Dédicaces

A mes parents

Aucun hommage ne pourrait être à la hauteur de l'amour

Dont ils ne cessent de me combler.

Que dieu leur procure bonne santé et longue vie.

A mon frère Madjid

A mes sœurs, Fazia et la mignonne Liza

A ma famille,

Aux familles haroune, boukline, lamani,…..

A mes amis :khaled, nassim, ahmed,rabah(massi),….

A mes enseignants :m^{me} Taouri, m^{re} Daoui, m^{me} Aoujit,….

Je dédie ce modeste travail.

Nassim berkani

Résumé

Le protocole IP acquiert de plus en plus d'importance dans le domaine des télécommunications en transmettant et recevoir des paquets IP, surtout pour les entreprises, grâce à ses avantages économiques et techniques.

Avec l'augmentation des applications envoyées sur les réseaux IP, et afin de fournir un service avec une bonne qualité, il faudrait utiliser des mécanismes assurant cette qualité et un degré plus élevé de performance pour la transmission optimale des paquets IP au sein d'un réseau.

Dans ce projet de fin d'étude, nous avons tout d'abord étudié le protocole IPv6 ou bien le protocole IPng (new generation) en implémentant à chaque fois un mécanisme de QoS (IntServ et DiffServ et MPLS) et en mesurant les différents paramètres de qualité de service.

Finalement, nous avons évalué ces mécanismes pour le protocole IPv6 pour assurer des qualités de performances.

Mots clés : Qualité de service QOS, Best effort**,** IntServ, DiffServ, MPLS, contrôle de flux, contrôle d'admission, classes de services, agregats de flux.

Avant propos

Ce document s'inscrit dans le cadre du projet de fin d'études en deuxième année master II académique en spécialité conduite des projets informatiques de l'université Mouloud Mammeri Tizi-Ouzou Algérie.

L'objectif de ce projet est de déterminer la QOS : qualité de service et les caractéristiques de performances dans le protocole IPv6 (IPng :IP new generation) pour assurer un acheminement optimal des paquets IP au sein d'un réseau en implémentant les mécanismes de garantie de QOS en implémentant les architectures IntServ et DiffServ et MPLS.

Table des matières

Liste des acronymes

6PE	IPv6 Provider Edge
AF	Assured Forwarding
BA	behavior agregat
BE	best effort
CIDR	Classeless Inter Domain Routing
CoS	Class of Service
DiffServ	Differenciated Services
DSCP	Differentiated Service Code Point

DP	Drop Precedence
EF	Expedited Forwarding
FEC	Forwarding Equivalence Class
FL	Flow Label
FIFO	First In First Out
ICMP	Internet Message Control Protocol
IETF	Internet Engineering Task Force
IP	Internet Protocol
ISDN	Integrated Services Digital Network
ITU	International Telecommunication Union
IntServ	INTegrated SERVices
LAN	Local Area Network
QoS	Quality Of Service
PBS	partial buffer sharing
PDU	Protocol Data Unit
PHB	Per Hop Behviour
PQ	priority queuing
LDP	Label Distribution Protocol
LER	Label Edge RouteurLSP Label Switch Path
LSP	Label Switched Path
LSR	Label Switching Router
MCU	Multipoint Control Unit
MF	multi field
MPLS	MultiProtocol Label Switching
MTU	Maximum Transfert Unit
OSPFv6	Open Shortest Path First version 6
RNIS	Réseau Numérique à Intégration de Services
RIP	Routing Information Protocol
RS	Redirect Server
RSVP	Resource Reservation Protocol
RTCP	Real-Time Transport Control Protocol
RTP	Real-Time Transport Protocol
SDP	Session Description Protocol
SLA	Service Level Agreement
TCA	Traffic Condition agreement
TCP	Transport Control Protocol
TOS	Type Of Service
UDP	User Datagram Protocol
VoIP	Voice over Internet Protocol
WAN	Wide Area Network
WFQ	weighted fair queuing

Introduction générale

La qualité de service ou QoS (Quality of service) est un nouveau concept incontournable dans le monde des réseaux des télécommunications. Bien que complexe, il n'a rien de révolutionnaire, puisqu'il se fonde sur des technologies préexistantes, qu'il vise à rationaliser, et souvent à simplifier, afin d'en faciliter la mise en œuvre. Néanmoins, la normalisation n'est pas encore achevée et de nombreuses philosophies s'affrontent. Ce concept revêt de multiples aspects technologiques, et qui doit être précisé selon ses objectifs et son contexte d'utilisation.

En fait, la plupart des réseaux informatiques et télécommunications reposent aujourd'hui sur le protocole IP, conçu à l'origine pour véhiculer des données informatiques. Cependant, avec la prolifération du réseau Internet marquée par la croissance exponentielle du trafic et la naissance de nouvelles applications, ce réseau se trouve face à des problèmes sévères. Le best effort n'est plus suffisant pour suivre l'évolution des technologies. Donc, il ne s'avère pas un support de communication qui permet de satisfaire pleinement les contraintes variées de ces nouvelles applications.

Face à cette impasse, un effort de recherche important a été mené, ces dernières années, pour qu'Internet offre un support d'interconnexion où de nombreux types d'applications puissent cohabiter et fonctionner raisonnablement. Des solutions ont émergé : celles requérant une nouvelle architecture appelée « avec état » tels que l'architecture IntServ et celles maintenant la propriété « sans état » comme DiffServ.

Et face à ces technologies de la QOS, on va les intégrer au nouveau protocole IP new generation pour assurer une transmission optimale des paquets IPv6.

Chapitre I : Généralités sur les réseaux informatiques :

I.1 introduction :

Un réseau est constitué d'un ensemble d'ordinateurs et d'organes périphériques, généralement éloignés et reliés entre eux par un système de communication, suivant une architecture, une topologie, et une technologie.

Le terme générique « réseau » définit un ensemble d'entités (objets, personnes, etc.) interconnectées les unes avec les autres. Et il permet ainsi de faire circuler des éléments matériels ou immatériels entre chacune de ces entités selon des règles bien définies.

I.2 définition des réseaux informatiques:

Réseau (informatique) : ensemble d'ordinateurs et de terminaux interconnectés pour échanger des informations numériques.
Ensemble de machines interconnectées qui servent à échanger des flux d'information.

- ***Réseau (Network)*** : Ensemble des ordinateurs et périphériques connectés les uns aux autres. (Remarque : deux ordinateurs connectés constituent déjà un réseau).

- ***Mise en réseau (Networking)*** : Mise en œuvre des outils et des tâches permettant de relier des ordinateurs afin qu'ils puissent partager des ressources.

Un ordinateur est une machine permettant de manipuler des données. L'homme, en tant qu'être communiquant, a rapidement compris l'intérêt qu'il pouvait y avoir à relier ces ordinateurs entre-eux afin de pouvoir échanger des informations.
Un réseau informatique peut servir plusieurs buts distincts :

- Le partage de ressources (fichiers, applications ou matériels, connexion à internet, etc.)
- La communication entre personnes (courrier électronique, discussion en direct, etc.)
- La communication entre processus (entre des ordinateurs industriels par exemple)
- La garantie de l'unicité et de l'universalité de l'accès à l'information (bases de données en réseau)
- Le jeu vidéo multijoueur

I.2.1 Similitudes entre types de réseaux :

Les différents types de réseaux ont généralement les points suivants en commun :

- **Serveurs** : ordinateurs qui fournissent des ressources partagées aux utilisateurs par un serveur de réseau
- **Clients** : ordinateurs qui accèdent aux ressources partagées fournies par un serveur de réseau
- **Support de connexion** : conditionne la façon dont les ordinateurs sont reliés entre eux.

- **Données partagées** : fichiers accessibles sur les serveurs du réseau Imprimantes et autres périphériques partagés : fichiers, imprimantes ou autres éléments utilisés par les usagers du réseau.
- **Ressources diverses** : autres ressources fournies par le serveur.

I.2.2 Equipement pour réseau :

Pour installer un réseau informatique il nous faut des ressources matériels et logiciels :

- **Les ressources matérielles** :
  - Connectiques : câbles, prises, fiches de connexion…
  - Cartes réseau et carte modem
  - Concentrateurs (hubs), routeurs, commutateurs (switchers) …

- **Les ressources logicielles :**
  - Système d'exploitation pour réseau
  - Protocoles de communication
  - Application pour réseau (messagerie, logiciels de gestion de réseau…).

I.2.3 type des réseaux :

- ✓ **Classification selon l'étendue géographique :**

1. **Réseaux locaux (*Local Area Networks*, LAN) :**
   - Communication au sein d'une organisation (département d'entreprise, etc.)
   - Administration unique
   - Couverture géographique limitée (~1 km)
   - Débit élevé, taux d'erreur faible
   - Topologies diverses : bus, anneau

2. **Réseaux à grande distance (Wide Area Networks, WAN) :**
   - Communication entre des organisations diverses
   - Administrations multiples
   - Couverture géographique étendue : un pays, toute la planète
   - Débit variable, taux d'erreur parfois non négligeable
   - Topologie maillée ; interconnexion de réseaux (exemple : l'Internet)

3. **Réseaux métropolitains (Metropolitan Area Networks, MAN) :**
   - Intermédiaires entre LAN et WAN - quelque dizaines de km, ville ou région
   - Les MAN interconnectent plusieurs LAN géographiquement proches.

4. **Réseau personnel (Personal area network) PAN :**
   - C'est réseau constitué autour d'une personne (de l'ordre de quelque mètres).

✓ **Classification selon l'architecture :**

1. Les réseaux clients serveurs :

Organisés d'un ordinateur servant de serveur (serveurs de fichiers, serveurs de bases de données, Serveurs d'impressions…) et des poste Clients, appelés également stations.

2. Les réseaux poste à postes (égal à égal, Peer to Peer)

Les réseaux Peer to Peer permettent aux stations connectées (de 2 à 30 ordinateurs) de communiquer directement entres elles. Toutes les machines du réseau possèdent le même statut.

✓ **Classification selon la technologie :**

La connexion d'un ordinateur à un réseau peut se faire par :

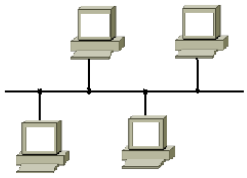
- Câble : réseau câblé ou filaire
- Onde : réseau sans fil (exemple WIFI…)

I.2.4 Topologie des réseaux :

Une topologie désigne la manière dont les dispositifs d'un réseaux sont organisés , autrement dit c'est la structure des réseaux en terme de lien d'interconnexion entre les stations. Et on distingue 4 types :

✓ **Bus :** C'est la topologie la plus simple, mais elle a des inconvénients. Il est nécessaire d'avoir des répéteurs lorsque le nombre d'ordinateurs augmente. Un problème sur le câble entraîne une panne du réseau.

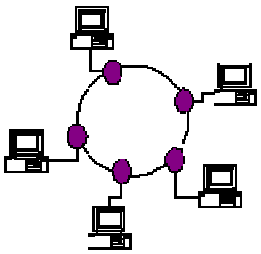
dans cette topologie chaque machine est reliée a un cable commun.



✓ **Anneau :** chaque machine est reliée à deux équipements voisins,

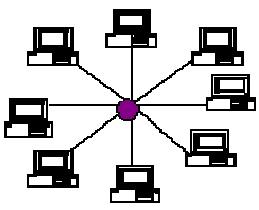
On obtient ainsi une boucle fermée.

C'est, en fait, un bus refermé sur lui-même.



✓ **Etoile :** Tous les ordinateurs sont reliés à un dispositif matériel central appelé **HUB**.

Chaque nœud est indépendant des autres.



- ✓ **Arbre :** c’est une topologie en bus dans la quelle une connexion donne naissence a un nouveau bus, comme il necessaite des repeteurs.

I .3 Le modele de reference OSI :

Le modèle OSI (Open Systems Interconnection) définit de quelle manière les ordinateurs et les périphériques en réseau doivent procéder pour communiquer.

- Il spécifie le comportement d'un système dit ouvert.
- Les règles de communication constituent les protocoles normalisés.
- Le modèle OSI est normalisé par l'ISO.

I.3.1 Description du modèle :

Le modèle OSI se décompose en 7 parties appelées couches.

- Ce modèle date des années 1980
- Chaque couche est responsable de l'un des aspects de la communication.
- Une couche de niveau N communique avec les couches N+1 et N-1 par le biais d'une interface.
- Une couche inférieure transporte les données vers la couche supérieure sans en connaître la signification.
- Les couches N de 2 systèmes communiquent à l'aide de protocoles de communication commun.

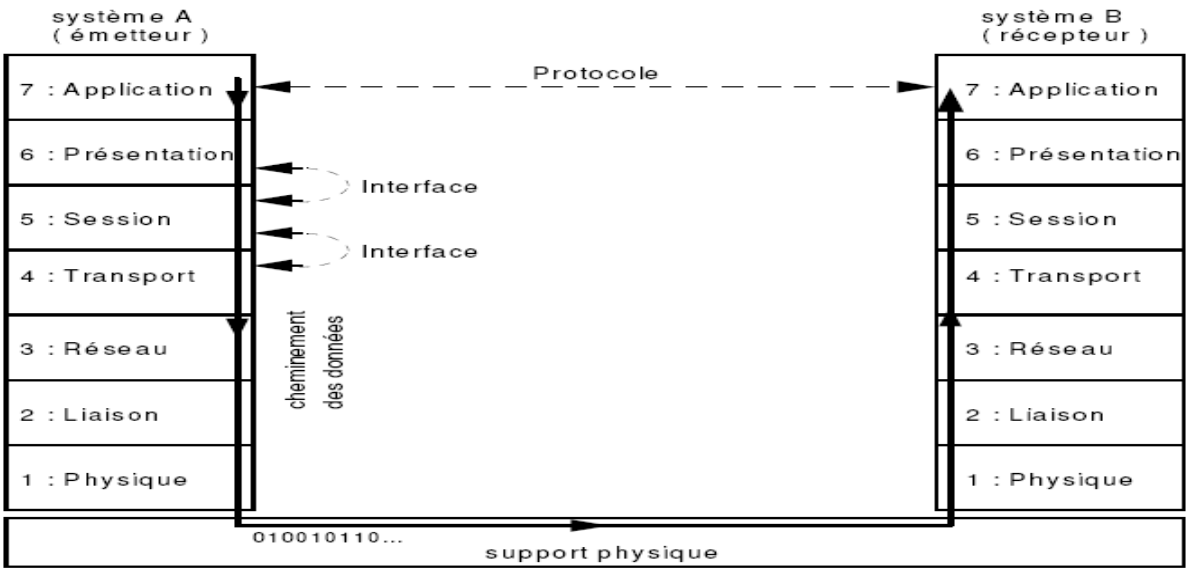


figure **I.3.1 : Description du modèle OSI**

Couches 1 à 3 : couches basses orientées transmission.
Couche 4 : couche intermédiaire.

Couches 5 à 7 : couches hautes orientées traitement.

L'organisation en couches permet d'isoler des fonctions réseaux et de les implanter indépendamment de l'ensemble du système.

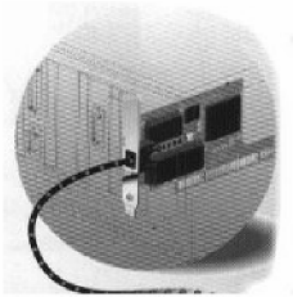
Cette organisation facilite l'évolution des logiciels réseau (Client / Serveur), en cachant les caractéristiques internes de la couche, au profit de la description des interfaces et des protocoles.

I.3.2 Services rendus par chaque couche :

I.3.2.1 Niveau 1: Couche Physique :

- Elle se charge de l'adaptation du signal au support de transmission,

ce qui définit les caractéristiques électriques,



logiques et physiques de la connexion de la station sur le réseau. (Câbles, connecteurs, cartes réseau...)

- Elle gère le type de transmission (synchrone ou asynchrone)
- S'il y a lieu, elle met en œuvre les mécanismes de modulation et démodulation du signal
- L'unité d'échange est le bit.

I.3.2.2 Niveau 2 : Couche Liaison :

- Elle définit les règles d'émission et de réception des données à travers la connexion physique de deux systèmes..
- Elle doit transmettre les données sans erreurs et détermine la méthode d'accès au support.
- Elle met en œuvre la détection et la correction des erreurs
- Elle gère les ré-émissions s'il y a lieu
- Elle établit et contrôle la liaison au niveau logique
- L'unité d'échange est la trame (frame)

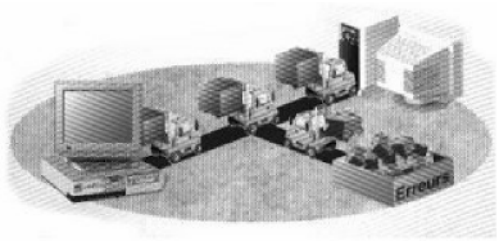
  EX : Ethernet,….

I.3.2.3 Niveau 3 : Couche Réseau :

- Elle gère l'acheminement des données en assurant le routage (choix du trajet) des paquets de données.
- Si un nœud est surchargé ou hors-service, les données seront alors routées vers un autre nœud.
- L'unité d'échange est le paquet.
- La couche réseau assure également la traduction des adresses logiques en adresses physiques.

EX :IP,… qu'on va développer ultérieurement.

I.3.2.4 Niveau 4 : Couche Transport :



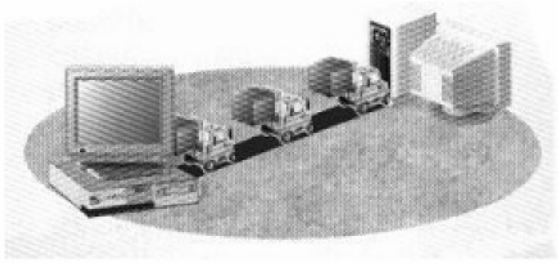
- Elle fournit un service de transport de bout en bout transparent pour l'utilisateur (même à travers plusieurs réseaux).

Ex : Etablissement, Maintien, Rupture, ...

- Elle assure également les services qui n'ont pas été pris en compte par les couches inférieures (gestion des erreurs, routage...).
- Elle permet de multiplexer plusieurs flux sur le même support
- En temps qu'émetteur, elle segmente les messages en paquets numérotés
- En temps que récepteur, elle reconstitue les messages en plaçant les paquets dans l'ordre

Ex : TCP, UDP,..

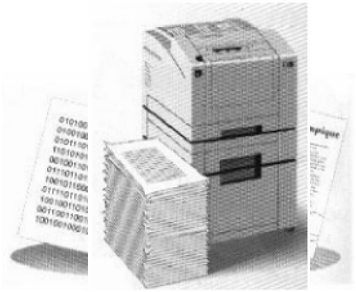
I.3.2.5 Niveau 5 : Couche Session



- C'est la première couche orientée traitement
- Elle permet l'ouverture et la fermeture d'une session de travail entre 2 systèmes distants et assure la synchronisation du dialogue.
- Elle définit le mode de transmission (Halfduplex, Full-duplex)
- Elle définit la liaison entre deux programmes d'application et gère le dialogue.

- Elle gère la chronologie du dialogue afin de mettre en place des procédures de reprise.

I.3.2.6 Niveau 6 : Couche Présentation



- Elle permet de transcrire les données dans un format compréhensible par les 2 systèmes (formatage des données).
- Elle assure la mise en forme de l'information pour qu'elle soit accessible à l'utilisateur
- Elle effectue les fonctions de codage, compression, cryptage et décryptage, ...

I.3.2.7 Niveau 7 : Couche Application

- Elle fournit des services utilisables sur le réseau par les applications installées.
- Les principaux services sont :
  - Transfert de fichiers (FTP)
  - Messagerie ou courrier électronique (pop, smtp)
  - Lecture de pages Internet (http)
  - Accès à distance (Telnet)

I.3.3 Le parcours des données dans le modèle OSI :

Lorsque les données sont transférées au sein d'un réseau :

- Elles parcourent chacune des couches du modèle OSI de l'émetteur (7-Application —> 1-Physique). Chaque fois qu'elles traversent une couche, elles sont enrichies de nouvelles informations : les informations délivrées par le protocole de la couche sont ajoutées (on parle d'encapsulation).
- Elles sont transmises sur le support.
- Elles parcourent chacune des couches du modèle OSI du récepteur (1 Physique —> 7- Application). Chaque fois qu'elles traversent une couche, les informations ajoutées par le protocole de même niveau de l'émetteur sont enlevées et exploitées (on parle de désencapsulation).

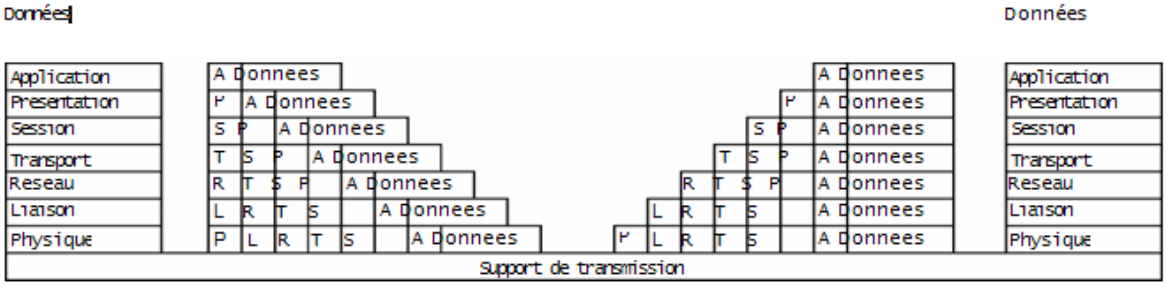


Figure **I.3.3 : Le parcours des données dans le modèle OSI**

I.4 le modèle applicatif TCP/IP :

TC P/IP désigne une architecture réseau, mais cet acronyme désigne enfait 2 protocoles étroitement liés : un protocole de transport, TCP (Transmission Control Protocol) qu'on utilise "par-dessus" un protocole réseau, IP (Internet Protocol). Ce qu'on entend par "modèle TCP/IP", c'est enfaite une architecture réseau en 4 couches dans laquelle les protocoles TCP et IP jouent un rôle prédominant, car ils en constituent l'implémentation la plus courante.

Par abus de langage, TCP/IP peut donc désigner deux choses : le modèle TCP/IP et la suite de deux protocoles TCP e t IP.

Le modèle TCP/IP, comme nous le verrons plus bas, s'est progressivement imposé comme modèle de référence en lieu et place du modèle OSI. Ce la tient tout simplement à son histoire. En effet, contrairement au modèle OSI, le modèle TCP/IP est né d'une implémentation ; la normalisation est venue ensuite.

✓ Description du modèle :

➢ Un modèle en 4 couches :

Le modèle TCP/IP peut en effet être décrit comme une architecture réseau à 4 couches :

- **La couche hôte réseau :**

Elle semble "regrouper" les couches physiques et liaison de données du modèle OSI. En fait, cette couche n'a pas vraiment été spécifiée ; la seule contrainte de cette couche, c'est de permettre un hôte d'envoyer des paquets IP sur le réseau. L'implémentation de cette couche est laissée libre. De manière plus concrète, cette implémentation est typique de la technologie utilisée sur le réseau local.

Par exemple, beaucoup de réseaux locaux utilisent Ethernet ; Ethernet est une implémentation de la couche hôte -réseau.

- **La couche internet :**

Cette couche est la clé de voûte de l'architecture. Cette couche réalise l'interconnexion des réseaux (hétérogènes) distants sans connexion. Son rôle est de permettre l'injection de paquets dans n'importe quel réseau et l'acheminement des ces paquets indépendamment les uns des autres jusqu'à destination. Comme aucune connexion n'est établie au préalable, les paquets peuvent arriver dans le désordre ; le contrôle de l'ordre de remise est éventuellement la tâche des couches supérieures.

Du fait du rôle imminent de cette couche dans l'acheminement des paquets, le point critique de cette couche est le **routage**. C'est en ce sens que l'on peut se permettre de comparer cette couche avec la couche réseau du modèle OSI.

La couche internet possède une implémentation officielle : le **protocole IP** (Internet Protocol).

Remarquons que le nom de la couche ("internet") , pour la simple et bonne raison que le mot internet est pris ici au sens large (littéralement, "interconnexion de réseaux "), même si l'Internet utilise cette couche .

- **La couche transport :**

Son rôle est le même que celui de la couche transport du modèle OSI : permettre à des entités soutenir une conversation.

Officiellement, cette couche n'a que deux implémentations : le **protocole TCP** (Transmission Control Protocol) et le **protocole UDP** (User Datagram Protocol). TCP est un protocole fiable, orienté connexion, qui permet l'acheminement sans erreur de paquets issus d'une machine d'un internet à une autre machine du même internet. Son rôle est de fragmenter le message à transmettre de manière à pouvoir le faire passer sur la couche internet. A l'inverse, sur la machine destination, TC P replace dans l'ordre les fragments transmis sur la couche internet pour reconstruire le message initial. TCP s'occupe également du contrôle de flux de la connexion.

UDP est en revanche un protocole plus simple que TCP : il est non fiable et sans connexion. Son utilisation présuppose que l'on n'a pas besoin ni du contrôle de flux, ni de la conservation de l'ordre de remise des paquets. Par exemple, on l'utilise lorsque la couche application se charge de la remise en ordre des messages.

On se souvient que dans le modèle OSI, plusieurs couches ont à charge la vérification de l'ordre de remise des messages. C'est là un avantage du modèle TCP/IP sur le modèle OSI. Une autre utilisation d'UDP : la transmission de la voix. En effet, l'inversion de 2 phonèmes ne gêne en rien la compréhension du message final. De manière plus générale, UDP intervient lorsque le temps de remise des paquets est prédominant.

- **La couche application :**

Contrairement au modèle OSI, c'est la couche immédiatement supérieure à la couche transport, tout simplement parce que les couches présentation et session sont apparues inutiles. On s'est en effet aperçu avec l'usage que les logiciels réseau n'utilisent que très rarement ces 2 couches, et finalement, le modèle OSI dépouillé de ces 2 couches ressemble fortement au modèle TCP/IP.

Cette couche contient tous les protocoles de haut niveau, comme par exemple Telnet, TFTP (trivial File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), HTTP (Hyper Text Transfer Protocol). Le point important pour cette couche est le choix du protocole de transport à utiliser. Par exemple, TFTP (surtout utilisé sur réseaux locaux) utilisera UDP, car on part du principe que les liaisons physiques sont suffisamment fiables et les temps de transmission suffisamment courts pour qu'il n'y ait pas d'inversion de paquets à l'arrivée. C e choix rend TFTP plus rapide que le protocole FTP qui utilise TCP. A l'inverse, SMTP utilise TCP, car pour la remise du courrier électronique, on veut que tous les messages parviennent intégralement et sans erreurs.

I.5. la couche réseau :

La couche réseau prend en charge l'optimisation des chemins de transmission entre les ordinateurs distants. Les paquets de données sont transmis grâce à l'établissement d'une

connexion logique entre les ordinateurs, qui peut comprendre plusieurs nœuds, L'adressage des ordinateurs est réalisé dans cette couche par des adresses logiques (par exemple des adresses IP) qui doivent être configurées sur chacun des ordinateurs.
Les protocoles chargés de la gestion de cette couche sont le protocole Internet Protocol (IP) de la famille TCP/IP.

I.5.1. IP version 4 :

I.5.1.1. Le format des adresses IP :

Les adresses IP sont composées de 4 octets. Par convention, on note ces adresses sous forme de 4 nombres décimaux de 0 à 255 séparés par des points.
L'originalité de ce format d'adressage réside dans l'association de l'identification du réseau avec l'identification de l'hôte.

- La partie réseau est commune à l'ensemble des hôtes d'un même réseau.
- La partie hôte est unique à l'intérieur d'un même réseau.

Prenons un exemple d'adresse IP pour en identifier les différentes parties : exemple adresse IP **192.168.1.1**

Adresse complète	**192.168. 1. 1**
Masque de réseau	255.255.255. 0
Partie réseau	192.168. 1.
Partie hôte	1
Adresse Réseau	192.168. 1. 0
Adresse de diffusion	192.168. 1.255

- **Le masque de réseau :**

Le masque de réseau sert à séparer les parties réseau et hôte d'une adresse. On retrouve l'adresse du réseau en effectuant un ET logique bit à bit entre une adresse complète et le masque de réseau.

- **L'adresse de diffusion :**

Chaque réseau possède une adresse particulière dite de *diffusion*. Tous les paquets avec cette adresse de destination sont traités par tous les hôtes du réseau local. Certaines informations telles que les annonces de service ou les messages d'alerte sont utiles à l'ensemble des hôtes du réseau.

- **Convention :**

- Une adresse dont tous les bits **hôte** sont à **0** est l'adresse d'un réseau.
- Une adresse dont tous les bits **hôte** sont à **1** est une adresse de **broadcast.**
- Si la partie **hôte** d'un réseau comporte **n** bits, on peut assigner **2n - 2** adresses.
- On assigne aux routeurs les adresses des passerelles.

I.5.1.2. Les classes d'adresses :

À l'origine, plusieurs groupes d'adresses ont été définis dans le but d'optimiser le cheminement (ou le *routage*) des paquets entre les différents réseaux. Ces groupes ont été

baptisés *classes d'adresses IP*. Ces classes correspondent à des regroupements en réseaux de même taille. Les réseaux de la même classe ont le même nombre d'hôtes maximum.

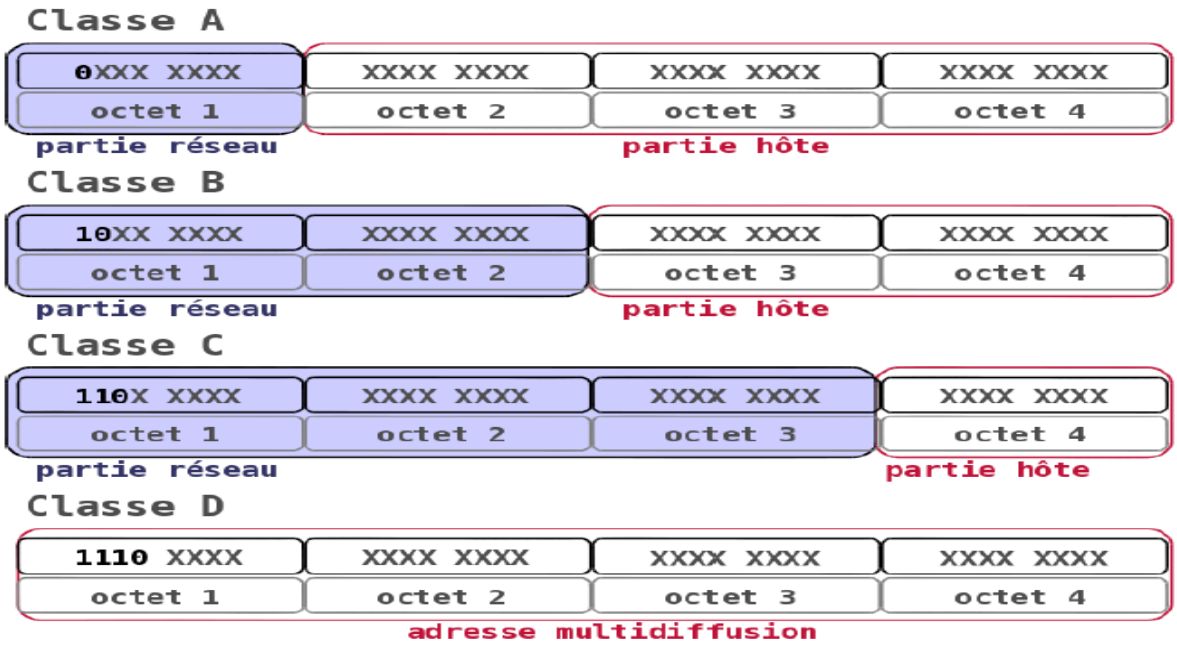


Figure **I.5.1.2. Les classes d'adresses**

- **Classe A**

Le premier octet a une valeur comprise entre 1 et 126 ; soit un bit de poids fort égal à 0. Ce premier octet désigne le numéro de réseau et les 3 autres correspondent à l'adresse de l'hôte.

- **Classe B**

Le premier octet a une valeur comprise entre 128 et 191 ; soit 2 bits de poids fort égaux à 10. Les 2 premiers octets désignent le numéro de réseau et les 2 autres correspondent à l'adresse de l'hôte.

- **Classe C**

Le premier octet a une valeur comprise entre 192 et 223 ; soit 3 bits de poids fort égaux à 110. Les 3 premiers octets désignent le numéro de réseau et le dernier correspond à l'adresse de l'hôte.

- **Classe D**

Le premier octet a une valeur comprise entre 224 et 239 ; soit 3 bits de poids fort égux à 111. Il s'agit d'une zone d'adresses dédiées aux services de multidiffusion vers des groupes d'hôtes (*host groups*).

- **Classe E**

Le premier octet a une valeur comprise entre 240 et 255. Il s'agit d'une zone d'adresses réservées aux expérimentations. Ces adresses ne doivent pas être utilisées pour adresser des hôtes ou des groupes d'hôtes.

I.5.1.3. En-tête de paquet IP :

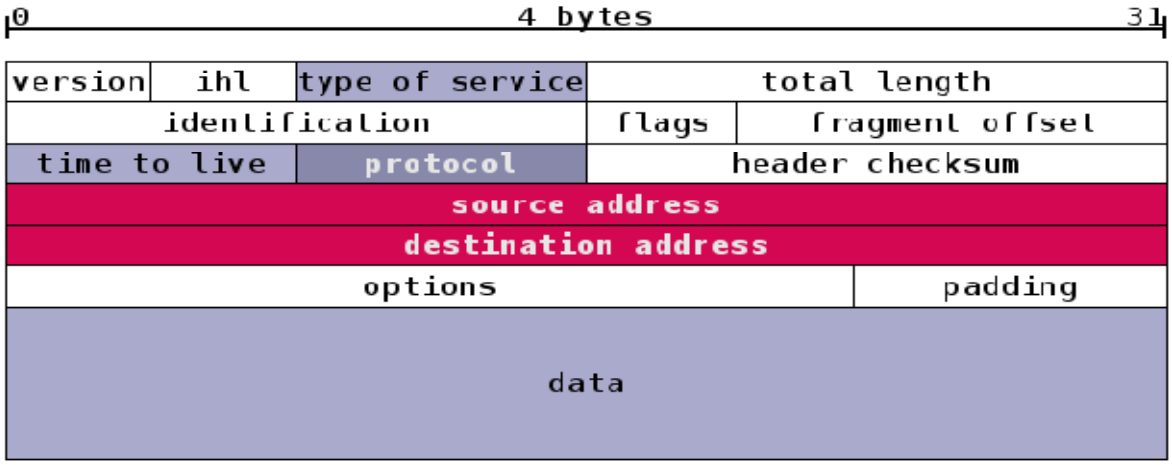


Figure **I.5.1.3 : En-tête de paquet IP**

***Version* :** 4 bits
Version du protocole IP codée sur 4 bits : 0100 pour IPv4 et 0110 pour IPv6.
Internet Header Length : 4 bits, IHL
Longueur de l'en-tête en mots de 32 bits. Cette valeur est utilisée pour distinguer la partie en-tête de la partie donnée du paquet. La représentation usuelle de l'en-tête se fait sur 32 bits de largeur. Comme les champs Options et Padding ne sont pas obligatoires, la valeur minimum du champ IHL est 5 (0101).
Type Of Service : 8 bits, TOS
Champ découpé en 2 parties. Les 3 premiers bits sont appelés *precedence* et les 5 derniers représentent le type de service. La définition d'origine prévoyait 3 choix : *low-delay*, *high-reliability* et *high-throughput*. Ce «marquage» des paquets est utilisable pour définir des flux prioritaires sur une interconnexion réseau «sous contrôle». Sur l'Internet, les opérateurs définissent leurs propres priorités ; donc leurs propres valeurs pour ce champ.
Total Length : 16 bits
Longueur du datagramme : en-tête & données. La taille minimum est de 21 octets (en-tête + 1 octet de donnée).
Comme se champ est représenté sur 16 bits, la taille maximum est de 2^16 - 1, soit 64 Ko.
Identification : 16 bits
Chaque paquet IP reçoit un numéro d'identification à sa création. Il est possible qu'un paquet soit découpé en *fragments* avant d'atteindre sa destination finale. Chaque fragment appartient au même paquet IP. Chaque fragment possède le même numéro d'identification.

Flags : 3 bits
Ce champ contient 3 indicateurs d'état :
• *Reserved flag* : doit toujours être à 0.
• *Don't Fragment* (DF) : à 0 si le paquet peut être fragmenté ; à 1 s'il ne doit pas être fragmenté.
• *More Fragments* (MF) : à 1 si d'autres fragments sont attendus ; à 0 s'il n'y a pas/plus de fragments.
Fragment Offset : 13 bits
Position du fragment dans le datagramme courant. Cette position est comptée en octets.
Time To Live : 8 bits, TTL
Ce compteur est décrémenté à chaque traversée de routeur. Si la valeur 0 est atteinte, le paquet est jeté. Cela signifie qu'il ne peut être délivré à sa destination finale. La valeur initiale du champ TTL dépend du système d'exploitation utilisé.
Protocol : 8 bits
Ce champ spécifie le protocole utilisé dans les données du paquet IP. Par exemple, la valeur 1 indique que le protocole utilisé est ICMP. On sait ainsi que ce paquet n'est pas destiné à une application.
Header Checksum : 16 bits
A chaque création ou modification d'un paquet, une somme de contrôle (*cyclic redundancy check*) est calculée sur son en-tête. Lorsque le paquet arrive à destination, cette somme est recalculée. Si le résultat diffère, c'est que le paquet a été endommagé lors de son trajet.
Source Address : 32 bits
Adresse IP de l'hôte qui a émis le paquet.
Destination Address : 32 bits
Adresse IP de l'hôte qui doit recevoir le paquet.
Options and Padding
Cette partie de l'en-tête est optionnelle. Ce champ est utilisé pour fournir des instructions spécifiques de distribution du paquet qui ne sont pas couvertes par les autres champs de l'en-tête. La taille maximum de ces instructions est limitée à 40 octets regroupés en double-mots de 32 bits. Les bits de *padding* servent à compléter le dernier double mot de 32 bits.
Data
C'est le dernier champ du paquet IP. Il contient les «données» du paquet. Celles ci peuvent débuter par un en-tête de couche transport (4) qui donnera d'autres instructions à l'application qui recevra les données. Le champ *Data* peut aussi contenir un message ICMP qui ne contient aucune donnée utilisateur.

I.5.2. IP version 6 (IPng, ou IP new generation) :

I.5.2.1. pourquoi un nouveau protocole IP ?

Lors de la conception d'IP (Internet Protocole) en 1978, les ingénieurs pensaient que seuls quelques milliers d'ordinateurs seraient concernés répartis sur une douzaine de réseaux. Or tout le monde sait, aujourd'hui que ce n'est pas le cas.

Avec IPv4 (Internet Protocole version 4), l'adressage se fait sur 32 bits, ce qui serait suffisant comme espace, mais, IPv4 est un protocole qui est trop restreint de par son utilisation et qui est donc coûteux en termes d'adresses gaspillées.
Bientôt, étant donné l'expansion de l'Internet, il n'existera plus d'adresse disponible sous Ipv4. Certains spécialistes pronostiquent la pénurie d'adressage sous Ipv4 d'ici 2008-2010.
Par exemple :
* Les adresses de classe A : elles représentent douze réseaux de 16.7 millions de nœuds. Toutes les adresses de classe A sont déjà toutes épuisées.
* Les adresses de classe B : elles représentent 16368 réseaux de 65534 nœuds. Ce sont les adresses les plus répandues parmi les industriels et certains fournisseurs d'accès. Elles sont déjà presque toutes utilisées.
* Les adresses de classe C : elles représentent 2 millions de réseaux de 254 nœuds. Elles sont principalement pour les petites organisations, et, actuellement, elles sont distribuées aux fournisseurs d'accès. Ces adresses sont déjà épuisées.
De plus, aujourd'hui, le nombre de réseaux connectés est devenu très important, les tables de routage ont pris des proportions considérables, donnant ainsi une charge de travail énorme aux administrateurs.
De par la taille des tables de routage, le traitement des paquets est fortement ralenti. Actuellement, les routeurs principaux des infrastructures de l'Internet comptent environ 7000 routes.
Le nouveau protocole, IPv6, doit permettre un adressage plus grand et un routage plus simple et plus rapide.

I.5.2.2. Adressage IPv6 :

L'adressage sous IPv6 se fait désormais sur 128 bits (32 bits sur IPv4).
Sous IPv6 on n'a plus de notion de classes, on a seulement un adressage hiérarchique, par préfixe. L'adressage se fait désormais par l'attribution d'une partie d'adresse fixe qui définit le réseau, cette partie est appelée le préfixe.
Cet adressage hiérarchique consiste en l'attribution d'un préfixe au premier routeur d'un réseau, sur 10 bits par exemple. Le routeur du sous réseaux aura lui une partie fixe qui sera celle de son père plus un complément de préfixe qui lui sera imposé. Et ainsi de suie l'attribution des adresses se fait de manière hiérarchique.

IPv6 reconnaît trois types d'adressage :

- **Adresse UNICAST** : Le type unicast, est le plus simple. Une adresse de ce type désigne une interface unique. Un paquet envoyé à une telle adresse sera donc remis à l'interface ainsi identifiée.
- **Adresse MULTICAST** : ce sont les successeurs des adresses broadcast (envoi à un ensemble de machines qui se doivent d'appartenir à une même classe). Une adresse de type multicast désigne un groupe d'interfaces appartenant, en général, à des équipements différents pouvant être situés n'importe où dans l'Internet. Lorsqu'un paquet a pour destination une adresse de type multicast, il est acheminé par le réseau à toutes les interfaces membres de ce groupe.
- **Adresse ANYCAST** : Ce type d'adresse est nouveau en IPv6. Comme dans le cas du multicast, une adresse de type anycast désigne un groupe d'interfaces, la différence étant que,

lorsqu'un paquet a pour destination une adresse de type anycast, il est routé à un seul des éléments du groupe et non pas à tous. Ce sera, par exemple, le plus proche au sens de la métrique des protocoles de routage. Ce type d'adresse est encore en cours d'expérimentation et est réservé pour le moment aux routeurs.

Les adresses anycast ont deux points communs avec les adresses unicast : elles sont allouées dans le même espace d'adressage et ont les mêmes formats.

I.5.2.2.1. Représentation des adresses :

Une adresse IPv4 est un mot de 32 bits tandis qu'une adresse IPv6 est un mot de 128 bits. La taille des adresses a donc été quadruplée, ce qui permet d'obtenir un espace adressable en IPv6 nettement plus large que celui en IPv4.

Une adresse sur IPv6 est un ensemble de 8 mots de 2 octets, qui sont en fait, 8 groupes de 4 lettres hexadécimales séparés par « : ».

(Ex : FEDC:BA98:7654:3210:FEDC:BA98:7654:3210).

Dans un champ, il n'est pas nécessaire d'écrire les zéros placés en tête. En outre plusieurs champs nuls consécutifs peuvent être abrégés par «::». Ainsi les deux notations suivantes sont équivalentes :

FEDC:0000:0000:0000:0400:A987:0043:210F

FEDC::400:A987:43:210F

Plus particulièrement, l'adresse formée uniquement par des zéros est représentée comme suit :

::

Naturellement, pour éviter toute ambiguïté, l'abréviation «::» ne peut apparaître qu'une fois au plus dans une adresse.

La représentation des préfixes IPv6 est similaire à la notation CIDR utilisée pour les préfixes IPv4. Un préfixe IPv6 est donc représenté par la notation :

adresse-ipv6 / longueur-du-préfixe -en-bits.

Les formes abrégées avec «::» sont autorisées :

3EDC:BA98:7654:3210:0000:0000:0000:0000/64

3EDC:BA98:7654:3210:0:0:0:0/64

3EDC:BA98:7654:3210::/64

::/0 (défaut)

Enfin on peut combiner l'adresse d'une interface et la longueur du préfixe réseau associé en une seule notation : 3EDC:BA98:7654:3210:945:1321:ABA8:F4E2/64

D'autres types d'adresses

· *Adresse indéterminée* : l'adresse indéterminée est utilisée comme adresse source par un équipement du réseau pendant son initialisation, avant d'acquérir une adresse. Sa valeur est 0:0:0:0:0:0:0:0 (ou en abrégé ::).

· *Adresse de bouclage (loopback address)* : L'adresse de bouclage vaut 0:0:0:0:0:0:0:1, soit en abrégé ::1. Il s'agit de l'équivalent de l'adresse 127.0.0.1 d'IPv4. Elle est utilisée par un équipement du réseau pour envoyer un paquet IPv6 à lui-même.

I.5.2.2.2. Structure des paquets IPv6 :

Le protocole IPv6 apporte une simplification de l'en-tête. Les champs sont désormais moins nombreux, sept champs au lieu de quatorze sous Ipv4. Cet allègement de l'en-tête permet une meilleure efficacité de commutation des équipements de routage qui ont moins de données à dépiler pour effectuer le routage.

On peut remarquer la disparition du « cheksum » qui fixait la taille de l'en-tête, ainsi que la disparition des en-têtes optionnels dans l'en-tête lui-même.

- **Format du paquet IPv6 :**

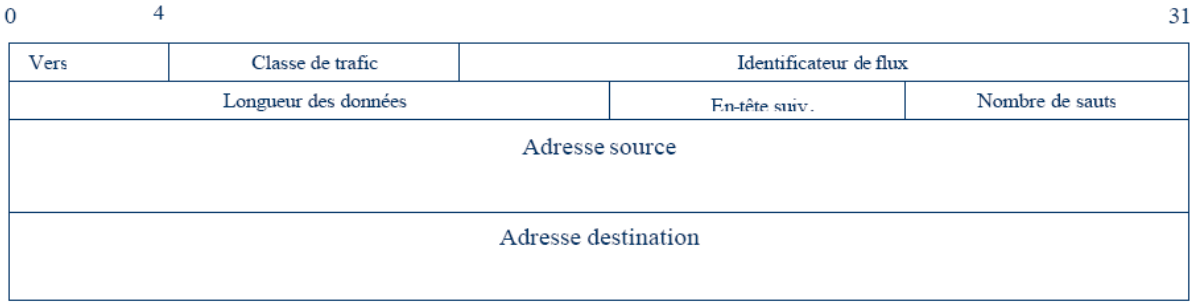
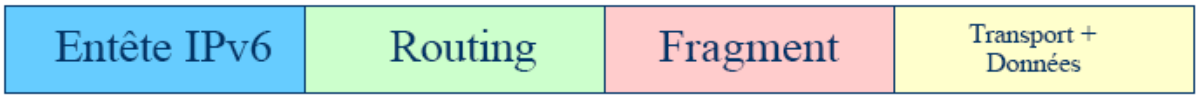


Figure **I.5.2.2.2 : Format du paquet IPv6**

Version : Seul champ inchangé par rapport à la version 4, contient la version IP du paquet. Seule modification, sur IPv6, sa valeur est 6.
Classe de trafic : Nature du trafic. Permet d'offrir un niveau de priorité aux paquets.
Identificateur de flux : Ce champ permet la mise en œuvre des fonctions de qualités de service. Il permet d'optimiser le routage par un acheminement plus rapide des données. Par ce champ, on peut donner un identifiant à la communication. Selon sa valeur, les routeurs du chemin reconnaissent la connexion et ne dépilent pas les informations, ils les transmettent directement.
Longueur des données : Longueur des paquets sans l'en-tête (en octets).
En-tête suivant : Ce champ indique le prochain entête dans le datagramme IPv6, c'est-à-dire l'emplacement des en-têtes optionnels si ils existent.
Nombre de sauts : Ce champ remplace le champ « TTL » d'IPv4. Sa valeur, sur 8 bits, est décrémentée à chaque traversée d'un routeur. Si sa valeur atteint la valeur 0, le paquet est détruit et un message d'erreur est émis par ICMPv6.

- **Entêtes optionnels :**

Le paquet IPv6 inclut un champ d'extension pour les fonctionnalités optionnelles (sécurité, source routing, ...). Les options de IPv6 sont placées dans des en-têtes séparés, intercalés entre l'en-tête IPv6 et l'en-tête de la couche transport.



Les différents entêtes optionnels sont :
- *Entête Hop-by-Hop* : Information examinée sur chaque nœud du chemin.
- *Entête End-to-end* : Information examinée par le destinataire uniquement.
- *Entête Routing* : Routage effectué par la source. Liste des nœuds intermédiaires "à visiter", s'ils existent.
- *Entête Fragment* : envoi de paquets plus long que le MTU. La fragmentation réalisée par la source uniquement.
- *Entête authentification et intégrité des données*
- *Entête Privacy* : encryptage des données à protéger. Un des aspects de la sécurité IPv6.

- **Fragmentation**

Alors que dans le cas du protocole IPv4 tous les routeurs pouvaient fragmenter les datagrammes, pour IPv6, ce n'est plus le cas ; Seule la source a le droit de fragmenter et seule la destination celui de défragmenter (fragmentation de bout en bout). S'il est nécessaire de fragmenter, la source insère un petit en-tête d'extension après l'en-tête de base de chaque fragment.
Le but de la fragmentation de bout en bout est de réduire les frais de gestion de la fragmentation dans les routeurs et permettre ainsi à chaque routeur de traiter plus de datagrammes par unité de temps. Une des conséquences est que si un routeur tombe en panne, il est difficile de changer le chemin car cela peut changer la MTU du chemin. Lorsqu'un protocole utilise la fragmentation de bout en bout, la source doit faire une recherche de MTU minimum tout au long du chemin et fragmenter tous datagrammes sortant inférieurs au MTU. La fragmentation de bout en bout s'accommode (adapter) mal des modifications de chemin.

I.5.2.3. ICMPv6 :

Le protocole de contrôle IP a été revu. Dans IPv4, ICMP (Internet Message Control Protocol) sert à la détection d'erreurs (par exemple : équipement inaccessible, durée de vie expirée…), aux tests et à la configuration automatique des équipements (redirection ICMP, découverte des routeurs). Ces trois fonctions sont mieux définies dans IPv6.

- **Champs d'un paquet ICMPv6 :**

Figure **I.5.2.3 : Champs d'un paquet ICMPv6**

Type : nature du message (0-127 : erreur, 128-255 : messages d'information, utilisés par exemple pour l'auto configuration)
Code : cause du message ICMPv6
Checksum : permet de vérifie r l'intégrité du paquet
Données : peuvent contenir un compte rendu de ce qui a provoqué l'erreur.

- **Fonctions intégrées :**

MLD

Grâce au sous-protocole MLD (1), ICMPv6 intègre les fonctions de gestion des groupes de multicast qui sont effectuées par IGMP (2) dans IPv4.
(1) MLD (Multicast Listener Discovery) est identique à IGMP[1].
(2) IGMP (Internet Group Management Protocol) Protocole de gestion des groupes multicast en réseau IP local.

ARP

ARP (Address Resolution Protocol) est un protocole de résolution d'adresse IPv4 en une adresse de niveau liaison. Il permet de générer une table des correspondances *adresse physique* ⇔ *adresse logique*.
ICMPv6 reprend les fonctions de ce protocole.

- **Découverte de la MTU :**

La MTU (Maximum Transmission Unit) est la taille maximum qu'un paquet peut avoir pour être acheminé sur un réseau, pour des raisons logicielles ou matérielles.
Il s'agit d'une opération qui permet de maintenir la valeur des MTU des différents chemins utilisés par un nœud. Tout au long de sa vie sur le réseau, chaque nœud maintient une base de données de cette information.
Pour détecter la MTU du réseau sur lequel elle se trouve, une machine émet des paquets de demande MTU à tous ses voisins.

IPV6 conseille l'utilisation d'une MTU de 1280 octets.

- **Messages d'erreur :**

ICMPv6 permet, aux différentes machines, d'émettre des messages d'erreur :

- *Fragmentation* (paquet trop gros) : La fragmentation à la demande des paquets IP augmente les risques de congestion du réseau et ralentit la transmission dans IPv4. Dans le cas de IPv6, si un paquet dépasse la MTU d'un réseau, le paquet est détruit, et un paquet ICMPv6 est envoyé à l'initiateur du paquet, qui va redéfinir la taille de tous les paquets qu'il va envoyer.

[1] Le protocole qui gère les groupes d'équipements multicast dans un réseau local.

- *Destination inaccessible*
- *Temps dépassé*
- *Entête invalide*

- **Messages d'information**

ICMPv6 permet aux différentes machines d'échanger des informations, comme pour l'auto configuration par exemple. Ces messages sont de deux types :

- *Message requête*
- *Message réponse*

I.5.2.4. Auto-configuration

Le besoin de simplifier le processus de configuration des machines se fait ressentir. Cela inclut les services DHCP comme les interactions avec les voisins. La configuration automatique est un atout principal d'IPv6.

La configuration automatique signifie qu'une machine obtient toutes les informations nécessaires à sa connexion à un réseau local IP sans aucune intervention humaine. Le protocole IPv6 introduit la notion de « Plug & Play » dans les réseaux.

- **Objectifs**

Les objectifs de l'auto configuration sous IPv6 sont

- L'acquisition d'une adresse quand une machine est attachée à un réseau pour la première fois.
- L'obtention d'une nouvelle adresse en cas de renumérotation des machines du site (changement de la partie haute de l'adresse)
- L'obtention d'une nouvelle adresse en cas de déplacement.

- **ID d'interface**

L'ID d'interface est une nouvelle notion introduite avec IPv6. L'ID occupe les 64 bits de poids faible de l'adresse. Dans un réseau local basé sur Ethernet, il est construit grâce aux 48 bits du nombre MAC des cartes réseau dans lesquels on insère par octets des valeurs constantes pour obtenir 64 bits.

Exemple : Une carte réseau de numéro MAC 00:E0:4C:39:B2:A9 donnera un ID 02E0:4CFF:FF39:B2A9

- **Méthodes d'auto configuration**

Adresse lien-local :

Les adresses lien-local sont des adresses dont la portée est restreinte à un site donné. Par exemple, un site qui n'est pas encore connecté à Internet peut utiliser ce type d'adressage, et sera dispensé d'emprunter un préfixe.

L’adresse lien-local est créée en prenant le préfixe FE80::/64 auquel on ajoute les 64 bits d’ID d’interface. L’adresse constituée est encore interdite d’usage. La machine doit encore vérifier l’unicité de cette adresse sur le réseau par le protocole de détection d’adresse dupliquée. Si la machine détermine que sa création d’adresse lien-local a échoué, alors une intervention manuelle est nécessaire.

Sinon, l’adresse provisoire devient définitive.

Auto configuration sans état :

Elle est utilisée dans un réseau connecté par routeur à un autre réseau (Internet par exemple), et quand la gestion stricte des adresses attribuées n’est pas nécessaire au sein d’un site. Cette méthode décentralisée permet à chaque machine du site de construire sa propre adresse IPv6. Elle ne demande ni une configuration particulière des machines ni de serveur supplémentaire. Elle se sert du protocole ICMPv6. Une machine construit son adresse IPv6 à partir d’informations locales et d’informations fournies par le routeur. Le routeur lui donne le préfixe (par un message d’annonce de routeur), puis elle construit son adresse comme vu précédemment.

Auto configuration avec état (DHCPv6) :

La plus complète car permet de configurer l’adresse du client, le nom de domaine, le serveur de nom, etc. Elle est retenue lorsqu’un site demande un contrôle strict de l’attribution des adresses. Ceci signifie que toute attribution d’une adresse IPv6 globale doit passer par un serveur DHCPv6 du site. Le routeur joue alors un rôle important : il dicte à la machine la méthode à retenir et fournit éventuellement les informations nécessaires à sa configuration.

Actuellement, le protocole DHCPv6 n’est pas encore standardisé.

- **Utilisation de routines ICMPv6**

L’auto configuration est effectuée grâce à un ensemble de routines ICMPv6 :

- Découverte des routeurs du réseau.
- Découverte des préfixes imposés par les routeurs.
- Détection des adresses dupliquées.
- Découverte des paramètres (dans le cas d’une auto configuration avec état).

I.5.2.5. Routage

- **SPF**

SPF (Shortest Path First) est une technique de routage basée sur le plus court chemin. Ce protocole ne subit pas de modifications dans IPv6.

- **OSPFv6**

Le protocole OSPFv6 (Open Shortest Path First) cherche à atteindre plusieurs objectifs dont :

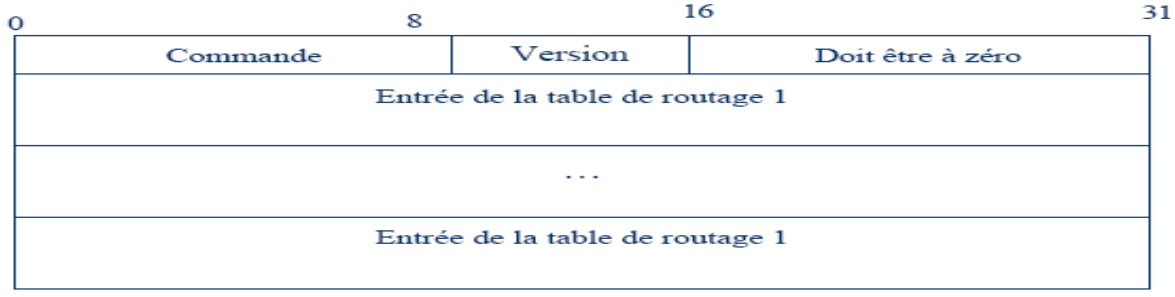
- *Routage par type de service* : les administrateurs peuvent définir plusieurs routes, de qualité de service différente, vers une destination donnée. Le routeur choisit alors la route de QoS adéquate pour acheminer les paquets.
- *Equilibrage des charges* : si un administrateur définit plusieurs routes de même QoS vers une destination donnée, OSPF répartit équitablement le trafic sur toutes ces routes.
- *OSPF* permet à un site de décomposer ses réseaux et routeurs en sous-ensembles appelés *zones*. Chaque zone est autonome et la topologie d'une zone reste invisible pour les autres zones.

- **Spécificités**

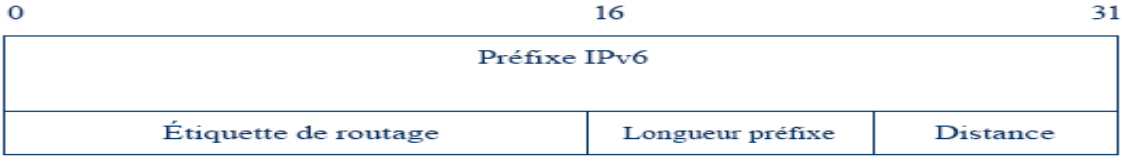
Le routage sous IPv6 inclut les extensions suivantes
- Le routage se fait de manière classique, sur les préfixes (attribution de plages d'adresses)
- Le routage peut se faire intégralement par la source (avec un en-tête optionnel « Routing »)

- **RIPng**

RIP (Routing Information Protocol) : Chaque routeur propage toutes les routes qu'il connaît vers les autres routeurs. Chaque routeur possède une table de routage qui comporte une entrée pour chaque destination possible Dans RIPng, un mécanisme de temporisation permet de gérer l'ensemble des évènements. Ainsi toutes les 30 secondes, le processus de routage est réveillé afin d'envoyer un message de type *réponse* contenant la table de routage complète. Ce message est envoyé à tous ses voisins.



Message RIPng



Format du champ "Entrée"

I.5.2.6. Plan de transition d'IPv4 vers IPv6 :

- **Plan de transition IPv4 vers IPv6**

La transition d'IPv4 vers IPv6 ne se fera pas du jour au lendemain. On prévoit qu'elle s'effectuera d'ici 2012. Elle doit se faire de manière progressive et doit permettre, durant la période de transition, la coexistence des protocoles IPv4 et IPv6. L'objectif principal est de terminer le passage à Ipv6 avant l'épuisement total des adresses Ipv4.

La transition d'IPv4 vers IPv6 peut se faire en 3 phases

1. Seuls les équipements IPv4 existent. On arrive en fin de cette phase. De nombreux constructeurs vont proposer dans un délai très court les premières versions d'IPv6 pour les postes de travail et les routeurs

2. Phase de coexistence. Phase qui sera très longue. Les machines devront conserver les adresses Ipv4 déjà allouées.

3. Phase où seuls subsisteront les équipements IPv6.

- **IPv6 techniques de transition**

Les machines actuelles et à venir devront être capables de traiter des paquets IPv6 autant que les paquets IPv4. Pour transiter de la version 4 à la version 6 d'IP, trois techniques ont été mises au point : la double pile IP, l'encapsulation d'IPv6 dans Ipv4 (tunelling), et la traduction des en-têtes IPv6 en en-têtes IPv4 (voire l'inverse).

Le modèle Dual Stack[2]

Les équipements ont une adresse dans chacun des plans d'adressage Ipv4 et Ipv6. Ils acheminent aussi bien les paquets Ipv6 qu'Ipv4.

La majeure partie du code de la pile IPv4 peut être réutilisée pour IPv6. En effet, un seul branchement est nécessaire pour distinguer le bon code, en regardant le premier champ de l'entête IP qui donne le numéro de version.

Les tunnels[3]

L'encapsulation consiste à faire transiter des données d'un protocole donné à l'intérieur d'un autre. A l'heure actuelle, tous les routeurs ne sont pas capables de router des paquets IPv6. On place alors un paquet IPv6 à l'intérieur d'un paquet IPv4 pour le faire passer dans les réseaux anciens, et pouvoir retrouver un paquet IPv6 en sortie. On crée ainsi des tunnels Ipv6 à travers une infrastructure Ipv4. Cette méthode conserve les problèmes inhérents à IPv4 (adressage, routage, fragmentation) quoiqu'il arrive.

I.6. IPv6 vs IPv4 (comparaison) :

La différence principale entre IPv4 et IPv6 est le nombre d'adresses IP. Alors qu'il n'existe qu'un peu plus de 4 000 000 000 d'adresses IPv4, il y a plus de 340 000 000 000 000 000 000 000 000 000 000 adresses IPv6.

Le fonctionnement technique d'Internet reste le même, quelle que soit la version du protocole, et il est fort probable que les deux versions continueront à fonctionner simultanément sur les réseaux pour de nombreuses années à venir. De nos jours, la plupart des réseaux qui utilisent IPv6 supportent les adresses IPv4 aussi bien que les adresses IPv6 sur leurs réseaux.

[2] C'est une architecture réseau qui permet d'acheminer des paquets Ipv4 et Ipv6.

[3] Connecter des machines Ipv6 distantes à un réseau MPLS Ipv4.

	Version 4 du protocole Internet (IPv4)	Version 6 du protocole Internet (IPv6)
Déploiement	1981	1999
Espace d'adresse	Numéro de 32 bits	Numéro de 128 bits
Format d'adresse	Notation décimale à point : 192.149.252.76	Notation hexadécimale : 3FFE :F200:0234:AB00:0123:456 7:8901:ABCD
Notation préfixée	192.149.252.76	3FFE:F200:0234:AB00:0123: 4567:8901:ABCD
Nombre d'adresses	2^{32} = ~4 000 000 000	2^{128} = ~340 000 000 000 000 000 000 000 000 000 000 000 000

La figure ci-dessous représente les différents de points de peering dans le monde, soit les d'échange entre plusieurs fournisseurs d'accès. Le tiers Nord Est représente l'Europe, le tiers Nord Ouest L'Asie et l'Océanie et enfin le Sud, L'Amérique (de Gauche à Droite : Amérique du Nord à Amérique du Sud).

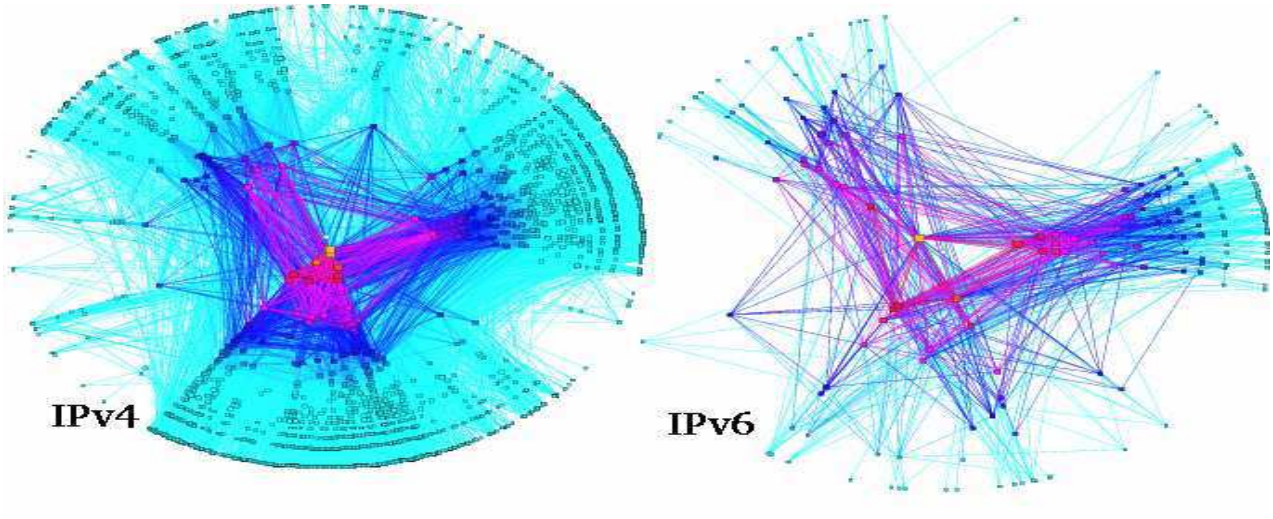


Figure **I.6 : IPv6 vs IPv4**

I.7. conclusion :

Le manque d'adresses étant de plus en plus menaçant, passer à IPv6 devient une nécessité. Ajouté à cela, IPv6 apporte de nouveaux services tels que la sécurité ou encore l'auto-configuration. Sa structure modulable et évolutive lui permet d'être plus performant qu'IPv4 en termes de rapidité de routage, mais aussi beaucoup plus ouvert à de nouvelles technologies.

Aujourd'hui IPv6 fait figure d'îlots au milieu de l'océan IPv4. Toutefois, la tendance évolue lentement, et sera surement inversé dans quelques années. Pour réaliser cette migration globale, les acteurs s'appuis sur les solutions de cohabitation. L'un des risques est que chacun se repose sur ces solutions et que la situation actuelle se fige. En effet, les solutions de cohabitation doivent rester transitoires, et les acteurs se doivent de concentrer leurs efforts autour de la nouvelle version d'IP.

On constate d’un point de vu mondial que certains pays sont très avancés, notamment en Asie. L’Europe reste cependant le lieu où il existe le plus de connexions IPv6, les réseaux de recherche européens y étant probablement pour beaucoup.

Voilà plus de 10 ans qu’IPv6 existe, et son déploiement n’en est parfois qu’à ses débuts. Une fois la norme adopté par tous les acteurs, il ne serait pas étonnant de voir de nombreux appareils trouver un usage à IPv6, notamment dans le domaine de la domotique.

Nous terminerons ce premier chapitre par cette analyse de Gautier Harmel :

"Le sens de l'histoire, c'est la convergence de tous les moyens de communication possibles par tous les terminaux possibles. Et pour que chacun puisse communiquer, il faut quelque chose qui fédère ces différents environnements. IP remplit ce rôle. Il permettra demain à un PC portable, à un réfrigérateur et à un téléphone, de dialoguer ensemble".

Chapitre II : La qualité de service dans les réseaux IP

II.1 introduction :

À ses débuts, Internet avait pour seul objectif de transmettre les paquets à leur destination. Conçu pour le transport asynchrone des données, IP (*Internet Protocol*) n'a pas été prévu pour les applications en temps réel comme la téléphonie ou la vidéo, très contraignantes. Le besoin en équipements de plus en plus fiables, d'un bout à l'autre du réseau, est donc devenu incontournable.

Cependant, les défauts rencontrés sur les réseaux (perte de paquets, congestion) ne peuvent pas être surmontés sans une rénovation profonde de l'architecture.

La qualité de service est la méthode permettant de garantir à un trafic de données, quelle que soit sa nature, les meilleures conditions d'acheminement répondant à des exigences prédéfinies. Elles fixent notamment des règles de priorité entre les différents flux.

La maîtrise de la qualité de service est un enjeu essentiel. La qualité de service doit être visualisée et mesurée de bout en bout. Le contexte joue un rôle crucial dans l'appréciation des paramètres de la qualité de service qu'il faut adapter au besoin de l'entreprise.

Dans ce chapitre, on va évoquer le problème de la qualité de service dans le réseau IP tout en détaillant les paramètres de performances du réseau afin de fournir un service meilleur et plus prévisible en terme de : débit, délai de latence, variation de délai ou gigue et taux de pertes de paquet.

II.2 Définition :

La qualité de service est un ensemble de caractéristiques de performance de service qui sont perçues et exprimées par l'utilisateur. Elle se manifeste par des paramètres pouvant prendre des valeurs qualitatives, c'est à dire qui ne peuvent pas être mesurées directement mais perceptibles par l'utilisateur ou bien se traduit par des valeurs quantitatives qui sont directement observées et mesurées aux points d'accès.

Comme on peut la définir étant « *l'ensemble des phénomènes pouvant influencer les performances du service qui détermine le degré de satisfaction de l'utilisateur de ce service* ». La QoS est appréhendée différemment selon l'acteur : client, opérateur de service ou de réseau et le rôle : demandeur, fournisseur, consommateur. Si elle se résume souvent à l'écoulement du trafic dans un réseau, beaucoup d'autres facteurs sont à prendre en compte. Ainsi, de nombreux facteurs comme la perte d'alimentation, la surcharge ou la panne de plate-forme de service, la rupture d'un lien de transmission, ... sont plus souvent la cause d'une mauvaise qualité perçue par l'utilisateur qu'une déficience du réseau au niveau de l'acheminement des données. Un service ou une application offrant une excellente qualité de service se doit d'être

fiable, robuste et tolérant aux pannes avant même de se préoccuper du bon acheminement des données.

II.3 Paramètres de qualité de service :

La maîtrise de la qualité de service est un enjeu essentiel. Elle doit être visualisée et mesurée de bout en bout. Le contexte joue un rôle crucial dans l'appréciation des paramètres de la qualité de service qu'il faut adapter aux besoins de l'entreprise.

Il y a cinq paramètres techniques à prendre en compte dans la qualité de service qui sont :

II.3.1 la disponibilité du réseau :

La disponibilité d'un réseau se définit comme le rapport entre le temps de bon fonctionnement du service et le temps total d'ouverture du service. C'est la forme la plus évidente de QoS, puisqu'elle représente la possibilité d'utiliser le réseau.

II.3.2 Débit:

Le débit maximum ou la bande passante est considérée comme étant le taux de transfert maximum pouvant être maintenu entre deux points terminaux. La QoS ne génère pas la bande passante. En revanche, ses mécanismes permettent de gérer de façon optimale la bande passante du réseau en fonction des demandes des applications. Dans les réseaux à commutation de paquets, la garantie de bande passante est un paramètre de performance très important pour garantir la QoS aux flux temps-réel. Ces derniers, ont une exigence minimale en terme de bande passante égale à leur débit moyen.

II.3.3 temps de réponse :

Il s'agit du temps d'attente pour mesurer le temps écoulé pour la transmission des paquets IP.

La maîtrise du délai de transmission est un élément essentiel pour bénéficier d'un véritable mode de transmission. Or la durée de traversée d'un réseau IP dépend de nombreux facteurs:

- Le débit de transmission sur chaque lien ;
- Le nombre d'éléments réseaux traverses.

Les éléments d'infrastructure, notamment les routeurs, peuvent également mettre en œuvre des buffers de gigue.

Les chiffres suivants (tirés de la recommandation UIT-T G114) sont donnés à titre indicatif pour préciser les classes de qualité et d'interactivité en fonction du retard de

transmission dans une conversation téléphonique. Ces chiffres concernent le délai total de traitement, et pas uniquement le temps de transmission de l'information sur le réseau.

Classe n°	Délai par sens	Commentaires
1	0 à 150 ms	Acceptable pour la plupart des conversations
2	150 à 300 ms	Acceptable pour des communications faiblement interactives
3	300 à 700 ms	Devient pratiquement une communication half duplex
4	Au delà de 700 ms	Inutilisable sans une bonne pratique de la conversation half duplex

Figure II.3.3 : Classes de service

II.3.4 Variation des délais de traversée (gigue ou jitter) :

La gigue se définit comme la variation des délais d'acheminement (latence) des paquets sur le réseau. Ce paramètre est particulièrement sensible pour les applications multimédias qui requièrent un délai inter paquet relativement stable. Il dépend principalement du type et du volume de trafic sur le réseau et du type et du nombre d'équipements réseau. La Figure II.3.4 montre une distribution typique du délai et illustre le concept de la gigue.

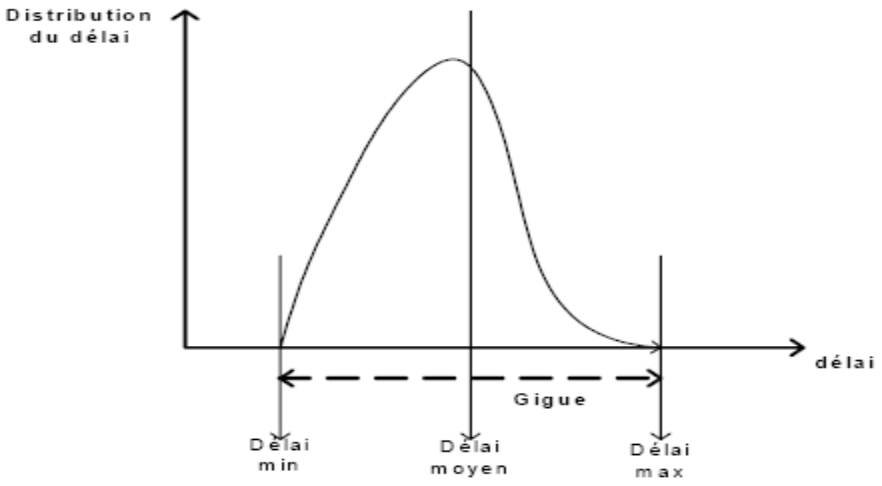


Figure II.3.4 : **Distribution de délai et illustration de la gigue**

Les flux temps-réel tels que les flux vidéo, audio et voix sur IP sont très sensibles à la gigue. En effet, le non considération de cette métrique implique une discontinuité au niveau de la restitution des données à la destination. Par exemple, la variation de délai dans une transmission vidéo pourrait entraîner des images saccadées aléatoirement ce qui dégrade considérablement la qualité de service de l'application. Pour réduire l'effet de la gigue, les paquets temps-réel sont bufférisés à la destination avant leur lecture.

Pour cela, la gigue doit être bornée pour prévoir la taille du tampon qui dépend principalement de la valeur de la gigue et du débit de réception des données.

II.3.5 Taux de perte de paquets :

Lorsque les buffers des différents éléments réseaux IP sont congestionnés, ils essaient automatiquement de libérer de la bande passante en se débarrassant d'une certaine proportion des paquets entrants, en fonction de seuils prédéfinis. Cela permet également d'envoyer un signal implicite aux terminaux TCP qui diminuent d'autant leur débit au vu des acquittements négatifs émis par le destinataire qui ne reçoit plus les paquets.

Le taux de pertes = nombre de paquets non arrivés / le nombre total de paquets transmis.

Les pertes sur IP sont causées par la congestion, l'instabilité du routage, les défaillances de liens. La congestion est la cause la plus importante de pertes. La perte de paquet peut se produire soit par dépassement de capacité des buffers dans les routeurs ou dans les systèmes d'extrémité soit par violation de délai borné. La distribution des pertes est aussi une métrique très importante pour les protocoles adaptatifs tels que TCP. Un lien réseau peut être caractérisé par son taux d'erreur **e** qui est calculé par intervalle de temps relativement long.

e = nombre de bits reçus erronés / le nombre total de bit reçus

Ou

e = nombre de paquet erronés / le nombre total de paquets reçus.

II.4 Les classes de services :

Ces paramètres sont ensuite regroupés entre eux en fonction des besoins des applications et des services. Ces groupes forment alors des **Classes de Services** (Class of Services : CoS).

Les requêtes de QoS des applications ou des services seront toujours affectées à une classe de service donnée. A chaque classe, correspond un ensemble de paramètres de QoS avec des objectifs quantifiés. Plusieurs modèles de CoS ont été standardisés et peuvent être utilisés indifféremment.

• **Voix :** Regroupe toutes les applications du type conversationnel (Voix, Visio, Conférence, ...) ayant pour contrainte forte des objectifs sur le délai et la gigue. Elles sont également sensibles au taux de perte bien qu'il ne soit pas possible de retransmettre les données et requièrent des débits assez faibles.

• **Vidéo :** Regroupe toutes les applications multimédia (Vidéo à la demande – VoD, la télévision sur IP – IP TV, ...) ayant pour contrainte forte le taux de perte et le débit et dans une moindre mesure le délai et la gigue,

• **Donnée :** Regroupe toutes les applications de transfert de données ayant pour seule contrainte un taux de perte nul et qui s'accommodent d'un délai et d'une gigue quelconque. Un débit garanti caractérise cette classe sans toutefois en faire une contrainte stricte,

• **Défaut :** Désigne toutes les applications n'exigeant aucune garantie de QoS. Bien connu sous l'anglicisme « **Best-Effort** » c'est le mode de transport du protocole IP.

II.5 Mécanismes de garantie de la qualité de service (modèles de services) :

La transmission des paquets à travers un réseau IP nécessite des performances respectables ainsi qu'une grande stabilité. Une transmission est gravement perturbée par d'éventuels retards ou coupures, Il faut donc veiller à ce que le flot soit le plus continu possible et que les variations restent faibles.

Un réseau IP classique offre un simple service qui est le best effort. La diversité des services à supporter entraîne également une stratégie de gestion de Qualité de Service différente. IntServ est une technique qui permet un service garanti en traitant les flux des paquets en fonction de la demande de la source juste avant de démarrer l'envoi des paquets utiles et cela par la réservation des ressources. Cependant, ce mécanisme se heurte à un autre problème, celui du facteur d'échelle. Avec IntServ, chaque routeur dans le réseau doit garder l'état de chaque flux qui y transite jusqu'au moment où la liaison s'achève. Un deuxième service qui est permis sur le réseau IP est le service différencié et cela par le mécanisme DiffServ. Ce dernier permet de différencier les classes au niveau de chaque routeur. Il résout le problème du facteur d'échelle de IntServ en définissant un nombre limité de comportement au niveau de chaque noeud. Le MPLS est aussi un mécanisme de qualité de service permettant des applications temps réel parce qu'il permet une optimisation de trafic et délai d'acheminement plus court.

Cette partie présente plus en détails les mécanismes : **Best effort, IntServ** (INTegrated SERVices), **DiffServ** (Differenciated Services) et finalement le **MPLS** (Multi-Protocol Label Switching).

II.5.1 Service Best effort :

Les réseaux IP classiques offrent un simple service : le service best effort. Un tel modèle de service permet aux routeurs d'être sans état et de ne garder aucune information à propos du trafic. L'architecture Internet est donc basée sur le concept que tous les états relatifs à un flux doivent être dans le système d'extrémité. Cette propriété confère au système global une grande robustesse.

En fournissant un modèle de service minimal, l'Internet est extensible en taille et en hétérogénéité des applications et des technologies. Ensemble, elles sont les deux raisons techniques majeures de son succès. Par ailleurs, l'utilisation de la couche réseau est libre de

toute tarification puisque les mêmes ressources sont disponibles pour tous les utilisateurs. L'inconvénient est que, puisqu'il n'y a pas de contrôle d'admission, le réseau peut être perturbé par des utilisateurs trop gourmands. Comme IP est un protocole sans connexion, le concept de contrat de trafic n'existe pas. Si le débit avec lequel le trafic est dirigé sur les interfaces dépasse la vitesse avec laquelle ces mêmes interfaces sont capables d'acheminer le trafic vers l'aval, des congestions peuvent se produire. Le trafic en excès est placé dans les files d'attente des dispositifs physiques jusqu'à débordement de ces files.
Ainsi, les applications peuvent faire l'expérience de délais variables ou de pertes de paquets. Les congestions peuvent entraîner des pertes transitoires ou bien des pertes de longue durée.

Le protocole d'extrémité TCP a été conçu pour assurer la fiabilité et la retransmission si nécessaire. Par ailleurs, comme le réseau n'effectue pas contrôle de congestion, cette fonction doit impérativement être assurée par les extrémités.

II.5.2 Services intégrés : IntServ (INTegrated SERVices) :

II.5.2.1 Présentation de IntServ :

Le modèle IntServ a marqué historiquement, en 1994, la volonté de l'IETF (*Internet Engineering Task Force*) de définir une architecture capable de prendre en charge la qualité de service en temps réel et le contrôle de partage de la bande passante sur les liens réseau.

IntServ se repose sur deux principes fondamentaux tels que le contrôle d'admission et le mécanisme de réservation de ressources.

En effet, le réseau doit être contrôlé et soumis à des mécanismes de contrôle d'admission. Ce mécanisme détermine si un routeur ou un hôte, est capable de répondre à une nouvelle demande de QoS, sans gêner les demandes qui ont été déjà accordées. Le contrôle d'admission, est invoqué dans chaque nœud afin de prendre la décision d'accepter ou de refuser une demande de service temps réel le long du chemin entre les utilisateurs finaux.

Afin de pouvoir garantir que la QoS demandée est bien présente, l'on confie au contrôle d'admission, d'autres tâches, comme l'authentification de ceux qui effectuent des réservations ainsi que d'établir des rapports concernant ce qui a été fait, qui a demandé quelle réservation, cela afin d'obtenir une sorte de feedback de l'utilisation des mécanismes de QoS.

Le deuxième mécanisme c'est la réservation de ressources par un protocole de signalisation établissant cette réservation **(**RSVP : **ReSerVation Protocol)**.

Ce dernier est utilisé pour transporter les messages de réservation de ressources. Ces messages sont ceux qui indiquent aux différents nœuds, la quantité de bande passante qu'une communication souhaite disposer.

Chaque routeur IntServ est ainsi constitué des éléments suivants :

- **Le classificateur :** afin de pouvoir effectuer un contrôle du trafic, il s'agit de pouvoir identifier chaque paquet entrant à l'aide de champ descripteur de flux et donc pouvoir l'associer à une certaine classe; sachant que tous les paquets figurants dans une classe sont soumis au même traitement. Le classificateur se basant sur le contenu de l'en-tête du paquet détermine à quelle classe appartient le paquet. Une classe correspond à une catégorie de flux,

par exemple le flux audio, ou encore le flux vidéo. Cela permet d'attribuer des caractéristiques distinctes à chaque flux.

- **L'ordonnanceur de paquets :** il contrôle l'acheminement vers la prochaine destination, qui peut être un autre routeur, ou un LAN des différents paquets. Pour cela, il travaille avec différentes files d'attente, ainsi que d'autres mécanismes comme par exemple les timers. Il doit être implémenté à l'endroit où les paquets sont mis en files d'attente, cela correspond souvent à l'interface de sortie et donc au protocole de niveau liaison de données.

- **Le contrôle d'admission :** il implémente l'algorithme qui détermine si un routeur ou un hôte, est à même de répondre à une nouvelle demande de QoS, sans entraver les demandes qui ont été déjà accordées. Le contrôle d'admission, est invoqué dans chaque nœud afin de prendre la décision d'accepter ou de refuser une demande de service temps réel le long du chemin entre les utilisateurs finaux.

- **Le protocole de réservation de ressources :** il est nécessaire afin de créer et maintenir un état décrivant les spécificités d'un flux dans chaque routeur le long du chemin, ainsi que dans les hots finaux. De manière à pouvoir indiquer de quel type de ressources a besoin une application, elle doit spécifier la QoS désirée en utilisant une liste de paramètres.

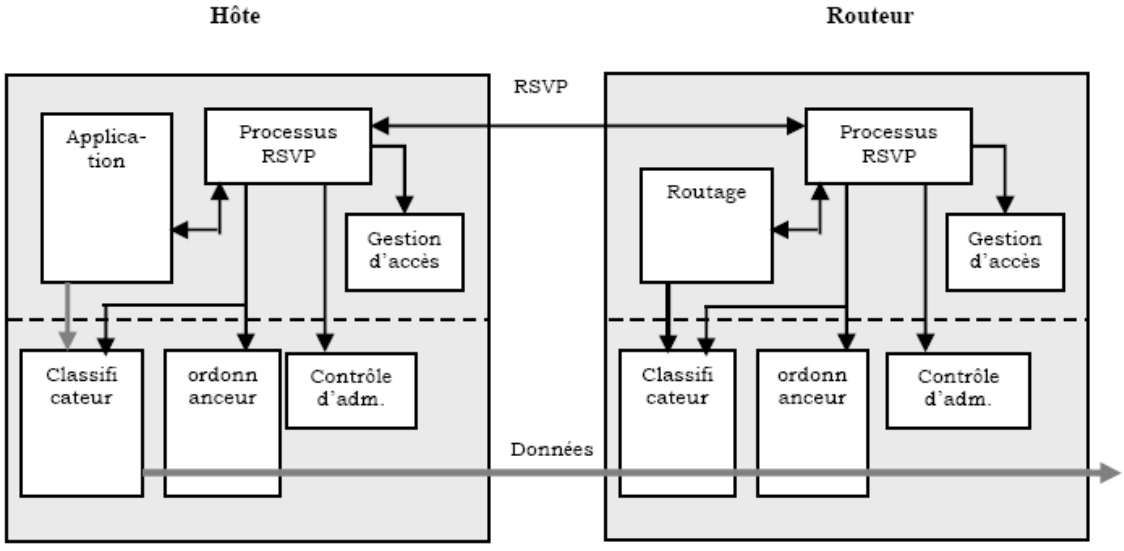


Figure **II.5.2.1** : **Modèle IntServ entre un routeur et hôte**

II.5.2.2 Principe de l'architecture à Intégration de Services :

Les applications traditionnelles non temps réel comme FTP se sont longtemps satisfaites du service best effort. Mais avec l'arrivée des communications multimédias, de nombreuses applications sont devenues sensibles au délai si bien que le service best effort

traditionnel ne suffit plus. Bien que certaines applications soient adaptatives (c'est à dire qu'elles réagissent dynamiquement en fonction de la charge du réseau) , il est souvent nécessaire de fournir de nouvelles classes de service offrant une meilleure QoS (en terme de bande passante, délai ou pertes). Ces nouvelles classes de service s'ajoutent au best effort traditionnel pour créer un
Internet à intégration de services.

L'architecture à Intégration de Services, développée par le groupe de travail Intserv de l'IETF, propose un ensemble d'extensions à l'architecture d'Internet. Dans le cadre d'Intserv, la QoS est associée au délai de transit des paquets.

L'architecture Intserv repose sur deux principes fondamentaux :

· le réseau doit être contrôlé et soumis à des mécanismes de contrôle d'admission,

· Des mécanismes de réservation de ressources sont nécessaires pour fournir des services différenciés.

Un mécanisme explicite est utilisé pour signaler les exigences de QoS par flot aux éléments du réseau (hôtes, routeurs ou sous-réseaux). Les éléments du réseau, selon les ressources disponibles, implémentent l'un des services Intserv en fonction du type de QoS souhaité pendant la transmission des données. Le modèle distingue plusieurs types de services, en fonction du délai de transit par paquet :

- les services élastiques ou non temps-réel (par exemple, connexion à distance, transfert de fichiers, messagerie). L'application attend d'avoir reçu les paquets avant de traiter les données, le délai de transit peut varier, le service est de type best-effort.
- les services temps-réel sensibles au délai et surtout à la gigue. Parmi ces services, le modèle distingue les services tolérant une variation du délai de transit pour lesquels la classe de service "à contrôle de charg**e"** est définie, et les services ne tolérant pas de variation pour lesquels la classe de service "garanti" est définie. Pour ces deux classes de service, le modèle définit une spécification de service QoS caractérisant le service attendu et des paramètres de trafic caractérisant le trafic à transmettre.

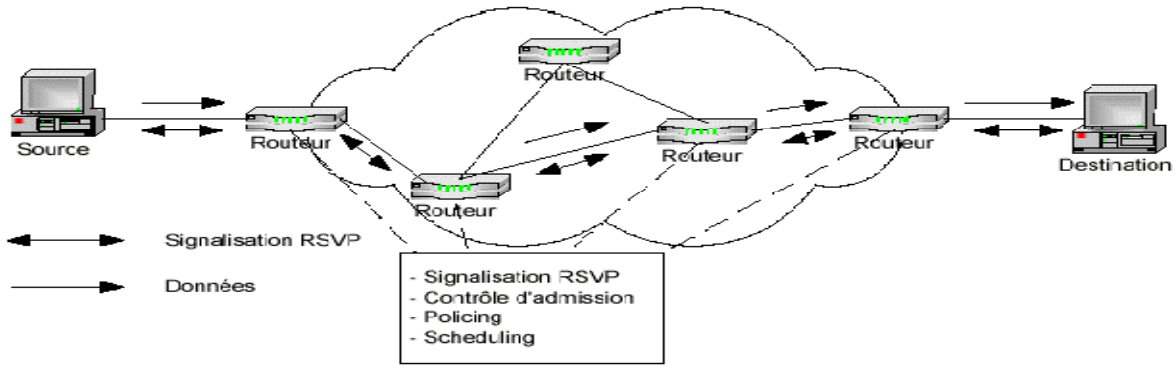


Figure **II.5.2.2 : Architecture RSVP/ Intserv**

II.5.2.3 Le protocole RSVP (ReSerVation Protocol) :

RSVP est un protocole qui réserve des flux de données simples, c'est à dire dans une seule direction. En fait, RSVP traite l'émetteur et le récepteur de manière distincte, cela même si parfois les deux interagissent sur la même application. RSVP n'est pas un protocole de transport de données, mais plutôt un protocole de contrôle. RSVP se situe au-dessus de IP, c'est un protocole qui se charge de contrôler la qualité de l'acheminement des paquets, et non de « router » les messages. Il consulte la table de routage locale, pour obtenir les routes des destinations à atteindre.

De manière à pouvoir satisfaire un grand nombre de récepteurs, RSVP rend responsable le récepteur de demander une configuration spécifique de QoS. A partir de là, une demande de QoS est acheminée au « processus » local de RSVP. Une fois la demande de QoS connue, le protocole RSVP achemine cette demande vers tous les nœuds (routeurs et hôtes), en empruntant le chemin inverse jusqu'à la source. Pendant la phase de réservation et de configuration, la demande de QoS passe au travers de deux modules différents, que sont "l'admission control" et "le policy control".

• **L'admission control :** garantit que le nœud a suffisamment de ressources disponibles pour répondre à la demande de QoS.

• **Le policy control :** détermine si l'utilisateur a les droits pour faire une réservation.

Si ces deux tests sont passés avec succès, les paramètres sont inscrits dans le "packet classifier" et dans ce qui sert de couche de liaison, afin d'obtenir la QoS demandée. Si l'un de ces deux tests échoue, le programme RSVP émet un message d'erreur à l'application qui est à l'origine de la demande.

Du fait que la disposition des topologies d'acheminement est susceptible de changer au cour du temps, RSVP a prévu cela. En fait RSVP envoie périodiquement des messages de rafraîchissement, afin de continuer à maintenir les différentes réservations le long du chemin. En absence de ces messages de rafraîchissement, l'état est automatiquement effacé et les ressources libérées.

Sept Types de Messages RSVP ont été prévus:

Path[4] **:** envoyé par la source pour indiquer la liste des routeurs du chemin suivi par les données.
Resv : demande de réservation.
PathErr : message d'erreur concernant le chemin.
ResvErr : message d'erreur de demande de réservation.
PathTear : indique aux routeurs d'annuler les états concernant la route.
ResvTear : indique aux routeurs d'annuler les états de réservation (fin de session).
ResvConf (optionnel) : message de confirmation envoyé par le routeur au demandeur de la réservation.

[4] RSVP utilise le message Path pour déterminer las caractéristiques du flux entrant.

Un message RSVP est constitué d'un en-tête et d'un nombre variable d'objets qui dépend du type du message.

L'entête est constitué de 64 bits:

Vers	Flags	Type du Msg	Checksum
Send_TTL		Réservé	Longueur Msg.

Figure **II.5.2.3 : Entête RSVP**

Vers (4 bits) : version du protocole RSVP (=1);
Flags (4) : non utilisé à ce jour;
Type de Msg (8) : 1 à 7 selon le type ci-dessus;
Checksum (16) : Contrôle d'erreurs;
Send_TTL (8) : valeur du TTL IP à comparer avec le TTL du paquet IP pour savoir s'il y a des routeurs non-RSVP;
Longueur (16) : longueur du message en octets (en-tête et objets).

RSVP travaille notamment avec les messages PATH et RESV. Le message PATH part de la source vers la destination et RESV emprunte le chemin inverse. PATH indique les caractéristiques du trafic, et RESV opère la réservation.

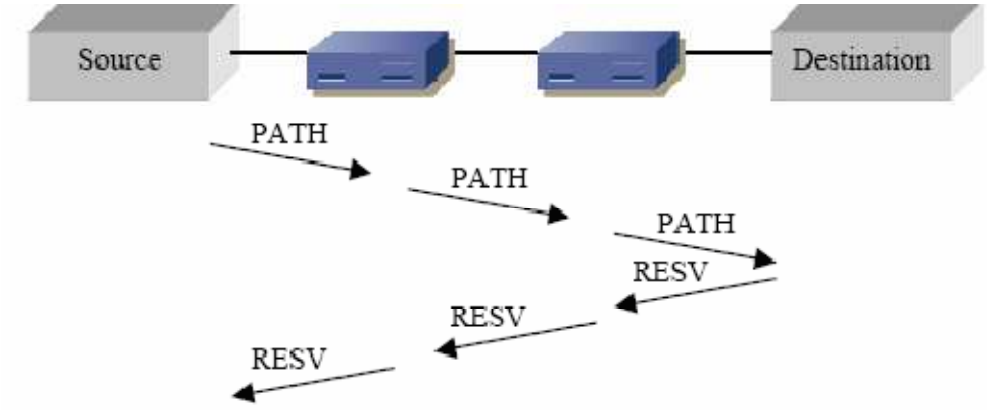


Figure **II.5.2.3 : Réservation de ressources**

II.5.3 Services différenciés (DiffServ) :

II.5.3.1 présentation de DiffServ :

DiffServ est un modèle qui permet de classifier chaque paquet selon son contenu, plus précisément selon l'information se trouvant dans le champ (Type Of Service sur IPV4, Class of Service et Flow Label sur IPV6) de la trame IP et d'appliquer un traitement en fonction de

celui-ci. Chaque paquet se verrait donc confier une estampille de couche réseau indiquant quelle information il transite. Dès son entrée sur un réseau, il lui sera donc accordé des privilèges de traitement valables sur tout ou partie de celui-ci. Cette méthode permet donc d'appliquer des concepts de priorité déjà supportés par des appareils de routage dans le cadre du traitement des files d'attente et de la congestion.

Les critères de classification des paquets doivent refléter les besoins réels de l'information qu'ils transportent ; il va de soi que le choix d'un degré de priorité n'est pas arbitraire. Pour ce faire, une petite étude des besoins en bande passante et de la sensibilité des différents types de données s'impose.

II.5.3.2 Caractéristiques du modèle :

Les modèles DiffServ et IntServ cherchent à résoudre les mêmes problèmes tels que le traitement correct des flux multimédia et un meilleur contrôle dans la distribution de la bande passante. Néanmoins, le modèle DiffServ s'attaque à ces contraintes d'une manière moins ambitieuse mais plus résistante. Dans ce modèle, il est impossible d'offrir des garanties strictes ou de réserver des ressources. Par contre, la simplicité d'implantation permet à cette nouvelle architecture de se voir déployée progressivement dans certaines régions de l'Internet.

Une des faiblesses du modèle IntServ est sa non-résistance au facteur d'échelle (évolutivité). Dans ce dernier, tous les équipements du réseau doivent garder un état par flux réservé. Il suffit qu'un noeud dans la route n'implémente pas les fonctionnalités IntServ pour que la QoS ne puisse plus être strictement garantie. Cependant, dans l'architecture DiffServ, la priorité est donnée au regroupement des flux dont les besoins sont similaires (agrégation) et à la définition des traitements nécessaires dans les routeurs pour que l'agrégat de ces flux soit traité correctement.

Pour assurer la robustesse du modèle, la création d'états et la classification par flux sont deux fonctionnalités réservées aux routeurs d'entrée au réseau. Dans ces équipements, le nombre de flux à traiter est considérablement réduit. Dans le reste des routeurs, des opérations très simples, ne demandant pas la création d'états, assurent le traitement différencié. Une autre approche faite au modèle IntServ est la complexité du protocole de signalisation
RSVP. Une grande partie de la lourdeur du protocole est due à la gestion des flux multicast et des routes symétriques. La réservation de ressources pour des flux exige la définition de règles d'agrégation et désagrégation dans les nœuds intermédiaires.

Dans DiffServ, ce problème n'a pas été spécifiquement abordé car les connexions multi-point n'interfèrent pas avec les mécanismes propres à l'architecture. Dans ce modèle, la seule signalisation requise est une étiquette contenue dans l'en-tête de chaque paquet. Nous résumons dans le tableau ci dessous, les différences entre ces deux approches :

	IntServ	DiffServ
Type de différenciation	garantie absolue	garantie statistique
Granularité de différenciation	micro - flux	Agrégats
Etats	Par flux	Par agrégat
Base de classification	Plusieurs champs d'entête	Un champ : DS
Signalisation	Explicite	Implicite dans le coeur
Réservation	requise (RSVP)	Non requise dans le coeur
Types routeurs	Un type	Deux types : bordure/cœur
Extensibilité	Limitée par nombre de flux	Limitée par nombre de classes

Figure **II.5.3.2 : Tableau comparatif des approches de QoS**

II.5.3.3 Architecture de DiffServ

L'idée consiste à diviser le réseau en domaines. Ainsi, on distingue à l'intérieur d'un domaine des routeurs d'accès (Core router) et d'autres de bordure (Edge router). Les routeurs d'accès sont connectés aux clients, tandis qu'un routeur de bordure est connecté à un autre routeur de bordure appartenant à un domaine différent. Le modèle DiffServ est basé sur une architecture qui permet des prises de décision complexe aux bords et donc moins de charge sur les routeurs du backbone. En effet, les routeurs d'extrémités sont chargés de conditionner le trafic entrant en indiquant explicitement sur le paquet le service qu'il doit subir. Ils examinent les paquets, les classifient selon une politique spécifiée par l'administrateur de réseau, les marquent et exécutent des fonctions de profilage. Ainsi, la complexité des routeurs ne dépend plus du nombre de flux qui passent mais du nombre de classes de service. Ces dernières sont identifiées par une valeur codée dans l'en-tête IP.

La figure ci dessous montre un exemple de configuration de réseau. Un site émetteur, souhaite que ses flux bénéficient d'un traitement différencié dans le réseau. Il établit un contrat de service avec son fournisseur. Ce contrat, appelé SLA (Service Level Agreement), contient la nature du trafic que l'émetteur peut produire pour chaque service demandé. D'un coté, le fournisseur s'engage à fournir la qualité demandée par l'émetteur. De son coté, l'émetteur est conscient des limitations imposées par le contrat.

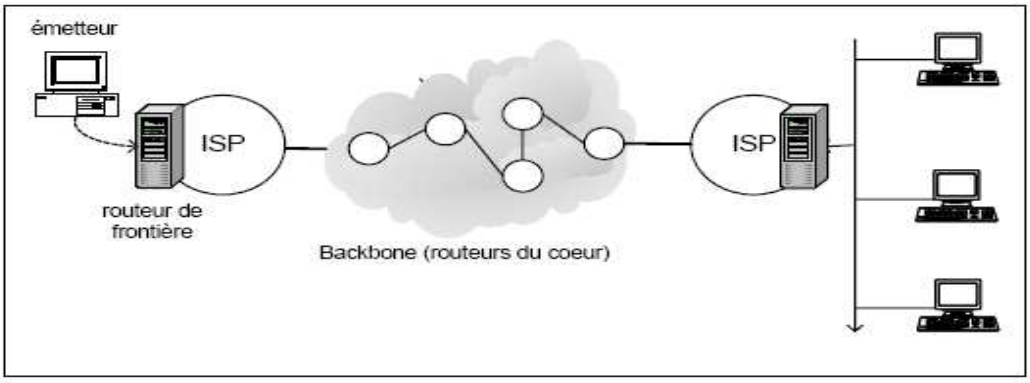


Figure **II.5.3.3 : Architecture du modèle DiffServ**

Le client peut effectuer un pré-marquage de ses flux avant de les transmettre au fournisseur[5]. Signaler une préférence en termes de classe de service ou indiquer l'importance relative des paquets sont deux raisons qui justifient cette action. La priorité des paquets peut varier au sein d'un flux. Par contre, les principes de DiffServ exigent que tous les paquets d'un même flux appartiennent à une même classe de service afin d'éviter le déséquencement.

Le fournisseur connaît les spécifications techniques du contrat établi avec chacun de ses clients. Quand le trafic produit par l'utilisateur arrive au routeur d'entrée du fournisseur, les flux sont identifiés et leur comportement est vérifié en fonction du contrat respectif. En cas de non-conformité, une action corrective est prise. Pour certains services, seuls les paquets conformes au contrat peuvent entrer dans le réseau. Pour d'autres services, tous les paquets sont acceptés, mais le routeur d'entrée modifie la priorité du paquet en fonction du niveau de conformité. En quittant le routeur d'entrée, les paquets du site émetteur contiennent tous une étiquette reflétant la classe de service souhaitée et le niveau de conformité du flux par rapport au contrat.

Les routeurs du cœur du réseau ne modifient pas les étiquettes. Ils se contentent de les utiliser pour choisir la file d'attente où le paquet est inséré. Pour cet exemple, la file d'attente représente la classe de service. Les ressources des routeurs sont distribuées entre les files en fonction des services qu'ils supportent. Dans chaque file d'attente, un algorithme de rejet sélectif est utilisé pour éliminer, en premier, les paquets de basse priorité en cas de congestion.

Cet exemple entre un site et un fournisseur d'accès peut s'appliquer également à deux opérateurs. En général, les paquets doivent traverser plusieurs domaines administratifs avant d'atteindre leur destination. Un accord est, donc, nécessaire entre domaines. Le fournisseur d'accès a aussi des contraintes à respecter vis-à-vis de son opérateur. Le routeur d'entrée est placé entre le réseau du fournisseur et celui de l'opérateur, réalisant des opérations similaires à celles du routeur décrit précédemment. Par contre, les fonctions de contrôle de conformité ne portent pas sur le trafic de chaque utilisateur, mais sur le trafic injecté par le fournisseur dans son ensemble[6]. Cette capacité d'agrégation assure la résistance du modèle au facteur d'échelle.

---

[5] Définir la qualité de service QoS souhaité par le client mais un contrôle de profil définit par le fournisseur est nécessaire pour le comparer avec ce dernier.

[6] Les fonctions de contrôle de conformité sont assurées par les routeurs de bordure et non par ceux du cœur.

Les capacités des routeurs de sortie constituent une autre particularité de l'interconnexion entre deux domaines. Un routeur de sortie est le dernier routeur traversé par un paquet avant de quitter un domaine. Ces routeurs peuvent être chargés de traiter l'agrégat des flux quittant le domaine afin de rendre le trafic conforme au contrat existant avec le prochain domaine. Pour ceci, les flux appartenant à une classe de service peuvent être retardés, tandis que des paquets d'une autre classe peuvent subir une modification dans leur niveau de priorité.

II.5.3.3.1 Agrégation de flux[7] :

Une caractéristique de l'architecture DiffServ, qui lui permet d'être étendue à des réseaux, est le fait que les routeurs de cœur ne considèrent pas les flux individuels. Les paquets IP ne sont pas distingués au niveau fin du flux mais à un niveau plus grossier d'agrégats de flux. L'agrégat d'appartenance d'un paquet est reconnu par un identifiant de classe enregistré dans le champ DSCP (DiffServ Code Point). Cette étiquette est le champ qui identifie le traitement que le paquet doit recevoir. Elle est codée sur 6 bits et fait parti des 8 bits codant le champ TOS d'IPv4 ou le champ classe de trafic d'IPv6.

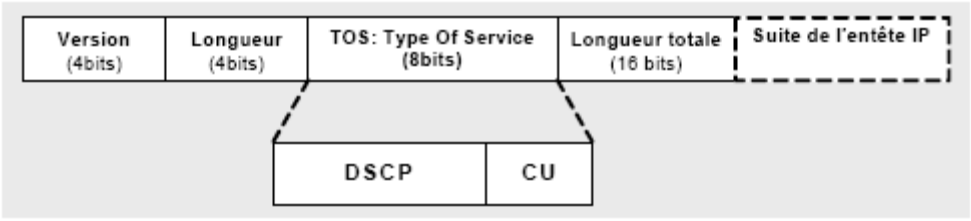


Figure **II.5.3.3.1 : Le champ DSCP**

En effet, les trois premiers bits de DSCP sont appelés Class Selector. Les codes DSCP de type xxx000 (ou x est une variable binaire) correspondent aux classes de services principales. Ceux-ci seront associés aux PHB (Per Hop Behviour) qui permettront le traitement différencié des flux dans les routeurs intermédiaires. Plus la valeur de code point est élevée, plus le flux correspondant sera prioritaire. Les deux derniers bits sont inutilisés.

Les agrégats sont en nombre réduit, fixé, bien inférieur au nombre de flux distincts qui peuvent traverser un routeur. C'est le petit nombre de classes qui permet aux routeurs de cœur de mettre en œuvre un traitement différencié moins coûteux que celui des routeurs IntServ. Le modèle DiffServ sépare le trafic par classes. Nous avons donc affaire à une granularité moins fine mais qui devient en revanche plus évolutive. En effet, la granularité du flot implique la réaction en chaîne suivante : plus il y a d'utilisateurs dans le réseau, plus il y a de flots ainsi que des variables de classification et d'ordonnancement dans les routeurs à maintenir. Ceci a pour conséquence une charge importante au niveau des routeurs qui deviennent alors de moins en moins performants.

II.5.3.3.2 Traitement par noeud (Per Hop Behavior)

L'architecture consiste aussi en un ensemble de mécanismes simples au niveau de cœur de réseau, agissant sur un nombre réduit de classes de service, dont la sémantique est définie non

[7] Agregats de flux désigne un ensemble de flux qui ont des besoins similaires.

pas de bout en bout mais par routeur (per-hop behavior). Les PHBs sont les mécanismes mis en œuvre dans le cœur de réseau, ceux qui pratiquent le traitement différencié entre les classes. Ils sont définis par les constructeurs dans les routeurs en utilisant des mécanismes de gestion de files d'attente (Round Robin, Weighted Fair Queuing,...) et de régulation de flux. En effet, deux modèles ont été standardisés par l'IETF tels que PHB Expedited Forwarding (EF) et PHB Assured Forwarding (AF).

II.5.3.4 Le Traitement différencié de paquet dans le routeur DiffServ :

Le routeur DiffServ agrège le flux en un ensemble de « Behaviour Aggregate » qui seront acheminés de la même manière, ce qui assure une simplification des traitements et de stockage dans le routeur. Ainsi, l'architecture DiffServ définit quatre types d'éléments qui constituent le chemin emprunté par les flux lorsqu'ils passent par le routeur. Ces composants sont le classificateur de trafic, les éléments d'actions, de mesures et les gestionnaires de file d'attente. La tâche du classificateur de trafic est de sélectionner les flux correspondants à des critères comme l'adresse IP et le numéro de port. Le marqueur, représentant un des éléments d'action, marque ces flux par un code DSCP pour recevoir un traitement particulier de la part des routeurs DiffServ à l'intérieur du réseau. La figure ci-dessous illustre les différents blocs fonctionnels du routeur DiffServ qui seront détaillés dans la suite.

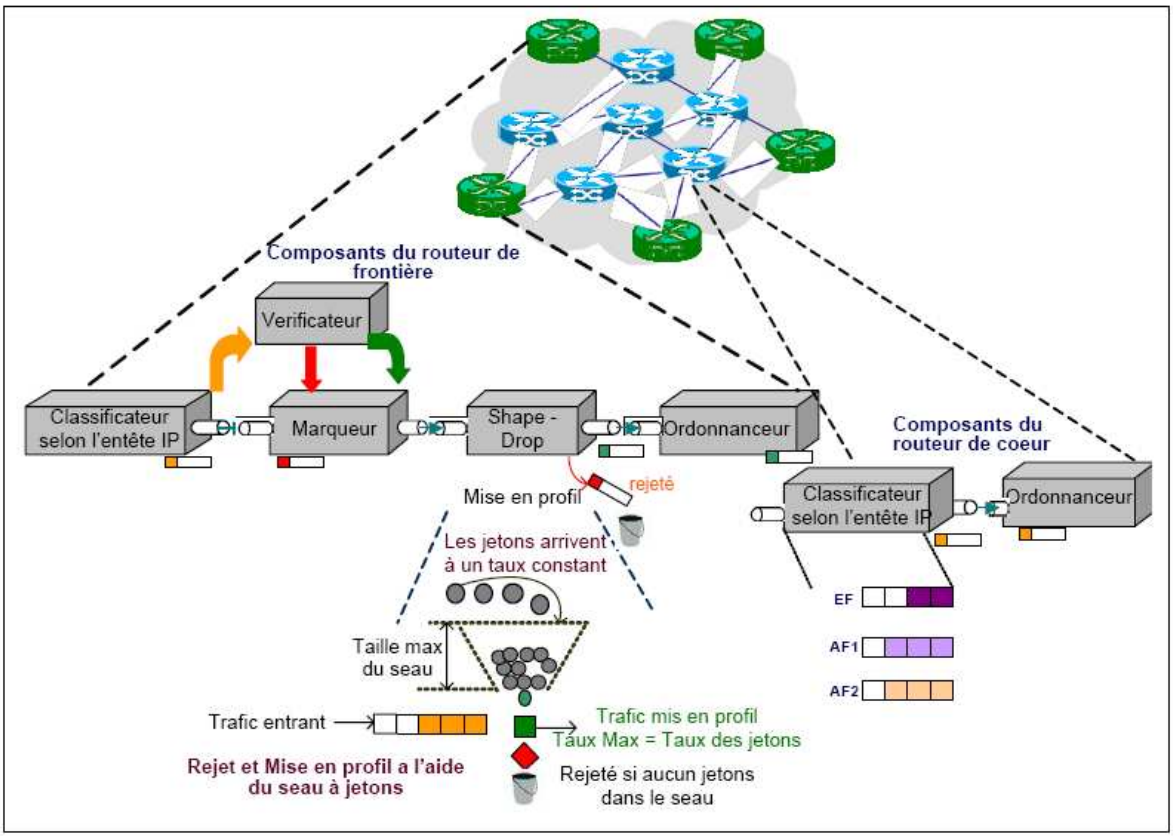


Figure **II.5.3.4 : Les Composants QoS d'un routeur IP DiffServ**

Les différents éléments présentés dans la figure II.5.3.4 peuvent être facilement contrôlés par un modèle de configuration. La configuration des ces éléments DiffServ reflète le SLA (Service Level Agreement) entre le fournisseur de service et l'utilisateur. Il pouvait exister une entité de gestion du domaine DiffServ qui représente un outil d'administration capable de fournir la configuration adéquate aux routeurs.

Chaque équipement frontière met en œuvre les mécanismes de classification, de conditionnement et d'ordonnancement des trafics. L'ampleur de conditionnement de trafic exigée est indépendante des détails des offres de service et peut s'étendre du simple marquage aux opérations complexes de lissage et « policing ». Les routeurs à l'intérieur d'un domaine, quant à eux, se consacrent exclusivement au traitement des paquets en utilisant le marquage spécifié par les nœuds de bord.

I. La classification de trafic :

Le classificateur au niveau de routeur DiffServ traite un flux en entrée et génère un flux en sortie. La séparation en des classes de flux est réalisée par des filtres. En effet, il s'agit de choisir des paquets dans un flot de trafic basé sur le contenu d'une certaine partie de l'entête du paquet. Généralement, on distingue deux types de classificateur. Le premier, appelé BA (Agrégat de comportement), utilise comme clé de classification le « code point » et le deuxième, appelé MF (Multi Field), sélectionne les paquets en se basant sur la valeur d'une combinaison d'une ou plusieurs zones d'entête, telles que l'adresse source, l'adresse destination, le champ DS, l'identification du protocole, les numéros de port source et destination…etc. Ainsi, le filtre consiste à un ensemble de conditions sur les valeurs composantes les clés de la classification du paquet.

Dans l'architecture DiffServ, lors d'une arrivée de flot non classifié au niveau de « Edge Router », les filtres sont utilisés pour sélectionner les paquets IP en fonction des informations contenues dans l'entête IP. Une fois les paquets sont filtrés, ils sont mis dans des classes indépendantes et peuvent être contrôlés d'une manière indépendante. Chaque agrégat possède son propre gestionnaire de file d'attente. Ainsi, le routeur arrive à faire du traitement différencié. Le modèle DiffServ introduit trois PHB ou classes de services définies comme suit :

1. *Expedited Forwarding (EF) ou Premium Service* : cette classe correspond à la valeur « 101110 » de DSCP. Son objectif est de fournir un service de transfert équivalent à une ligne virtuelle dédiée à travers le réseau d'un opérateur. Les paquets excédentaires sont lissés ou rejetés à l'entrée pour toujours rester conforme au contrat SLA. De plus, les flux ne doivent avoir que très peu de perte, la gigue doit être minimale et la bande passante garantie. On utilise couramment cette classe pour le transport de données temps réel tel que la voix ou la visioconférence.
2. *Assured Forwarding (AF)* : Il s'agit du second modèle de service standardisé par l'IETF qui représente quatre (4) classes de services et trois (3) niveaux de taux de perte dans chaque classe. Il regroupe plusieurs PHB garantissant un acheminement de paquets IP avec une haute probabilité sans tenir compte des délais. Cette famille de PHB, scindée en quatre classes, soutient un partage plus flexible et plus dynamique des ressources de réseau en maintenant

des garanties fines de bande passante, un délai minimum et de pertes appropriées pour le trafic à flot. Chaque classe AF supporte un mécanisme permet de gérer les flux qui dépassent la capacité souscrite. Ceci est réalisé grâce à trois niveaux de priorités (Drop Precedence) définis dans une classe AF et différenciés par un algorithme de rejet sélectif. En cas de congestion dans une des classes AF, les paquets de basse priorité sont rejetés en premier. La priorité peut être modifiée dans le réseau par les opérateurs en fonction du respect ou non des contrats SLA. Ainsi le service peut offrir une meilleure différenciation (classe et priorité), il ne demande pas une coordination entre domaines. Cependant la qualité offerte dépend énormément du niveau d'agrégation et de la présence de flux concurrents.

	Classe AF1	**Classe AF2**	**Classe AF3**	**Classe AF4**
Low Drop Prec	001010	010010	011010	100010
Medium Drop Prec	001100	010100	011100	100100
High Drop Prec	001110	010110	011110	100110

Figure **II.5.3.4.1 : La valeur du champ DSCP pour les différentes classes AF**

3. Best Effort (priorité basse) : représente le PHB par défaut et dont le DSCP vaut « 000000 ». Le principe du Best Effort se traduit par une simplification à l'extrême des équipements d'interconnexion. Quand la mémoire d'un routeur est saturée, les paquets sont rejetés.

La figure ci dessous II.5.3.4.2 illustre la procédure de classification. On distingue une topologie de réseau qui se compose de trois sources : la première envoie un trafic CBR (Constant Bit Rate) à 64 Kbit/S modélisant la voix, la deuxième génère un flux continu modélisant le trafic AF1 et la dernière représente un trafic web. Le rôle de classificateur consiste, alors, à lier un flux particulier et ses paramètres et l'assigner à une classe de service tout en traitant le champ « Differentiated Service » DS. Ainsi, trois comportements sont distingués dans le routeur. Le trafic voix est placé dans la file de haute priorité (EF), le flux web est classé dans une file AF et le trafic provenant d'un serveur de base de données, par exemple, aura la priorité la plus faible.

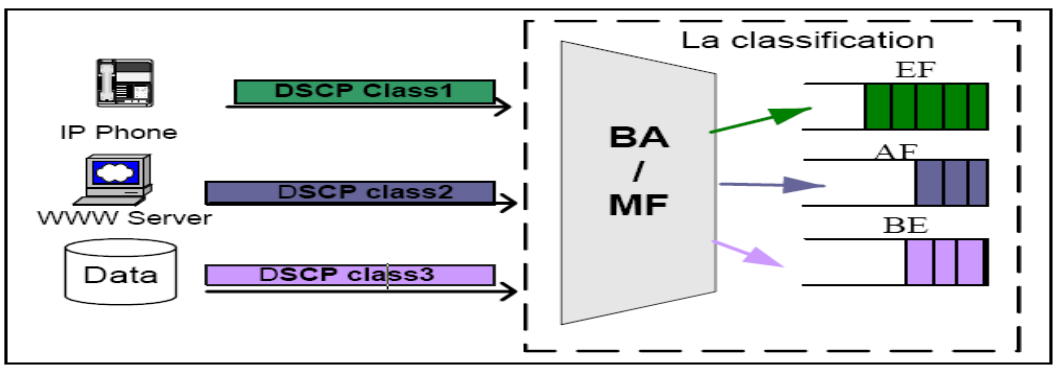


Figure **II.5.3.4.2** : **La fonction de classification**

II. Le conditionnement de trafic :

Le conditionneur de trafic peut contenir un ensemble d'éléments tels que le vérificateur, le marqueur, le « shaper » et le « dropper ». En effet, une fois un flot de trafic est choisi par un classificateur, il le dirige vers un module de conditionnement spécifié pour continuer le processus de traitement. Un conditionneur de trafic peut ne pas contenir nécessairement chacun des quatre éléments tels que le cas où aucun profil de traitement n'est présent (les paquets peuvent seulement passer par un classificateur et un marqueur).

A- Le vérificateur

Les vérificateurs du trafic mesurent les propriétés temporelles de flux de paquet sélectionné par le classificateur contre un profil spécifié dans le TCA (Traffic Condition Agreement). Il passe l'information d'état à d'autres fonctions de traitement pour déclencher une action particulière pour chaque paquet qui est hors profil ou non. Ce « traffic profile » a pour objet la prise en compte du taux d'arrivée des paquets, afin de ne pas dépasser le seuil maximum de paquets pouvant être envoyés sur le réseau. Ainsi, un mécanisme de mesure du trafic permet de savoir si le flot de paquets entrants correspond au profil de trafic négocié. Le « meter » est paramétré par un profil temporel et des niveaux de conformité. Chaque niveau est associé à une sortie. En DiffServ, par exemple dans la classe AF, trois niveaux de conformités sont discutés. Ces niveaux seront utilisés pour déclencher différents traitements dans la file, de même pour le module de marquage et de rejet. Pour un service EF, un trafic jugé nonconforme par le « meter » est dirigé vers le module de rejet.

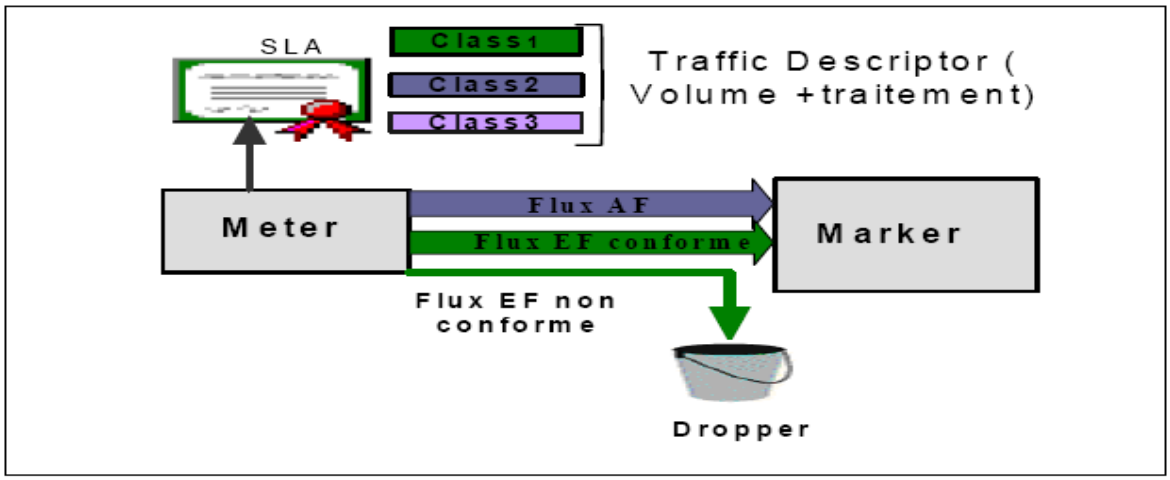


Figure **II.5.3.4.3** : **Le principe de la vérification**

B- le « Shaper » et le « Dropper »

Suite à la vérification de la conformité vis-à-vis d'un profil, le lissage peut s'effectuer lorsque les flux d'une classe dépassent le contrat SLA prédéfini. Cette fonctionnalité n'est pas systématique et correspond à une règle de « policing » explicite dans le SLA. Les paquets sont alors mis en file d'attente afin d'être transmis un peu plus tard lorsque le débit de la classe

sera considéré comme étant dans le profil du contrat. Le rejet des paquets intervient pour garantir le débit fixé pour chaque classe de service. Dans le cadre d'un lissage, pour les files d'attente ayant une taille finie, des dépassements de profil trop importants peuvent aussi provoquer un rejet des paquets. Ainsi le régulateur de trafic assure principalement deux fonctions :

- *Mise en forme du trafic (Traffic Shaping)* : implantée généralement auprès des sources de trafic. Cette fonction permet de lisser le flux généré à l'entrée du réseau pour être conforme à une spécification particulière. Elle permet de contraindre le trafic pour le rendre conforme à sa spécification connue par le réseau.
- *La surveillance du trafic (Traffic Policing)* : Implantée dans les dispositifs du réseau, cette fonction permet de vérifier la conformité du trafic à son contrat, c'est-à-dire à sa caractérisation déclarée lors de sa négociation avec le réseau en phase du contrôle d'admission.

La différence entre ces deux fonctions réside dans leurs actions vis-à-vis d'un paquet non conforme. Dans ce cas, la fonction de mise en forme du trafic retarde le paquet jusqu'à ce qu'il soit conforme. Cependant, le traitement d'un tel paquet dans la fonction de surveillance du trafic dépend de la politique adoptée. En effet, lors de l'arrivée d'un paquet non conforme, l'algorithme de surveillance de trafic peut décider de retarder le paquet jusqu'à ce qu'il soit conforme, de l'éliminer entièrement ou de décrémenter sa priorité avant de le transmettre. Les mécanismes de régulation de trafic permettent d'imposer une forme spécifique au flux régulé. Théoriquement, cette forme est exprimée par une fonction quelconque qui limite le nombre cumulatif d'arrivées de paquets, appelée courbe d'arrivée, dans n'importe quel intervalle de temps. En pratique, cette fonction est typiquement linéaire pour des raisons de simplicité d'implémentation. Il existe deux modèles de courbes d'arrivée linéaires :

- *Le modèle de débit crête* : permet de limiter le débit maximal d'une source de trafic à un débit crête. Il est caractérisé par la taille de paquet maximal noté *M* et le temps d'inter-arrivée minimal entre deux paquets consécutifs notés *Tmin*. Ainsi, le débit crête qu'autorise le régulateur de trafic est $P = M / Tmin$.
- *Le modèle à débit moyen :* permet de limiter le débit moyen d'une source de trafic. Ce régulateur est caractérisé par deux paramètres : la taille de la rafale permise notée *b* et le débit moyen noté *r*.

Ces formes de trafic linéaires sont généralement obtenues par des régulateurs appelés seau à jetons (Token Bucket) et seau percé (Leaky Bucket). Le régulateur du seau à jetons dispose d'un seau de jetons de taille *b* qui sont générés au rythme constant de *r* jetons par seconde.

Chaque paquet qui arrive consomme un nombre de jetons proportionnel à sa taille. S'il n'y a pas assez de jetons dans le seau, le paquet sera mis en attente dans la file jusqu'à avoir le nombre nécessaire de jetons, sinon il sera transmis. Le régulateur du seau percé dispose d'un seau de taille *b* initialement vide. A chaque arrivée d'un paquet le seau est rempli avec une quantité de fluide égale à la taille du paquet. Le seau percé est vidé du fluide au rythme constant *r*. Si la quantité de fluide ajoutée lors de l'arrivée fait déborder le seau, alors le paquet est rejeté par le régulateur. Dans le cas contraire, le paquet est transmis vers le réseau. La figure suivante illustre les deux mécanismes du seau à jetons et du seau percé.

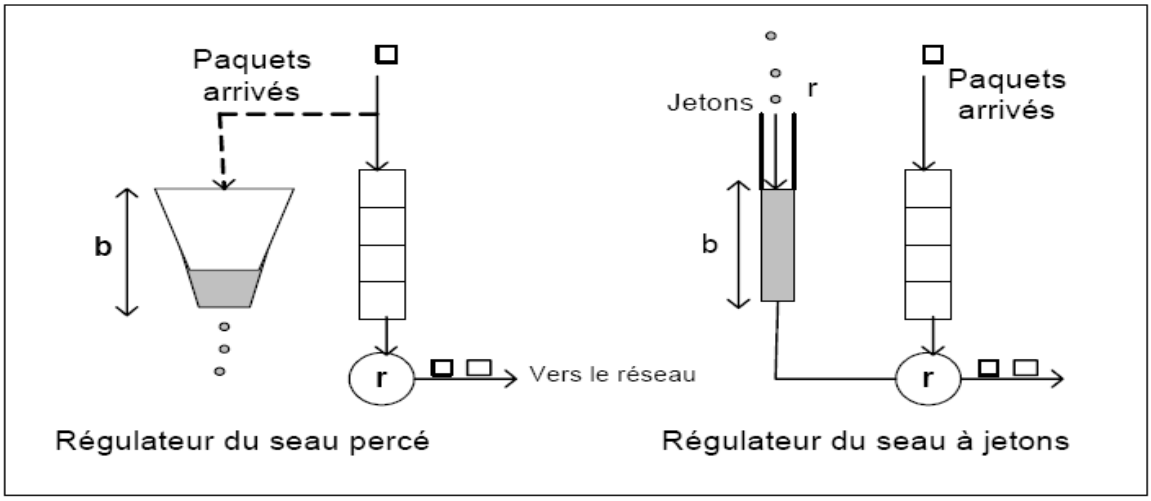


Figure **II.5.3.4.4** : **Les mécanismes de régulation de trafic : le seau percé et le seau à jetons**

C- Le marqueur

Le marquage fait référence à la mise à jour de la valeur du champ DS dans l'en-tête IP. En effet, les informations fournies en entrée par le vérificateur et le classificateur permettront de faire un choix sur la priorité à appliquer à chaque flux. Le bloc de marquage par exemple peut décider qu'en cas de dépassement du contrat, les flux excédentaires seront marqués avec une priorité moindre. Les marqueurs placent dans la zone DS d'un paquet un code point particulier de comportement DS. Donc, c'est dans ce module que sera affecté le champ DSCP. Ce traitement est basé actuellement sur deux token buckets. Le principe se résume comme suit :

- Si le trafic est conforme aux deux, les paquets sont marqués en vert,
- Si le trafic n'est conforme qu'un des deux, les paquets seront marqués en orangé.
- Si le trafic n'est conforme à aucun alors les paquets seront marqués en rouge et ils auront une probabilité de rejet plus importante que les autres.

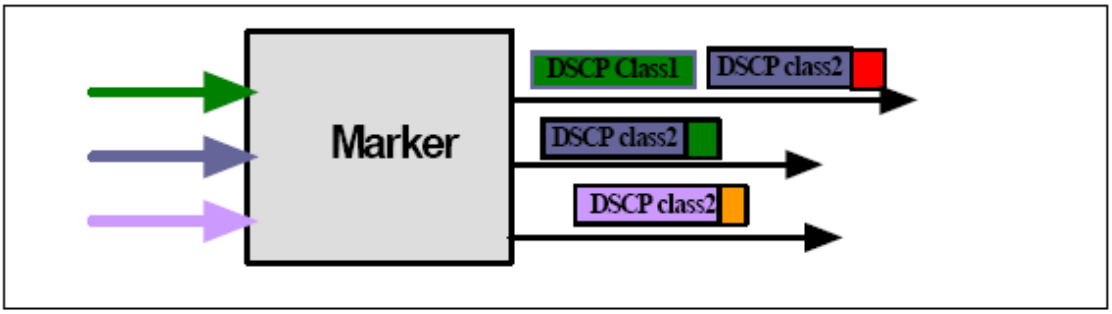


Figure **II.5.3.4.5 : La fonction de marquage**

III. L'ordonnancement de trafic

Une fois les paquets sont marqués et placés dans différentes files d'attente selon la valeur de la classe de service identifiée dans l'en-tête IP, ils seront servis par un ordonnanceur. Ce

dernier utilise des algorithmes et des disciplines d'ordonnancement qui permettent de contrôler la distribution de ressources. Plusieurs types de comportements d'ordonnancement et de politiques de rejets peuvent être employés pour fournir le PHB adéquat.
Les mécanismes d'ordonnancement permettent d'assurer le partage des ressources selon une politique de service spécifique à la nature de garantie à fournir aux applications en concurrence d'accès aux ressources. Plusieurs algorithmes d'ordonnancement ont été proposés pour satisfaire les exigences des applications temps-réel et fournir un service plus sophistiqué que l'ordonnancement FIFO. Ce dernier ne présente aucune garantie particulière à une application vis-à-vis d'une autre. Diverses classifications, présentées, dans la figure ci-dessous, peuvent être appliquées à ces algorithmes pour caractériser leurs comportements:

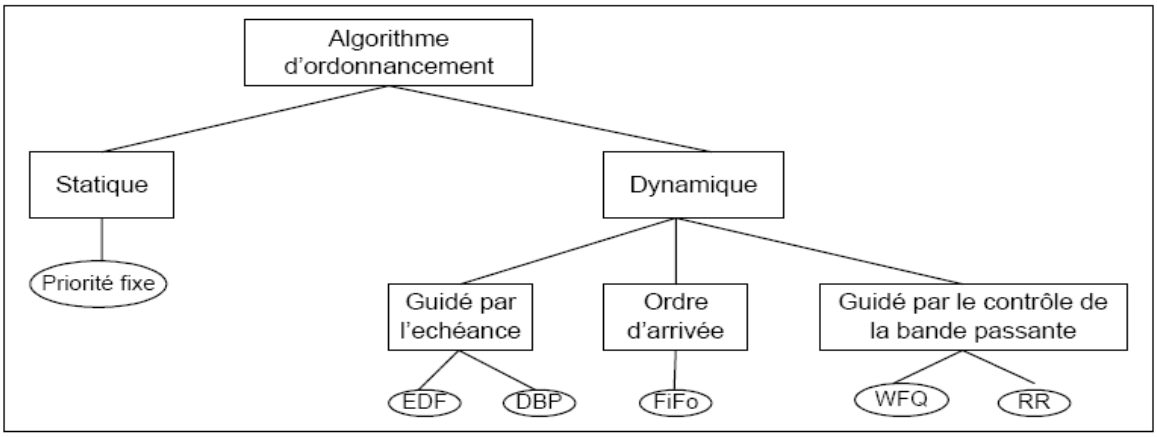


Figure **II.5.3.4.6 : Classification de l'ordonnancement**

Dans un algorithme d'ordonnancement statique, les priorités des applications sont fixes et invariantes au cours du temps. Un message moins prioritaire n'est servi que si tous les messages plus prioritaires en attente sont tous servis. Dans un algorithme d'ordonnancement dynamique, la priorité des messages est recalculée, chaque fois qu'un nouveau paquet arrive, suivant une règle spécifique à l'ordonnancement. Par exemple, l'algorithme Earliest Deadline First (EDF) cherche à chaque tour de sélection le client ayant l'échéance la plus proche parmi tous les clients en attente. En ce qui concerne la catégorie d'ordonnancement guidé par le contrôle de bande passante, elle permet de partager dynamiquement des ressources en respectant le critère de la bande passante. L'algorithme EDF est l'algorithme le plus connu dans les réseaux industriels.

Dans les réseaux à commutation de paquets, l'ordonnancement guidé par le contrôle de bande passante est le plus utilisé pour assurer un partage équitable des ressources. Nous retrouvons principalement les algorithmes Round Robin, WFQ et toutes leurs variantes.

Le choix d'un algorithme d'ordonnancement adéquat a un impact très important pour la fourniture de la QoS surtout des applications temps-réel. Dans les réseaux à commutation de paquets, la technique classique pour privilégier le service d'une application sous contraintes temporelles, par rapport à une autre application moins contraignante, est de lui affecter une priorité plus élevée et d'ordonnancer les flux en concurrence d'accès au médium de transmission selon leurs priorités.

C’est l’ordonnancement à priorité fixe. Bien que cette technique reste efficace pour améliorer la QoS temporelle des flux prioritaires, elle pourrait être coûteuse pour les flux moins prioritaires.
En effet, dans le cas où il n’existerait pas de mécanismes de contrôle d’admission pour contrôler l’accès de nouveau flux de priorités élevées, le problème de famine serait inévitable pour les flux moins prioritaires dans le cas où les flux de priorité consommeraient la quasi-totalité des ressources. En plus, un flux plus prioritaire malveillant pourrait affecter sérieusement le délai et la gigue exercée par les autres flux moins prioritaires partageant les mêmes ressources d’accès au médium de transmission.

Pour éviter ces problèmes, la tendance actuelle de l’ordonnancement des applications temps-réel dans les réseaux à commutation de paquets consiste à réserver les ressources nécessaires en terme de bande passante par le biais des algorithmes à partage de bande passante Round Robin, WFQ.

II.5.4 Optimisation de trafic: MPLS (Multi-Protocol Label Switching) :

II.5.4.1 Présentation :

Il s'agit d'un nouveau standard de l'IETF permettant de simplifier l’administration d’un tel cœur de réseau en ajoutant de nouvelles fonctionnalités particulièrement intéressantes pour la gestion de la qualité de service. Dans le même esprit que l’architecture DiffServ, MPLS permet de réduire le coût des traitements associés au relayage des paquets en les reportant à la périphérie du réseau et en en réduisant la fréquence. Il apporte aussi un mécanisme de routage hiérarchique efficace, c’est-à-dire des tunnels permettant de gérer les réseaux privés virtuels (VPN).

Il permet également de pouvoir acheminer tous les types d'applications, données, audio, vidéo. Ainsi que de différencier le trafic selon les classes de service employées.

L’architecture MPLS est constitué de :

- **Routeur d'extrémité (LER : Label Edge Router)** : situé à la frontière de réseau. Il responsable d’insérer et de retirer le Label à un paquet au moment de son entrée et de sa sortie.

- **Routeur (commutateur) central (LSR : Label Switcher Router) :** responsable de la commutation des paquets en fonction du Label. Dans le coeur de réseau, les LSR lisent uniquement les labels, et non les adresses des protocoles de niveau supérieur c à d les adresses IP.

Les objectifs principaux de *MPLS* sont :

• Permettre un acheminement rapide des paquets *IP* en remplaçant la fonction de routage par une fonction de commutation rapide.

• Faciliter les fonctions d’ingénierie de trafic en fournissant aux opérateurs la maîtrise de l’acheminement des données, qui s’avère très complexe avec des protocoles de routage classiques.

• Implémenter des mécanismes de résiliences aux pannes.

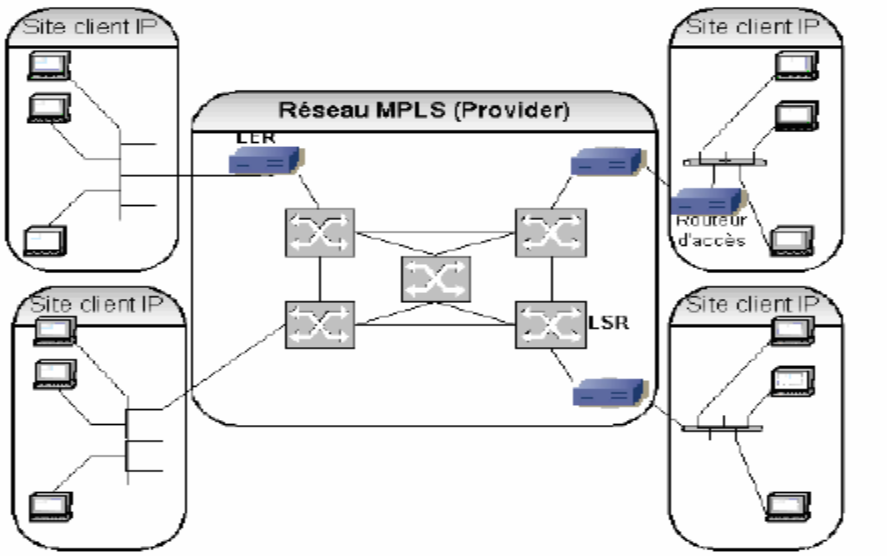


Figure **II.5.4.1 : Modèle MPLS**

II.5.4.2 : Principe de fonctionnement de MPLS

Le principe de MPLS est d'attribuer un label (une étiquette) à chaque paquet lorsqu'il entre dans le réseau. Ce label est attribué en fonction de la classe de relayage (FEC : Forwarding Equivalent Classe) à laquelle appartient le paquet. La définition de ces classes dépend de l'opérateur du réseau, généralement une classe correspond à une entrée de la table de routage ou à un routeur de sortie du réseau. Le routeur à l'entrée décide de la FEC à laquelle appartient un paquet en fonction des informations contenues dans son en-tête (adresse destination, classe de service DiffServ, appartenance à un VPN, ...) et éventuellement de la connaissance qu'il a de la topologie du réseau. Une fois à l'intérieur du réseau les paquets ne sont plus traités qu'en fonction du label qui leur a été associé et l'en-tête IP n'est plus consulté. Ce label décide donc dans chaque routeur : du prochain routeur, du comportement DiffServ et de l'utilisation éventuelle des ressources réservées.

Les tables de commutation qui sont interrogées pour chaque paquet dans chaque routeur peuvent rester de taille réduite puisque le nombre d'étiquettes ne dépend plus du nombre de préfixes annoncés par les opérateurs mais du nombre de routeurs de sortie du réseau. MPLS ne remplace pas le routage IP mais utilise les informations que celui-ci calcule pour établir des chemins entre les routeurs d'entrée et les routeurs de sortie. Les chemins sont établis grâce à une signalisation explicite ou implicite. Le protocole de distribution des labels LDP se charge de la signalisation implicite en établissant automatiquement un chemin pour chaque préfixe contenu dans les tables de routage IP. Les tables de routage ne servent donc plus à relayer les paquets mais à construire les chemins. La commutation de label est plus efficace que le routage IP classique mais elle se fait sur les mêmes bases.

II.6 Intégration avec d'autres services :

II.6.1 Intégration IntServ/DiffServ :

Des travaux sont menés au sein de l'IETF pour faire coexister les architectures IntServ et DiffServ dans l'Internet. Il semble assez naturel que l'approche IntServ soit utilisée dans les réseaux d'entreprise ou les réseaux de petite taille, alors que DiffServ sera utilisé pour les réseaux de transit ou les cœurs de réseaux importants. Nous aboutissons, donc, vers un scénario où la visibilité de l'utilisateur est IntServ, alors que le système de communication utilise DiffServ, rendant ainsi IntServ client de DiffServ. L'intégration de ces deux mécanismes est en cours d'étude.

Plusieurs propositions ont été soumises. La première solution consiste à réaliser l'intégration de service que dans les sites terminaux. Le cœur du réseau ne traite pas les messages de signalisation mais les transmet comme des paquets normaux qui sont à nouveau interprétés dans le site destinataire.
Un contrôle d'admission en bordure du réseau DiffServ permet de déterminer si le flux peut entrer dans la classe de service. L'autre possibilité est de considérer le réseau DiffServ avec la classe EF (Expedited Forwarding) comme un élément de réseau et le caractériser pour permettre de construire un service garanti.

II.6.2 Intégration MPLS/DiffServ :

MPLS permet de simplifier l'administration d'un coeur de réseau en ajoutant de nouvelles fonctionnalités particulièrement intéressantes pour la gestion de la qualité de service. Dans le même esprit que l'architecture DiffServ, MPLS permet de réduire le coût des traitements associés au relayage des paquets en les reportant à la périphérie du réseau. Il apporte aussi un mécanisme de routage hiérarchique efficace, c'est-à-dire des tunnels permettant de gérer les réseaux privés virtuels. Le principe de MPLS est d'attribuer un label à chaque paquet lorsqu'il entre dans le réseau. Ce mécanisme permet d'éviter le calcul complexe de routage classique et doter le monde IP d'un mode connecté. Cette étiquette est attribuée en fonction de la classe de service à laquelle appartient le paquet. La définition de ces classes dépend de l'opérateur du réseau mais elle peut prendre aussi en compte la classe de service DiffServ. L'étiquette décide donc dans chaque prochain routeur, du comportement DiffServ et de l'utilisation éventuelle des ressources réservées. Ainsi la mise en correspondance va consister à associer à chaque classe Diffserv un LSP-MPLS distinct doté de QoS équivalente. Ceci peut être possible en utilisant le champ expérimental de l'entête MPLS pour stocker la valeur de Drop Precedence.

	La solution MPLS	Le modèle Diffserv
Type de services	Une connexion virtuelle (LSP)	Service sans connexion
Routage	Basé sur les labels.	Routage classique (entête IP)
Réservation des ressources	Protocoles RSVP et CR-LDP	Au niveau de DSCP
QoS	Plusieurs contrats (plusieurs labels)	SLA établi avec l'émetteur

Figure **II.6.2 : Tableau comparatif entre MPLS et DiffServ**

II.7 Conclusion :

Dans ce chapitre, nous avons présenté les paramètres de qualité de service pour un réseau IP ainsi que les différents mécanismes en introduisant les trois principaux protocoles de QoS sur les réseaux IP, à savoir IntServ, DiffServ et MPLS ainsi l'intégration de ces protocoles entre eux.

Le chapitre suivant illustre la mise en œuvre de ces protocoles pour garantir la Qualité de Service dans protocole IPv6.

Chapitre III : la qualité de service dans le protocole de nouvelle génération IPng(IPv6) :

III.1 Introduction :

Les réseaux de transmission de données ont pour fonction initiale le transport d'informations numériques entre des ordinateurs distants. Les trois principales utilisations de l'Internet furent d'abord l'accès distant, le transport de fichiers et la messagerie électronique. L'avènement du WorldWideWeb qui permet une navigation transparente au travers de milliards de données stockées à travers le monde, a fait exploser les réseaux et particulièrement la technologie IP. Aujourd'hui, de nouvelles applications voient le jour et se répandent telles que la diffusion des applications de commerce électronique (e-business), l'apparition des applications pair à pair, les applications sensibles au délai telles que la téléphonie IP, la vidéoconférence et les jeux vidéo interactifs, ainsi que les applications temps réel.

Mais face à la limitation d'IPv4 en termes d'adresses, de routage, et de fonctionnalité devant les applications multimédias. Actuellement la taille de l'Internet double tous les 12 mois a causé l'épuisement des adresses IP, l'explosion de la taille des tables de routage, donc IP devient victime de son succès !!, D'où la nécessitée d'un nouveau protocole qui répond a ces préoccupations.

Donc ce nouveau protocole doit répondre aux besoins des utilisateurs en termes de routage, de fragmentation ainsi le contrôle du flux circulant sur un tel réseau.

Dans ce chapitre on va évoquer la notion de qualité de service (QOS) dans ce protocole de nouvelle génération qui est le protocole IPv6 (internet protocole version 6).

III.2 Problématique :

Les réseaux IP classiques offrent un simple service : le service best effort. Un tel modèle de service permet aux routeurs d'être sans état et de ne garder aucune information de grain fin à propos du trafic. L'inconvénient est que, puisqu'il n'y a pas de contrôle d'admission, le réseau peut être perturbé par des utilisateurs trop gourmands. Comme IP est un protocole sans connexion, le concept de contrat de trafic n'existe pas. Si le débit avec lequel le trafic est dirigé sur les interfaces dépasse la vitesse avec laquelle ces mêmes interfaces sont capables d'acheminer le trafic vers l'aval, des congestions peuvent se produire.

Le trafic en excès est placé dans les files d'attente des dispositifs physiques jusqu'à débordement de ces files.

L'utilisation du réseau IP pour transmettre les paquets IP nécessite des performances respectables ainsi qu'une grande stabilité. Une conversation téléphonique par exemple est gravement perturbée par d'éventuels retards ou coupures à cause de congestion généralement. Il faut donc veiller à ce que le flot soit le plus continu possible et que les variations restent faibles.

Afin de pouvoir assurer une bonne exploitation d'IP, les opérateurs doivent fournir un niveau de qualité de service pour éviter la dégradation de la performance de la transmission des paquets qui sont particulièrement sensible au délai de transfert des données et à la perte de ces derniers.

III.3 Les apports d'IPv6 pour la Qualité de service :

Les ordinateurs sont identifiés dans l'Internet par des adresses IP uniques. Le principe du datagramme impose que l'adresse de destination se retrouve dans l'ensemble des paquets émis sur le réseau. Pour permettre un traitement très rapide, les routeurs doivent trouver rapidement cette adresse. Dans IPv4, ces adresses sont codées dans un mot binaire de 32 bits et se retrouvent toujours à la même place dans l'en-tête. Ce principe d'ingénierie a montré son efficacité puisqu'il permet de construire des équipements d'interconnexion simple traitant un nombre important de paquets à la seconde.

Une adresse codée sur 32 bits permet théoriquement d'adresser 2^32 machines, soit à peu près 4 milliards. Ce nombre pourrait paraître au premier abord très élevé, mais les ordinateurs ne sont pas numérotés séquentiellement. Ils sont regroupés par réseaux. À chaque réseau est affecté un numéro qui est codé sur une partie des 32 bit de l'adresse des machines. On s'aperçoit alors que le nombre de réseaux disponibles n'est pas si important que cela conduit à une pénurie. La tendance actuelle consiste à freiner au maximum l'allocation des adresses réseaux. Ce n'est pas un problème pour les sites déjà équipés disposant déjà de larges plages d'adresses. Cette contrainte est déjà forte pour les nouveaux sites dans les pays dits "développés" pour lesquels un grand nombre d'adresses a été réservé mais se révèle être un problème majeur pour les pays émergeants où parfois moins de 10 réseaux de 250 machines ont été attribués pour l'ensemble du pays.

Les équipements d'interconnexion des réseaux, orientant les paquets vers leur destination finale, sont des routeurs. Pour prendre leurs décisions, ils consultent une table dite de routage. Le nombre de réseaux dans l'Internet croissant de manière vertigineuse, ces tables de routage deviennent de plus en plus volumineuses et difficiles à maintenir. Pour pallier ce problème, une solution d'adressage hiérarchique permettent de réunir un ensemble de numéros de réseaux contigus en un seul préfixe a été conçue (CIDR : *Classeless Inter Domain Routing*). En plus de la réduction des tables de routage, CIDR permet aussi de réduire la sur-allocation d'adresses aux sites terminaux, réduisant quelque peu la pénurie d'adresses. Avec CIDR le propriétaire de l'adresse est modifié. Dans les plans d'adressage initiaux, le site était propriétaire de son préfixe, avec CIDR le préfixe devient la propriété de son opérateur, rendant la renumérotation du réseau nécessaire, si le site change d'opérateur. Cet adressage hiérarchique a montré son efficacité opérationnelle et les règles d'adressage actuelles pour IPv6 généralisent ce principe.

Il était devenu impératif de s'attaquer simultanément à ces deux problèmes d'épuisement des adresses disponibles et d'explosion des tables de routage en s'appuyant sur les principes fondamentaux qui ont fait la réussite de l'Internet. C'est à cette tâche que depuis 1992 s'est attelé l'IETF (Internet Engineering Task Force), l'organisme de standardisation de l'Internet, pour définir le protocole IPv6.

Après plus de 10 ans d'efforts de standardisation, les spécifications de base du protocole et les règles d'attribution des adresses sont clairement définies. La plupart des routeurs et des systèmes d'exploitation incluent cette nouvelle version du protocole IP. Il reste maintenant à faire sortir IPv6 des laboratoires et des plates-formes d'expérimentation, à assurer l'interopérabilité avec IPv4 quand cela est nécessaire et à développer de nouvelles applications profitant de cette espace d'adressage quasi-illimité qu'offre IPv6. Un des défis dans les années à venir est d'utiliser IPv6 dans des domaines jusque là ignorés des réseaux (audio-visuel, domotique, automobile,...).

III.3.1 Simplification du format de l'en-tête :

Certains champs de l'en-tête Ipv4 ont été enlevés ou rendus optionnels. L'en-tête est passé de 15 à 8 champs. Ceci réduit donc les coûts de gestion des paquets dans les situations classiques et limite le besoin en bande passante pour cet en-tête.

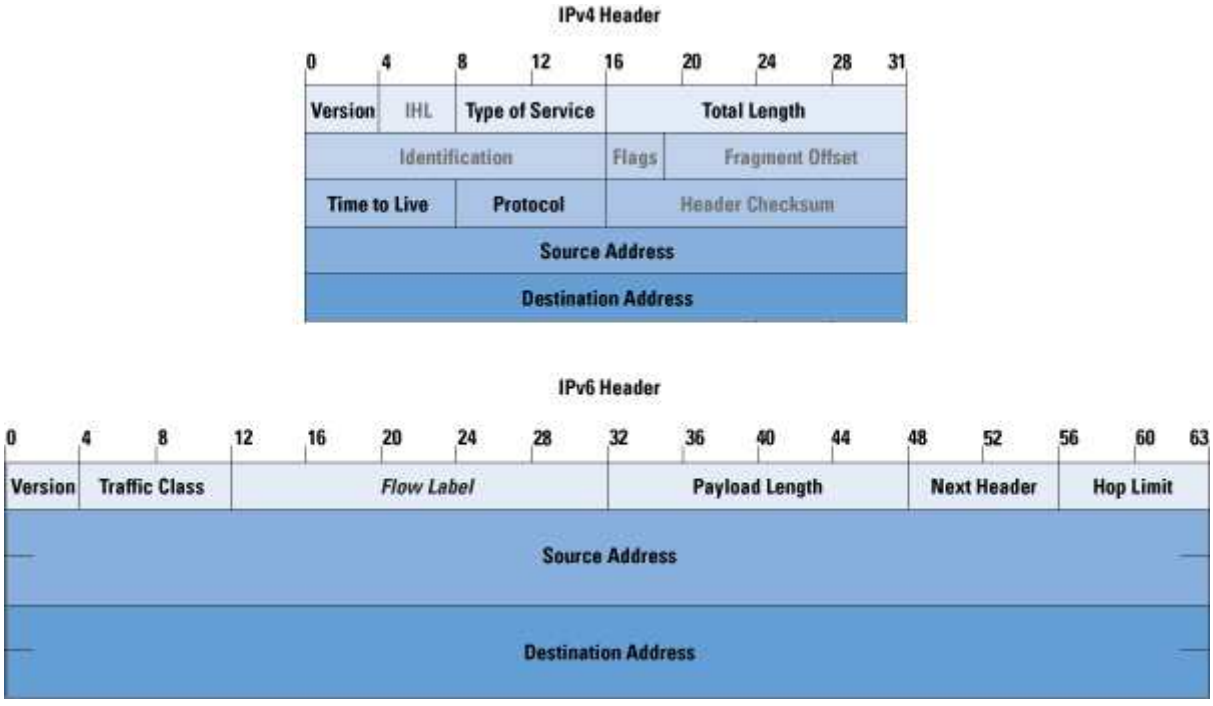


Figure **III.3.1 : l'en-tête IPv4 et IPv6**

III.3.2 Un processus de routage accéléré :

Généralement, les en-têtes d'extension ne sont pas examinés ou traités par un routeur intermédiaire jusqu'à ce que le paquet atteigne le nœud (ou ensemble de nœud en multicast)

identifié par le champ " adresse de destination ".

Il existe cependant une exception : l'en-tête des options "sauts après sauts" (Hop-by-Hop Options Header) transporte les options qui doivent être examinées et traitées par chaque nœud le long du chemin emprunté par le paquet, incluant les nœuds source et destination.

Les différents types de routage sont passés en revue: routage statique, routage interne et routage externe. A l'issue du chapitre, on constatera que IPv6 est maintenant bien intégré dans ces protocoles et que cette évolution a eut un impact très faible pour l'utilisateur final.

III.3.2.1 routage statique :

Le routage statique est le même en IPv6 qu'en IPv4, on ajoutant la contrainte suivante : il n'est pas recommandé d'utiliser une adresse unicast globale comme passerelle en routage.

Dans un routage statique, on peu toujours se ramener a une a une route pas default ainsi que cette dernière Facilite la circulation des données sur un réseau de grande taille, pour atteindre une destination inconnue, et la possibilité de son utilisation si le prochain saut ne figure pas explicitement dans la table de routage.

L'avantage de ce type de routage est la Sécurité par masquage de certains parties d'un Inter-réseau et de réduire un peu la surcharge par rapport au routage dynamique.

La figure III.3.2.1 ci-dessus illustre ce type de routage.

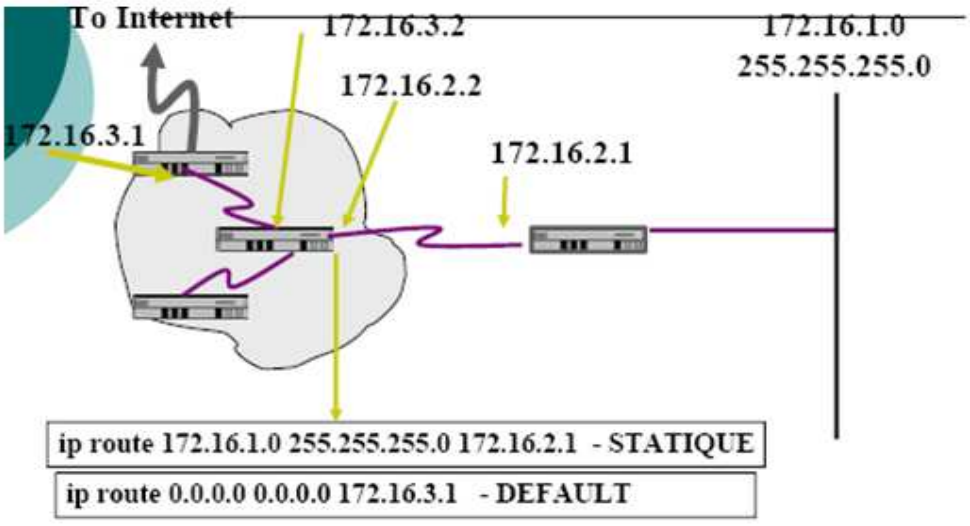


Figure **III.3.2.1 : le routage statique**

Exemples:

- transférer des paquets du réseau 2001:DB8::0/32 vers 2001:DB8:1:1::1 avec une distance administrative de 10.

Router(config)# ipv6 route 2001:DB8::0/32 2001:DB8:1:1::1 10

- Route par défaut au 2001:DB8:1:1::1 *Router(config)# ipv6 route ::/0 2001:DB8:1:1::1*

- Configuration recommandée pour un next hop lien local Router (config)# ipv6 route 2001:DB8::/32 FE80::260:3eff:fe47:1530 interface GigabitEthernet0/1

III.3.2.2 routage interne :

Les protocoles de routage internes permettent une configuration automatique des tables de routage des routeurs à l'intérieur d'un même système autonome. Les routeurs déterminent le plus court chemin pour atteindre un réseau distant. Les protocoles de routage internes nécessitent une configuration minimale du routeur notamment en ce qui concerne les annonces de routes initiées par ce routeur (ex. réseaux directement accessibles par une interface du routeur, annonces statiques ...).

Deux types de protocole de routage interne existent: les protocoles à état de lien ("link state" en anglais) et les protocoles à vecteur de distance ("distance vector" en anglais). Les premiers calculent le chemin le plus court en comptant le nombre de sauts pour atteindre le préfixe de destination, tandis que les seconds attribuent un coût à chaque lien en fonction de divers paramètres (type du lien...).

III.3.2.2.1 RIPng :

RIP (Routing Information Protocol) : Chaque routeur propage toutes les routes qu'il connaît vers les autres routeurs. Chaque routeur possède une table de routage qui comporte une entrée pour chaque destination possible Dans RIPng, un mécanisme de temporisation permet de gérer l'ensemble des évènements. Ainsi toutes les 30 secondes, le processus de routage est réveillé afin d'envoyer un message de type *réponse* contenant la table de routage complète. Ce message est envoyé à tous ses voisins.

Ce type de routage est basé sur l'algorithme de Bellman-Ford.

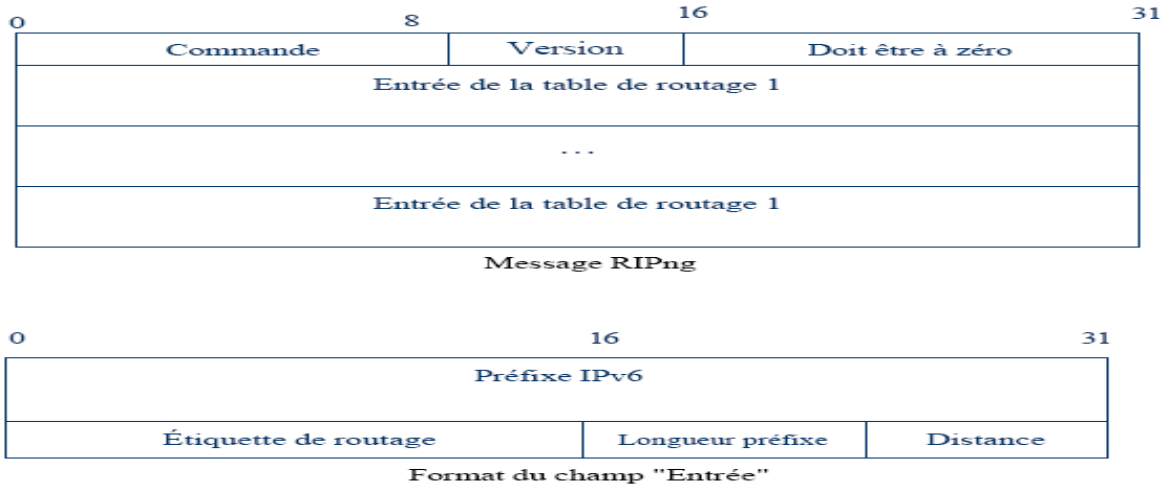


Figure **III.3.2.2.1 : format du message RIPng et champs "entrée" d'un paquet RIPng**

Le format des messages RIPng est le suivant :

- Le champ version est aujourd'hui défini à la valeur 1.

- Le champ commande contient :

1 : pour une requête de table de routage.

2 : pour une réponse à une requête ou une émission périodique.

- Les champs préfix et longueur de préfixe décrivent les réseaux annoncés.

- Le champ métrique comptabilise le nombre de routeurs traversés. Pour résoudre les problèmes de convergence ("comptage jusqu'à l'infini"), la valeur maximale de métrique ("infini") est 255. Cette valeur est largement suffisante pour le domaine d'application du protocole. Le champ métrique peut donc prendre toutes les valeurs comprises entre 0 et 255. Si le champ métrique d'une entrée vaut 0xFF, (cf. figure Format d'un champ "Entrée" d'un paquet RIPng.) le champ préfixe de cette entrée donne l'adresse d'un routeur (prochain saut). Une telle entrée indique que les destinations des entrées suivantes sont accessibles par le routeur explicitement indiqué, et non par celui envoyant le paquet RIPng (utilisation en mode "proxy"). Dans ce cas, les champs "marque de routage" et "longueur du préfixe" sont mis à 0.

Dans RIPv2, cette possibilité était indiquée dans chaque information de routage. Vue la longueur des adresses IPv6, dans RIPng l'indication du prochain saut est valable pour toutes les routes qui suivent jusqu'à la fin du paquet ou jusqu'à la prochaine entrée de ce type. Ainsi, bien que les adresses soient quatre fois plus grandes qu'en IPv4, la taille des informations de routage est la même que dans RIPv2.

III.3.2.2.2 OSPFv6 :

Le protocole OSPFv6 (Open Shortest Path First) cherche à atteindre plusieurs objectifs dont :
- *Routage par type de service* : les administrateurs peuvent définir plusieurs routes, de qualité de service différente, vers une destination donnée. Le routeur choisit alors la route de QoS adéquate pour acheminer les paquets.
- *Equilibrage des charges* : si un administrateur définit plusieurs routes de même QoS vers une destination donnée, OSPF répartit équitablement le trafic sur toutes ces routes.
- *OSPF* permet à un site de décomposer ses réseaux et routeurs en sous-ensembles appelés *zones*. Chaque zone est autonome et la topologie d'une zone reste invisible pour les autres zones.

III.3.2.3 routage externe :

La troisième famille des protocoles de routage concerne le routage externe. Le terme externe vient du fait qu'il s'agit d'un échange d'informations de routage entre les deux domaines d'administration distincts que sont les systèmes autonomes. Ces systèmes autonomes sont de deux types : les systèmes autonomes terminaux (exemple celui d'un client) et les systèmes autonomes de transit (exemple celui d'un fournisseur d'accès IP).

Ce type de routage est assuré par l'architecture MPLS qui définit les trois usages suivants :

- **Transition IPv4 vers IPv6 :** dans ce contexte, MPLS a été identifié comme une technologie permettant le transport de flux IPv6 à moindre coût. En effet, une fois que le paquet IPv6 est encapsulé dans une trame MPLS sur le routeur d'entrée (le PE-routeur en terminologie MPLS), celle-ci est commutée comme toute autre trame sur les routeurs MPLS de cœur (les P-routeurs). Cette méthode, appelée 6PE (IPv6 Provider Edge) permet de connecter des sites distants IPv6 au travers d'un réseau de coeur MPLS IPv4.

- **mise en place des tunnels MPLS :** des protocoles spécifiques (LDP : Label Distribution Protocol, TDP : Tag Distribution Protocol) ou adaptés (RSVP) construisent les chemins MPLS (les LSP : Label Switched Path) sur la base des informations contenues dans les tables de routage interne.

- **réseaux privés virtuels :** les VPN MPLS représentent le service le plus utilisé de la technologie MPLS. Ils permettent le déploiement de réseaux privés (virtuels car une seule infrastructure physique est utilisée) en assurant une étanchéité entre eux, tout comme si chaque réseau était physiquement différent.

III.3.3 Fragmentation :

Ipv6 ne gère pas la fragmentation.

Ipv6 exige que chaque lien inter-réseaux ait une MTU (Maximum Transfert Unit) supérieure ou égale à 1280 octets. Pour tout lien n'ayant pas la capacité requise, les services de fragmentation et de réassemblage doivent être fournis par la couche inférieure à Ipv6.

III.3.4 Labels relatifs aux flux d'informations (Identificateur de flux) :

Ce champ contient un numéro unique choisi par la source, qui a pour but de faciliter le travail des routeurs et la mise en œuvre des fonctions de qualité de service comme RSVP. Cet indicateur peut être considéré comme une marque à un contexte dans le routeur. Le routeur

peut alors faire un traitement particulier : choix d'une route, traitement en "temps-réel" de l'information.

Avec IPv4, certains routeurs, pour optimiser le traitement, se basent sur les valeurs des champs adresses de la source et de destination, numéros de port de la source et de destination et protocole pour construire un contexte. Ce contexte sert à router plus rapidement les paquets puisqu'il évite de consulter les tables de routage pour chaque paquet. Ce contexte est détruit après une période d'inactivité.

Avec IPv6, cette technique est officialisée. Le champ identificateur de flux peut être rempli avec une valeur aléatoire qui servira à référencer le contexte. La source gardera cette valeur pour tous les paquets qu'elle émettra pour cette application et cette destination. Le traitement est optimisé puisque le routeur n'a plus à consulter cinq champs pour déterminer l'appartenance d'un paquet. De plus si une extension de confidentialité est utilisée, les informations concernant les numéros de port sont masquées aux routeurs intermédiaires.

Le groupe de travail MPLS (*Multi Protocol Label Switching*) a intégré les travaux sur le Tag Switching et a précisé la manière dont la commutation des paquets pourra être faite.

L'identificateur de flux d'IPv6 n'est plus utilisé, mais un en-tête spécifique est introduit entre l'encapsulation de niveau 2 et celle de niveau 3. L'identificateur de flux n'a plus à être modifié en cours de transmission. Cette évolution clarifie l'utilisation du protocole RSVP (*Reservation Protocol*) qui peut se baser sur cette valeur, identique tout au long du chemin, pour identifier un flux.

- **Ipv6 et VoIP (utilisation des labels de flux) :**

Outre le fait que les communications IP sont possibles en mode end-to-end, grâce à l'abondance des adresses IP, Ipv6 permet d'améliorer la performance de la VoIP : Le mode end-to-end permet de communiquer entre deux terminaux IP, directement reliés au réseau IP sans passer par le réseau commuté pour aboutir sur un téléphone standard, ni sans transiter par un ordinateur qui transformera les paquets IP en voix : les terminaux sont des " téléphones IP " et transforment la voix en paquets de données et inversement, et sont directement connectés comme des nœuds de réseau.

III.3.5 Classes de trafic :

Ce champ donne un moyen, séparément du Flow ID pour une source, d'identifier la priorité de distribution pour les paquets émis.

Le champ Classe de trafic a été créé pour être utilisé par des nœuds origines et/ ou des routeurs transmetteurs pour identifier et distinguer différentes classes ou priorités de paquets

Ipv6. Cette gestion est effectuée par le protocole Diffserv qui permet d'exploiter ce champ "classe de trafic".

L'émetteur de flux spécialisé (multimédia, temps réel...) spécifie une classe de service à travers le champ classe de trafic. Les routeurs, équipés d'algorithmes (Packet Classifier), interprètent ce champ et mettent en œuvre un traitement différencié (adaptation file d'attente...) à l'aide du répartiteur de paquet (packet scheduler).

Le type de trafic est divisé en deux "gammes".

Les valeurs **0 à 7** sont employées pour étiqueter des paquets à flots contrôlés (par exemple les paquets d'une connexion RTCP).

Les valeurs **8 à 15** sont utilisées pour étiqueter les paquets de flots non contrôlés comme des paquets "**temps réel"** envoyés sans contrôle de flux depuis les récepteurs.

Par exemple, la plus petite valeur (8) devrait être utilisée pour les paquets que l'émetteur est plus disposé à "jeter" dans des conditions de congestion (i.e. trafic vidéo de haute-fidélité). La plus haute valeur (15) pour les paquets que l'émetteur est le moins disposé à "jeter" (i.e. trafic audio de basse qualité).

Ce champ comporte deux parties :

- **DScp** : DiffServ Code Point sur 6 bits qui contient les valeurs des différents comportements.
- **CU** : sur 2 bits, non utilisé.

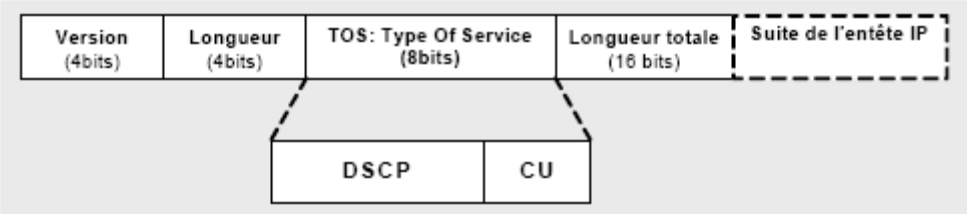


Figure **III.3.5 : Format de l'octet classe de trafic**

Aujourd'hui, seuls deux types de comportement sont standardisés par l'IETF :

- **Assured Forwarding (AF)** : il définit 4 classes de service et trois priorités. Les classes sont choisies par l'utilisateur et restent les mêmes jusqu'au destinataire. La priorité peut être modifiée par les routeurs.
- **Explicit Forwarding(EF) :** comportement équivalent à une liaison louée à débit constant.

III.3.6 Durée de vie maximale du paquet :

Les nœuds Ipv6 ne sont pas obligés d'imposer un temps de vie maximum au paquet. Il y a donc une réduction des pertes d'information du fait d'une absence de paquets rejetés à la fin de cette durée.

III.3.7 ICMPv6 :

Le protocole de contrôle d'IP a été revu. Dans IPv4, ICMP (Internet Message Control Protocol) sert à la détection d'erreurs (par exemple : équipement inaccessible, durée de vie expirée,...), au test (par exemple ping), à la configuration automatique des équipements (redirection ICMP, découverte des routeurs). Ces trois fonctions ont été mieux définies dans IPv6. De plus ICMPv6 intègre les fonctions de gestion des groupes de multicast (MLD : Multicast Listener Discovery) qui sont effectuées par le protocole IGMP (Internet Group Message Protocol) dans IPv4. ICMPv6 reprend aussi les fonctions du protocole ARP utilisé par IPv4.

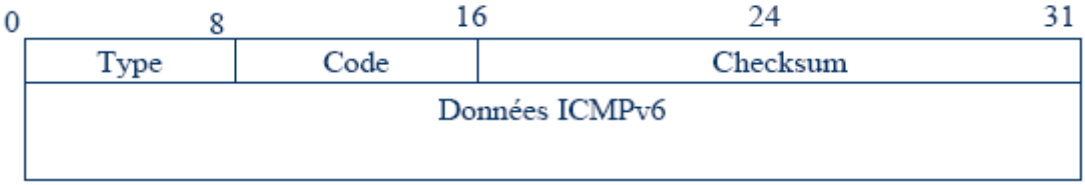


Figure **III.3.7 :** ***Format*** **générique d'un message ICMP**

Le protocole se voit attribuer le numéro 58. Le format générique des paquets ICMPv6 est donné figure Format générique d'un message ICMP :

- Le champ type (voir tableau Valeurs des champs type et code d'ICMPv6) code la nature du message ICMPv6. Contrairement à IPv4 où la numérotation ne suivait aucune logique, les valeurs inférieures à 127 sont réservées aux messages d'erreur. Les autres valeurs réservées aux messages d'information, parmi lesquels se trouvent ceux utilisés par le protocole découverte des voisins (*neighbor discovery*) pour la configuration automatique des équipements.

- Le champ code précise la cause du message ICMPv6.

- Le champ checksum permet de vérifier l'intégrité du paquet ICMP. Ce champ est calculé avec le pseudo-en-tête décrit au chapitre Checksum au niveau transport.

Les messages ICMPv6 de compte rendu d'erreur contiennent dans la partie ‘’données’’ le paquet IPv6 ayant provoqué l'erreur. Pour éviter des problèmes de fragmentation puisqu'il est difficilement envisageable de mettre en œuvre la découverte du MTU, la longueur du message ICMPv6 est limitée à 1 280 octets et par conséquent le contenu du paquet IPv6 peut être tronqué.

Valeurs des champs type et code d'ICMPv6 :

type	code	nature
Gestion des erreurs		
1		Destination inaccessible :
	0	* aucune route vers la destination
	1	* la communication avec la destination est administrativement interdite
	2	* hors portée de l'adresse source
	3	* l'adresse est inaccessible
	4	* le numéro de port est inaccessible
2		Paquet trop grand
3		Temps dépassé :
	0	* limite du nombre de sauts atteinte
	1	* temps de réassemblage dépassé
4		Erreur de paramètre :
	0	* champ d'en-tête erroné
	1	* champ d'en-tête suivant non reconnu
	2	* option non reconnue

III.3.7.1 Messages d'erreur ICMPv6 :

III.3.7.1.1 Destination inaccessible :

Figure **III.3.7.1.1 : Format d'un message ICMP Destination inaccessible**

Ce message est émis par un routeur intermédiaire quand le paquet ne peut pas être transmis parce que soit :

- le routeur ne trouve pas dans ses tables la route vers la destination (code = 0) ;
- le franchissement d'un équipement de type firewall est interdit ("raison administrative", code = 1) ;

- l'adresse destination ne peut être atteinte avec l'adresse source fournie, par exemple si le message est adressé à un destinataire hors du lien, l'adresse source ne doit pas être une adresse lien-local (code = 2) ;

- toute autre raison comme par exemple la tentative de routage d'une adresse locale au lien (code = 3) ;

- le destinataire peut aussi émettre un message ICMPv6 de ce type quand le port destination contenu dans le paquet n'est pas affecté à une application (code = 4).

III.3.7.1.2 Paquet trop grand :

Figure **III.3.7.1.2 : Format d'un message ICMP Paquet trop grand**

Ce message ICMPv6 est utilisé par le protocole de découverte du MTU pour trouver la taille optimale des paquets IPv6 afin qu'ils puissent traverser les routeurs. Ce message contient la taille du MTU acceptée par le routeur pour que la source puisse efficacement adapter la taille des données. Ce champ manquait cruellement dans les spécifications initiales de IPv4, ce qui compliquait la découverte de la taille maximale des paquets utilisables sur l'ensemble du chemin (cf. découverte du PMTU).

III.3.7.1.3 Temps dépassé :

Figure **III.3.7.1.3 : Format d'un message ICMP Temps dépassé**

C’est un format de message ICMPv6 qui indique que le paquet a été rejeté par le routeur :

- soit parce que le champ nombre de sauts a atteint 0 (code = 0) ;

- Soit qu'un fragment s'est perdu et le temps alloué au réassemblage a été dépassé (code = 1).

Ce message sert aussi à la commande trace-route pour déterminer le chemin pris par les paquets.

III.3.7.1.4 Erreur de paramètre :

Figure **III.3.7.1.4 : Format d'un message ICMP Erreur de paramètre**

Ce message est émis par un nœud ayant détecté une erreur de syntaxe dans l'en-tête du paquet IP ou des extensions. Le champ code révèle la cause de l'erreur :

- la syntaxe de l'en-tête n'est pas correcte (code = 0) ;
- le numéro en-tête suivant n'est pas reconnu (code = 1) ;
- une option de l'extension (par exemple proche-en-proche ou destination) n'est pas reconnue et le codage des deux bits de poids fort oblige à rejeter le paquet (code = 2).

Le champ pointeur indique l'octet où l'erreur est survenue dans le paquet retourné.

III.3.8 architecture DiffServ :

DiffServ est un modèle qui permet de classifier chaque paquet selon son contenu, plus précisément selon l'information se trouvant dans le champ Class of Service et Flow Label sur IPV6 de la trame IP et d'appliquer un traitement en fonction de celui-ci.

Dans la figure ci-dessus, Chaque site dispose par le biais de son routeur de bordure d'un point d'entrée à l'ISP, caractérisé par un certain contrat de trafic (appelé « SLA : Service Level Agreement ») établi statiquement, et constitué pour chaque classe de service :

− Des règles de classification.

− Du profil de trafic à respecter en émission (appelé « TCA : Traffic Conditioning Agreement »).

− Des actions entreprises en cas de non respect du profil annoncé.

C'est au routeur de bordure de mettre en œuvre ce SLA lorsqu'il a à introduire des flux applicatifs dans l'ISP.

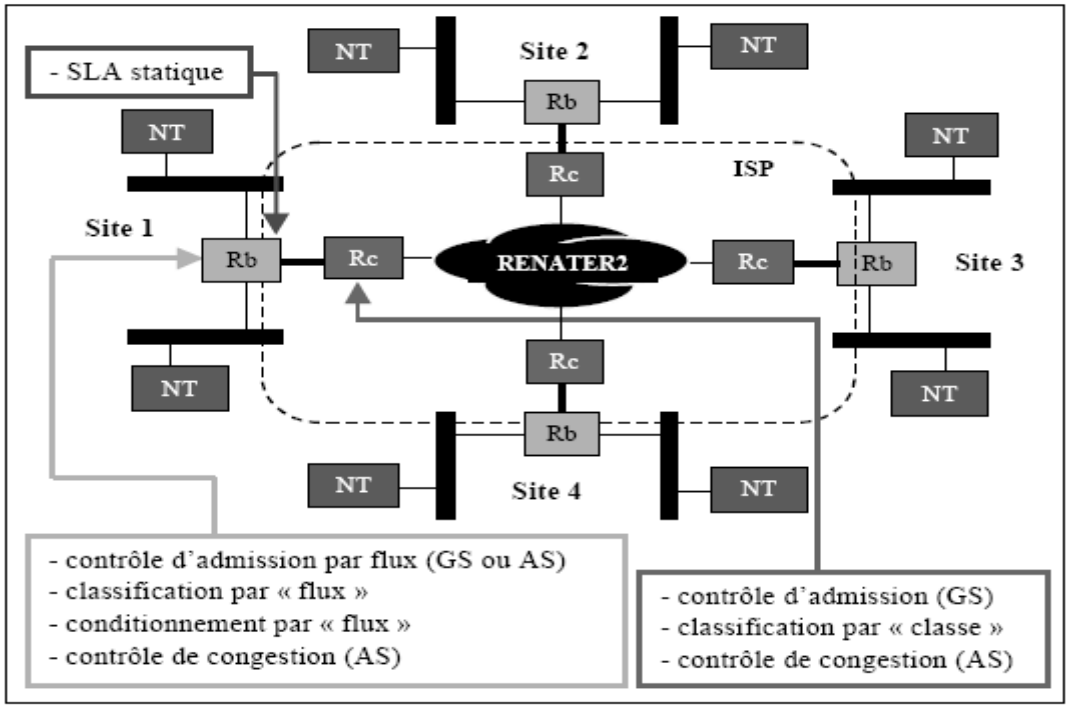


Figure **III.3.8 : Plate-forme d'expérimentation**

III.3.8.1 Rôle des routeurs de DiffServ :

III.3.8.1.1 Rôle des routeurs de cœur :

Le contrôle de trafic appliqué au sein de l'ISP (i.e. dans les routeurs de cœur) est effectué par classe (le marquage, ayant eu lieu au niveau des routeurs de bordure), que ces paquets appartiennent ou non à un même flux.

Trois marques ont été définies (EF : Expedited Forwarding, AF : Assured Forwarding et BE : Best Effort), chacune correspondant à l'un des services supportés.

Plus précisément, le contrôle de trafic mis en œuvre au niveau des routeurs de cœur comportent les fonctions principales suivantes (Figure III.3.8.1) :

− Un contrôle d'admission (pour le service GS uniquement) ;

− Une classification de type BA (*behavior aggregate*) : tous les paquets marqués identiquement (même valeur du champ DSCP) sont rangés dans une même file d'attente (trois files d'attente sont donc définies) ;

− un ordonnancement des trois files précédentes couplant mécanismes de priorité (PQ : *priority queuing*) et « *weighted fair queuing* » (WFQ)

− Un contrôle de congestion (pour le service AS uniquement), destiné à faire diminuer le débit des paquets opportunistes en cas de saturation d'un ou plusieurs routeur de cœur dans l'ISP ; ce contrôle inclut en particulier un mécanisme de rejet sélectif à seuil de type « *partial buffer sharing* » (PBS).

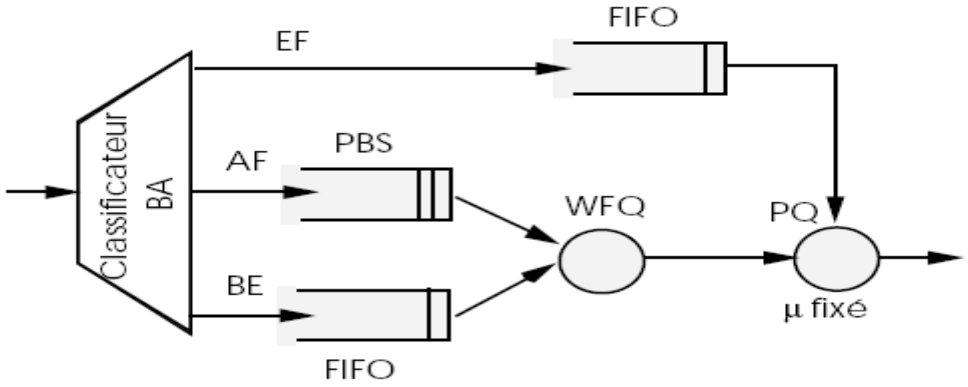


Figure **III.3.8.1 : Eléments fonctionnels d'un routeur de cœur**

III.3.8.1.2 Rôle des routeurs de bordure :

Au sein des sites, les trois services proposés sont également accessibles par les hôtes, qui doivent pour cela (excepté pour le BE) s'adresser à leur routeur de bordure.

Du point de vue d'un routeur de bordure (Figure III.3.8.2), la disponibilité des services AS ou GS est fonction :

− du contrat de trafic (TCA) négocié de façon statique avec le routeur de cœur pour le service considéré ;

− de l'état courant d'utilisation du service considéré par le site.

Ceci étant et indépendamment du service choisi, le contrôle de trafic appliqué par les routeurs de bordure est effectué par flux. En conséquence et de façon à respecter le TCA local : il est nécessaire de mettre en œuvre un contrôle d'admission, que le service requis soit de type AS ou GS.

Les autres fonctions des routeurs de bordure sont les suivantes :

− un conditionnement du trafic par flux, c'est à dire :

− un marquage Diffserv des paquets fonction de la QoS requise pour le flux (EF, AF et BE) ;

− une vérification de la conformité du trafic à la caractérisation annoncée par l'application (le modèle état celui du *Token Bucket*) ;

− un marquage « hors profil » des paquets « opportunistes » (dans le cas du service AS), ou le rejet de ces paquets (dans le cas du service GS) en cas de non-conformité du trafic ;

− une classification par flux (ou de type MF : *multi field*) à partir de plusieurs champs des paquets IPv6, comme par exemple l'identificateur de flot et l'adresse source (trois files d'attente sont définies) ;

− un ordonnancement couplant là encore mécanismes de priorité.

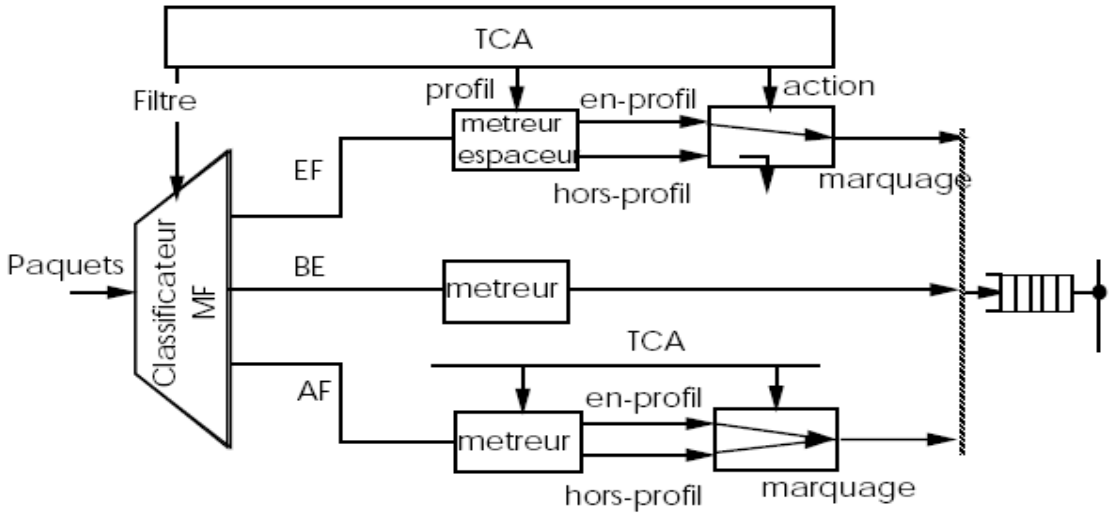


Figure **III.3.8.2 : Eléments fonctionnels d'un routeur de bordure**

III.3.8.2 DiffServ et les mecanismes de contrôle :

Tel qu'introduit dans la section précédente, les principaux mécanismes mis en œuvre au niveau des routeurs de bordure et/ou de cœur pour offrir un service de contrôle d'admission pour chaque classe de service de l'architecture DiffServ :

III.3.8.2.1 cas de EF et AF :

L'hypothèse sous-jacente est que le réseau (l'ISP) est suffisamment bien provisionné pour satisfaire un service assuré sous la seule condition que les différents TCA soient chacun respectés par les routeurs de bordure. Toutefois, le service AS ayant été défini de façon à absorber un débit de paquets « opportunistes » (c'est à dire excédant le contrat initial tant qu'il n'y a pas de congestions dans le réseau), un contrôle de congestion s'avère nécessaire.

III.3.8.2.2 Cas du best effort :

Le service BE utilise la bande passante disponible, non utilisée par les classes AS et GS. Notons qu'une fraction fixe de la bande passante non utilisée par la classe GS est réservée à la classe BE par l'ordonnancement, ce qui assure que ce service est toujours disponible. La valeur précise de cette fraction de bande passante réservée au service BE est laissée à la charge de l'administrateur.

III.4 Conclusion :

Dans ce chapitre, nous avons présenté Les apports d'IPv6 pour assurer l'acheminement des paquets IPv6 en implémentant des caractéristiques de performances pour garantir la Qualité de service, en définissant les paramétrer de la QOS en terme de routage, identificateur de flux, classes de trafic ainsi que Le protocole de contrôle d'IP ICMPv6.

Chapitre IV : approche proposée pour assurer la QoS dans IPv6

IV.1 Introduction :

Dans l'architecture DiffServ, le traitement différencié des paquets s'appuie sur trois opérations fondamentales. La classification des flux en classes de service, l'introduction de priorités au sein des classes et la gestion du trafic dans une classe donné.

IV.2 Mise en œuvre de DiffServ :

Le trafic entrant dans le réseau est classifié et se voit attribué des ressources, en fonction des critères de gestion du modèle de service. Aucune réservation n'est faite, mais un dimensionnement adéquat assure qu'il aura assez de ressources dans le réseau pour les demandes de toutes les applications. Les garanties données par le modèle vont dans le sens du partage des ressources disponibles. Pour offrir un certain niveau de QoS, les classifications donnent un traitement différentiel à des applications sensées d'avoir des besoins plus exigeants c'est-à-dire des priorités.

Une fois le mécanisme de priorité disponible, la mise en place d'une qualité de service suppose la définition des règles pour utiliser ce mécanisme. Pour garantir le respect de ces règles, on emploie une politique qui les impose aux limites d'un périmètre.

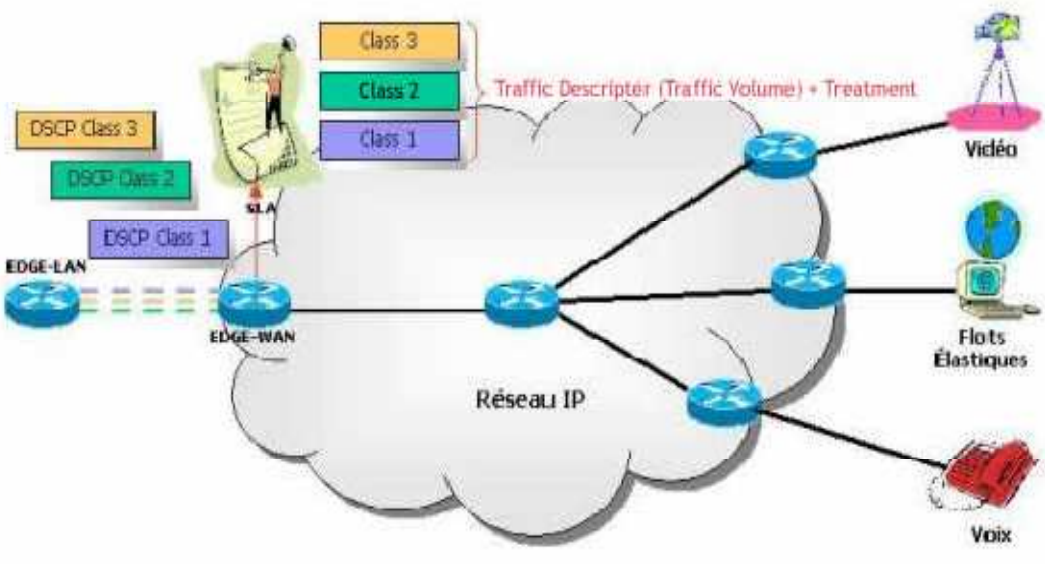


Figure **IV.1 : classification, marquage et conditionnement du trafic**

Dans le cadre de la QoS, nous pourrions tenir compte de différents paramètres qui sont : le délai, le débit et le taux de perte de paquets. Cependant, ces paramètres sont difficiles à mesurer du fait du caractère aléatoire des pertes de paquets et du temps variable passé dans les files d'attente sur tous les routeurs qui constituent le chemin entre une source et une

destination. Il faut donc se limiter, dans le cas d'une mise en œuvre, aux paramètres tels que : le délai, le taux de perte de paquets et la variation du délai de bout en bout.

Le délai de bout en bout détermine le temps que mettent les données pour être acheminée d'une source à une destination, il est composé d'une partie constante : le temps de propagation et une partie variable qui tient compte du temps d'attente dans les différentes mémoires tampons des routeurs traversés.

La variation du délai de bout en bout est due aux délais dans les files d'attente des routeurs.

Le taux de perte de paquets est un facteur important car dans le cas d'un taux trop élevé, pour certain type d'application, il est impossible de réémettre le paquet perdu.

IV.3 Interface amont (interface d'entrée) :

L'interface réseau amont se trouve exclusivement dans le routeur de bordure en entrée du domaine. Elle a en charge le filtrage du trafic entrant. Les éléments fonctionnels de l'interface amont sont représentés par les figure IV.4(a) et IV.4(b). Les règles et paramètres du conditionnement du trafic d'un utilisateur sont décrits dans le TCA. On y trouvera les règles de filtrage pour l'identification de l'utilisateur du service, le profil du trafic et les actions pour le trafic soumis en excès. La classification multicritère (*Multi-field : MF*) repose sur les règles de filtrage. Dans le cas d'IPv6, ce travail d'identification est grandement facilité par le champ *flow label* présent dans l'en-tête des datagrammes.

En effet, un flot applicatif est identifié de manière unique par la paire (adresse source, *flow label*).

Concernant les flots GS, le profil de trafic est constitué uniquement d'un débit crête. La vérification du respect du TCA consiste alors simplement en un métreurespaceur, qui n'est autre qu'une file d'attente de taille finie, dont la vitesse du serveur correspond à ce débit crête. Les paquets acceptés dans la file sont marqués EF, les paquets non conformes sont rejetés. La file d'attente permet d'absorber des rafales de paquets en retardant ces paquets afin de respecter le débit crête ; il faut cependant que la taille de cette file d'attente ne soit pas trop grande pour que le délai d'attente maximal dans cette file soit acceptable.

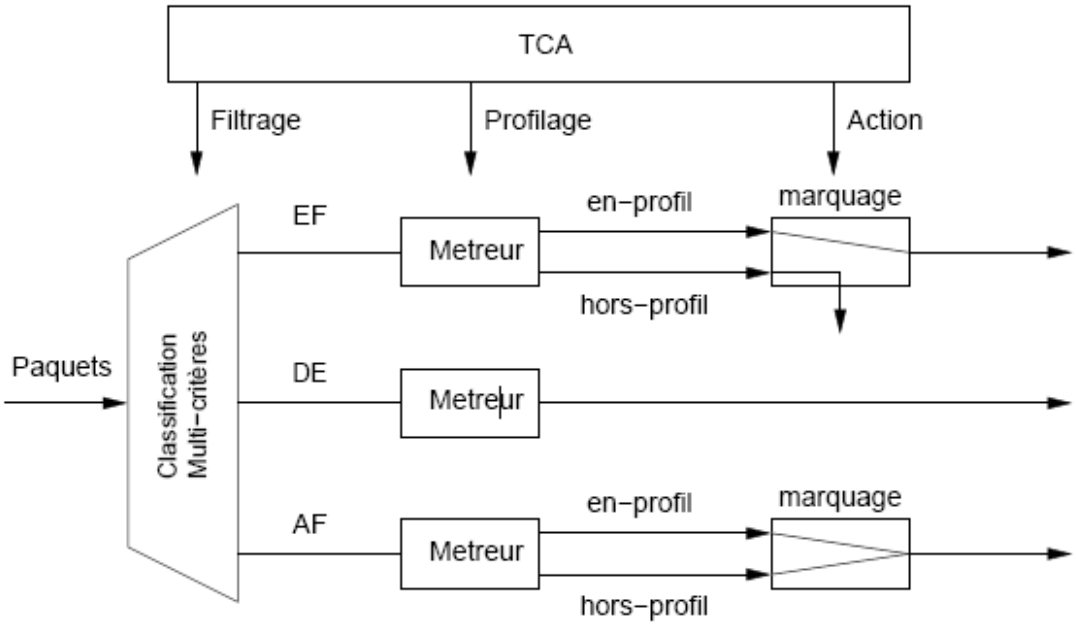
Concernant les flots AS, le profil d'un flot est défini par un débit minimum et par une sporadicité. La sporadicité sert à ajouter une tolérance au débit (en termes de variation) et est représentée par la taille maximale permise de la rafale. Selon cette définition, la sporadicité est exprimée par une quantité de données. Le contrôle de conformité du flot par rapport au profil se fait par un *leaky bucket* (ou saut percé) caractérisé par deux paramètres (r,b), à savoir le taux de fuite du seau et la taille du seau.

Chaque unité de données émise ajoute un jeton dans le seau. Le seau a une capacité de b jetons et fuit à un taux de r jetons par seconde. Quand une unité de données fait déborder le

seau, elle est déclarée non conforme. Les paquets sont marqués AF et une priorité spatiale leur est également attribuée en fonction de leur conformité au profil.

Les paquets détectés conformes reçoivent une priorité à la perte faible (marque *IN*) et les paquets non conformes ont une indication de forte priorité à la perte (marque *OUT*). Ces derniers deviennent des paquets opportunistes. La priorité à la perte est utilisée quand le paquet passe par des routeurs saturés. En l'absence de congestion, cette priorité ne sert pas.

Concernant les flots BE, aucun contrôle particulier n'est effectué : tous les paquets sont marqués DE. Le métreur n'a qu'un rôle d'information pour l'administration de réseau.



Figur*e* **IV.3 : Éléments fonctionnels d'une interface d'entrée d'un routeur de bordure**

IV.4 Interface de sortie :

L'interface de sortie d'un routeur de bordure ou bien de cœur comprend l'ensemble des mécanismes de relayage. Ces mécanismes ont en charge l'ordonnancement des paquets et la gestion des pertes en cas de congestion. La figure 3 décrit la solution retenue et développée. A l'entrée de l'interface, les paquets sont classés dans l'une des trois files d'attente. Cette classification est effectuée en fonction de la marque attribuée par le conditionnement lors de leur entrée dans le domaine. Une telle classification est dite BA (*Behavior Aggregate*), ou autrement dit selon la classe de service. La notion de flot individuel n'existe plus.

La politique d'ordonnancement s'appuie sur 2 ordonnanceurs mis en étage afin de pouvoir gérer les trois PHB de l'architecture. Le PHB EF de la classe GS est obtenu par une priorité stricte (non préemptive) notée PQ (*Priority Queuing*), ce mécanisme assure un traitement

rapide des paquets marqués EF. Le délai de paquétisation[8] est de *Lmax/c* où *Lmax* est la taille maximale d'un paquet EF.

Un ordonnancement unique de type WFQ (*Weighted Fair Queuing*) introduirait un délai de paquétisation de *Lmax/r* où *r* est la bande passante allouée au PHB EF (avec $r<c$).

La motivation de PQ se trouve dans la faiblesse du délai de paquétisation ajoutée par rapport à WFQ.

Le PHB EF est destiné à un service exigeant sur les délais. Le mécanisme retenu donne le relayage le plus rapide. Il protège le trafic de la classe GS du trafic des classes AS et BE, à l'exception du cas où il faut qu'un paquet de la classe GS attende la fin du service en cours d'un paquet d'une autre classe.

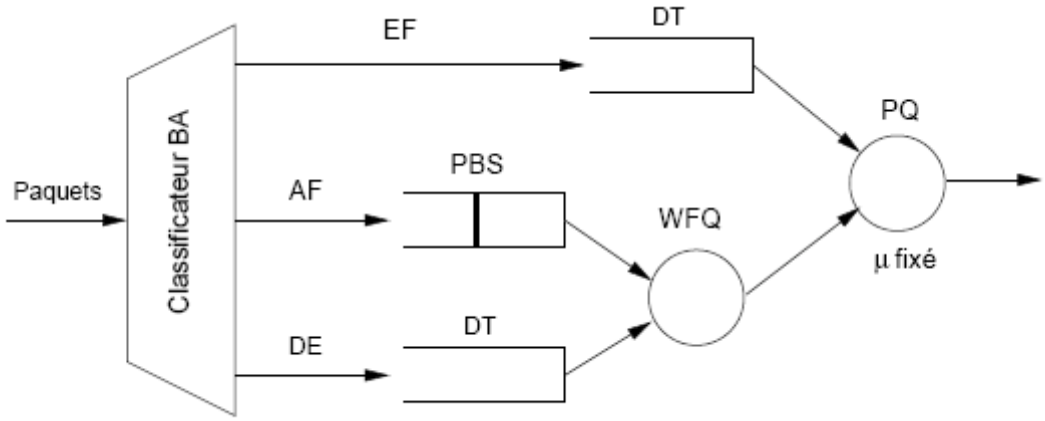


Figure **IV.4 : Éléments fonctionnels d'une interface de sortie**

Donc pour chaque paquet IPv6 entrant, l'intervention d'un routeur de bordure est immédiate, il va classifier chaque flux en classes de service par l'identificateur de flux ainsi l'adresse IP de l'entête IPv6.

Pour les agrégats EF, le contrôle d'admission est nécessaire pour des mesures de conformité a un profil définit par le fournisseur d'accès IP.

Pour les agrégats AF, le contrôle de cogestion est nécessaire pour définir la régulation des paquets non conforme.

Tandis que flots BE, aucun contrôle particulier n'est effectué : tous les paquets sont marqués DE. Le métreur n'a qu'un rôle d'information pour l'administration de réseau.

Et pour les interfaces de sortie des routeurs (cœur et bordure) une gestion de traitement des files d'attentes en terme d'ordonnancement est nécessaire, La figure ci-dessus illustre ces étapes :

[8] Retard induit par le fonctionnement en paquet par rapport à un modèle fluide.

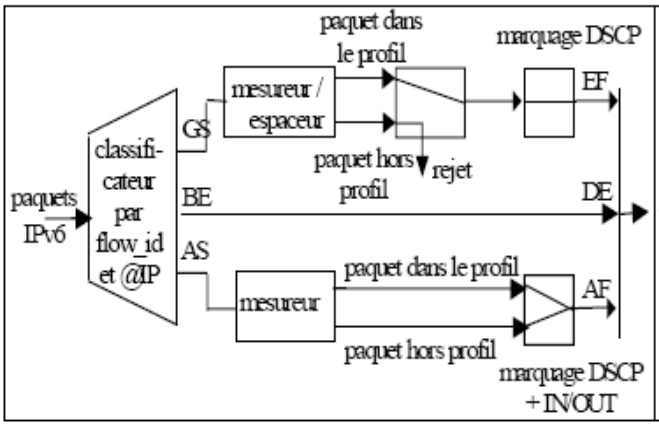


Figure **IV.4(a)** : **Interface d'entrée des routeurs de bordures**

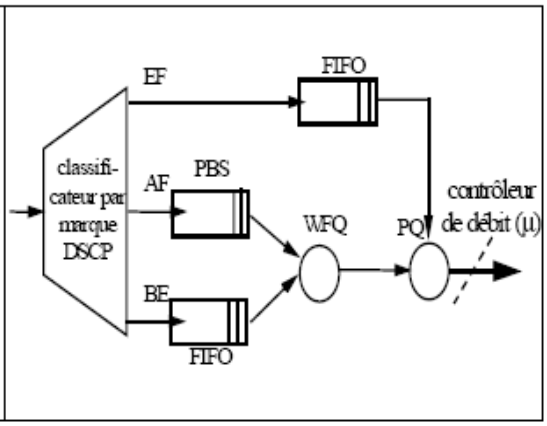


Figure **IV.4(b)** : **Interface de sortie de tous les routeurs (bordure et cœur)**

IV.5 implémentation d'un scenario :

Pour l'implémentation, deux types de flux seront utilisés dans chacun de deux scénarios : Un flux qui modélise le trafic EF/AF (flux prioritaire) et un flux qui modélise le trafic BE (flux non prioritaire), donc deux files d'attentes seront gérées, une pour chaque type de flux (EF, BE).

Ces files sont servies par un Ordonnanceur suivant le modèle PQ où la file de la classe EF a une priorité plus haute que la file BE.

Le premier scénario/démonstrateur **(figure** utilisé afin de tester cette implémentation, représente un domaine. Il est constitué de trois nœuds interconnectés, deux nœuds de bordures et un nœud de cœur. Le trafic est généré par deux sources clients, Sender1 et Sender 2. Ils sont connectés aux routeurs de bordure, edge1 et edge2.

Ces deux nœuds de bordure contiennent des TCBs comportant la classification du trafic, le policing et le marquage. Deux récepteurs, receiver1 et receiver2 connectés avec le routeur de cœur, reçoivent les flux envoyés respectivement par sender1 et sender2. Le routeur de cœur implémente une « priority scheduling » algorithme pour les agrégats selon leurs priorités.

Le deuxième scénario (figure 30) est plus complexe due à la nécessité d'utilisé un classifier multi field aux routeurs de bordure. Pour ce deuxième scénario, le premier émetteur génère un flux AF, par contre le deuxième génère un flux best effort.

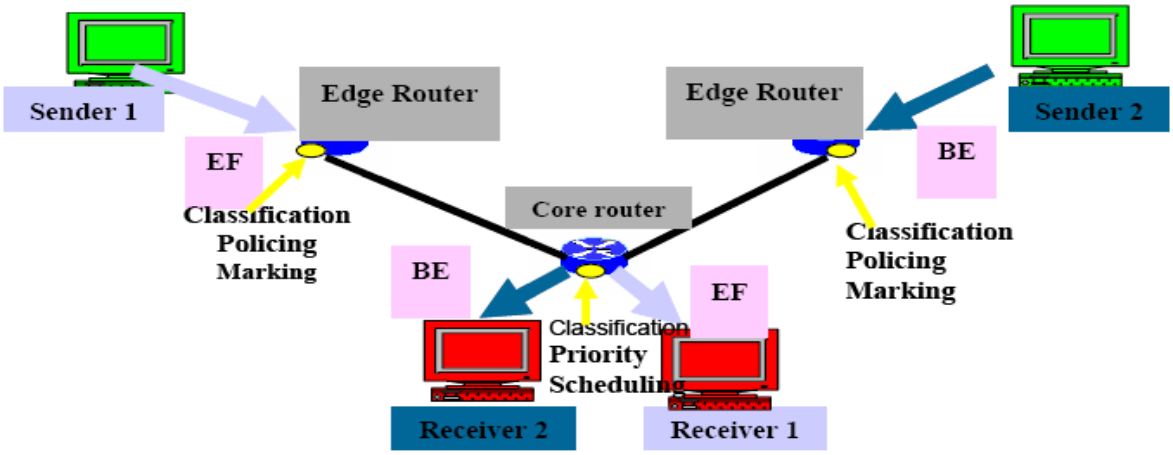


Figure **IV.5(a) : Scénario (1) de différentiation des services.**

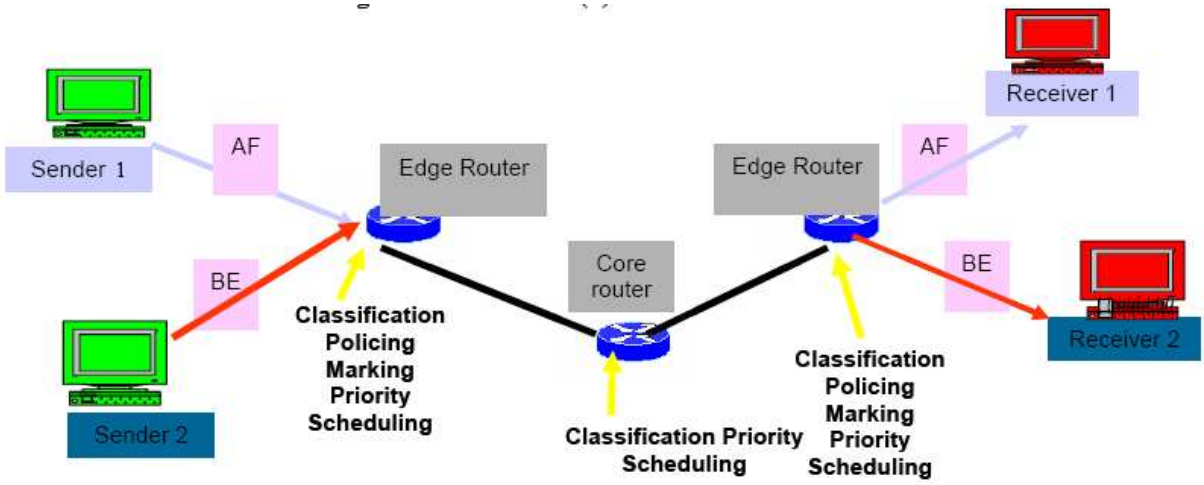


Figure **IV.5(b) : Scénario (2) de différentiation des services.**

IV.6 Les discipline pour les routeurs :

IV.6.1 Les disciplines pour les routeurs de bordure :

Le Classificateur par flux (Multi field classifier): Il est utilisé aux routeurs de bordure. Dans le premier scénario, dés qu'une seule application à la station du client génère du trafic, on n'a pas besoin des filtres de classification aux routeurs de bordure.

Le Marker : Dans notre Démo, on ne mesure pas le trafic avant de marquer ses paquets. Les paquets classés par le classifier seront directement marqués par le DSCP correspondant à leurs classes de service. Ces DSCP sont les suivants : « 000000 » pour le trafic BE et « 101110 » pour le trafic EF

Le Policer/ dropper : On utilisera dans notre Démo la discipline « Absolute Dropper » où tous les paquets qui arrivent à son entré seront rejetés.

Le Scheduler : Pour le premier scénario, une discipline de « FIFO » est appliquée aux routeurs de bordure. Par contre, au routeur de cœur, une discipline de « Priority Queuing » est utilisée.

IV.6.2 Les disciplines pour les routeurs de cœur:

Le classifier par agrégat : un classificateur BA sert à classer les paquets selon leurs DSCPs.

Le *scheduler* : Le traitement différentié dans le cœur d'un réseau DiffServ implique deux axes : les classes de service et la notion de priorités. Chacune des files d'attente dans le routeur représente une classe de service. Leurs propriétés, en termes de bande passante ou de retard moyen, dépendent du mécanisme de scheduling. Pour notre démonstrateur, une discipline de priority queuing a été choisie.

Priority Queuing, PQ

Le modèle PQ utilise plusieurs files d'attente. Les paquets classifiés sont mis dans l'une des files correspondant à sa valeur du DSCP. Les files sont ensuite servies suivant un algorithme spécifique. Celle qui contient les paquets avec la plus haute priorité sera favorisée par rapport aux autres files. Cependant, l'algorithme PQ induit une dégradation de performances. Lorsque le trafic classé prioritaire est anormalement élevé, le trafic non prioritaire peut être rejeté par manque de buffer.

IV.7 conclusion :

Pour assurer la qualité de service QoS pour le protocole IPv6 l'approche de DeffServ est mise en service pour garantir et assurer les caractéristiques de la qualité de service souhaité par le client en termes de SLA.

La configuration des routeurs de bordure et cœur de l'architecture DeffServ est un impact essentiel pour assurer le bon déroulement de bout en bout des datagrammes IPv6.

Comme on a vu un exemple d'implémentation d'un scenario de DeffServ en termes de classificateur, markage et la gestion de l'ordonnancement de ces agrégats.

Conclusion générale

L’Internet, comme la majorité des réseaux en mode paquets, n’a pas été initialement prévu pour prendre en compte les paramètres de qualité de service, donc il a fallu mettre en place de nouveaux mécanismes de qualité de service par les opérateurs et les équipementiers.

Pour garantir la qualité de service il y’a cinq paramètres techniques à prendre en compte qui sont le temps de réponse ou latence, le débit, le taux d’erreur et la gigue et la disponibilité du réseau.

Le but de ce projet est de mesurer les paramètres de la qualité de service dans le protocole IPv6 en implémentant les trois architectures IntServ et DiffServ et MPLS pour assurer un acheminement optimale des paquets IPng circulant dans un réseau, en définissant les apports d'IPv6 de la QOS en terme de routage, le contrôle de flux ainsi l’acheminement des ces paquets.

D’après l’etudes des chapitres L’implémentation de la qualité de service coûte plus de temps, l’architecture IntServ convient plutôt aux réseaux de petite taille, le contrôle de flux se fera avec RSVP, dans DiffServ le contrôle se fera a base d’agrégats I.E par classes de flux, et pour MPLS se fera a base d’étiquettes.

La qualité de service est un mécanisme sensible aujourd’hui pour la transmission des paquets IP en temps-réel en implémentant des applications qui nécessite moins de gigue, taux de perte négligeable, une bande passante plus importante, et un délai très court.

IPv6 répond au contraintes précédentes en terme de routage, de traitement d’erreurs, et le bon acheminement des données transmis, mais avérai dire que l’optimisation de la QoS n’est pas atteint au sens complet du terme, peu être cette optimisation ca sera un jour dans IPv8 et IPv16 les successeurs de IPv6 !!!!??

Bibliographie

Webgraphie:

[Quality of Service Forum]
www.qosforum.com

[Groupe de travail QoS de Internet]
www.internet2.edu/qos

[Les approches successives de la QoS dans IP]
www.ittc.ukans.edu/~rsarav/ipqos/ip_qos.htm

[Introduction to QoS in IPv6]
http://www.urec.fr/IPv6/
http://www.ipv6.imag.fr
http:// www-guill.net
http:// www-memoireonline.com

[Les principes de la QoS]
http://www.rli.cran.u-nancy.fr/sci/biblio/lesprincipesdelaqos.html

[Informations sur IPv6 en général]
http://hs247.com/

[Pages d'information sur le protocole IPv6]
http://www.ipv6.org/

[Groupement d'industriels pour promouvoir IPv6]
http://www.ipv6forum.org/

[Liste des mises en œuvre d'IPv6 dans les équipements]
http://playground.sun.com/pub/ipng/html/

Articles :

QoS dans les réseaux IP
http://www.prism.uvsq.fr/~mea/cours/pdf/dea/RM_qos3.pdf
ftp://ftp.uar.net/pub/e-books/Administering Cisco QoS.pdf
http://www.nanog.org/meetings/nanog36/presentations/sathiamurthi.pdf

QoS in IPv6
http://www.6init.org/public/iir_qos04.pdf

QoS support in IPv6 environments
http://www.6diss.org/workshops/see-1/qos.pdf
http://www.cisco.com/en/US/technologies/tk648/tk872/technologies_white_paper0900aecd8026004d.pdf

généralités sur les réseaux
http://tstri.com/v3/download/RI/Cours%20g%C3%A9n%C3%A9ralit%C3%A9%20sur%20les%20reseaux.ppt
http://ecogestion.scola.ac-paris.fr/disciplines/stg/ftpdoc/FichesReseau.pdf
http://christian.caleca.free.fr/pdf/Les%20Reseaux.pdf
http://langevin.univ-tln.fr/CDE/REZO/ppt/rip.ppt
http://www2.ac-lyon.fr/serv_ress/reseau/telechargement/doc_formation/generalites_reseaux.pdf