot isochrone
- Transfert en rafale

Communications en groupe de diffusion supportées.

Toutefois, pas de protocole de gestion de ressources, ni de QoS.

Autres travaux :

- HeiTS : Heidelberg Transport System, destiné au multimédia, conçu avec IBM
- METS : Multimedia Enhanced Transport Service, destiné au Multimédia, conçu par l'université de Lancaster, fonctionne au-dessus d'ATM, et gère une QoS statistique.