

## La gestion des qualités de services dans Internet

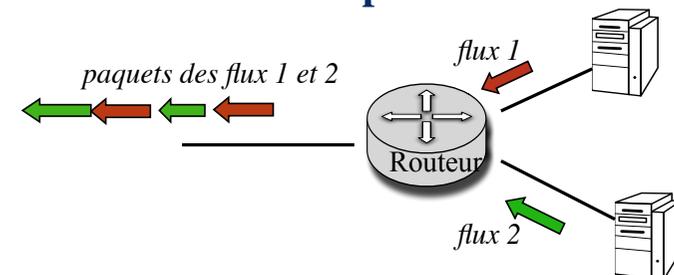
## Introduction

- Aujourd’hui les applications (en particulier multimédia) nécessitent des qualités de service de natures très différentes
- Exemples:
  - Transfert de fichier: Taux erreur nul, débit n’est pas primordial, le temps de transit non plus
  - Téléphone: Taux d’erreur peut être non nul, débit minimum indispensable, temps de transit minimum ( $< 0,25$  s)
- Dans l’idéal choix d’une QoS désirée par l’utilisateur du réseau (concepteur d’application):
  - Débit, latence, perte de paquets, temps de réponse à une requête
  - Qualité plus “fines”: variation du débit (gigue)....

## La garantie de QoS

- Pour garantir des QoS, il faut mettre en place des mécanismes permettant de traiter de manière **différenciée** les différents flux de paquets dans les organes du réseau (routeurs principalement)
- L’ordonnancement des paquets (scheduling) permet de décider de l’ordre de traitement des paquets dans un routeur
- Une bonne gestion de la congestion dans le réseau est primordiale pour les garanties de QoS. En effet les problèmes de garantie de QoS arrivent lorsque le réseau est “chargé”

## Exemple



- Ordonnancement des paquets dans le flux de sortie ?
  - Equité
  - priorité, mais sur quels critères ?
- Que faire quand la bande passante de la ligne de sortie n’est plus suffisante pour supporter la somme des débits des flux 1 et 2 (cas de congestion)?

## La garantie de QoS non prévue dans Internet

- Garanties de QoS non résolues dans les protocoles initiaux d'Internet
  - TCP/IP prévus seulement pour les transferts de fichier
- Malheureusement IP est en mode Datagram (traitement individuel de chaque paquet), un mode avec connexion (circuit virtuel) serait plus adapté
- Dans IP, les paquets sont traités indépendamment les uns des autres, il est donc difficile de “prévoir” les flux et garantir des débits, latences...

## Garanties de QoS au niveau de la couche Transport

- Niveau Transport TCP inutile pour beaucoup d'application “temps-réel”
  - Exemple: le téléphone
    - Les paquets perdus et re-émis arrivent trop tard
    - Un taux d'erreur non nul est possible
  - Utilisation de UDP
- Une couche supplémentaire entre UDP et l'application: RTP (Real Time Protocol) normalisé
  - Numérotation des paquets, estampillage temporel, rapports du récepteur à l'émetteur pour signaler: la QoS courante (délai de transit, taux d'erreur, débit...)
  - Adaptation par l'application (changement de taux de compression, correction à l'arrivée, tampon d'amortissement à l'arrivée ...)

## La garantie de QoS Au niveau de la couche Réseau

- Deux solutions envisagées:
  - Garantie stricte: demande de réservation de ressource dans un routeur et traitement particulier des paquets d'un flux. Permet aussi de prévenir les congestions
  - “Au mieux”: traitement des paquets “au mieux” suivant un marquage effectué par la source. Le marquage permet de différencier des classes de services et des priorités dans ces classes
- Groupes de travail de l'IETF (Internet Engineering Task Force)
  - **IntServ**: intégration de service
  - **DiffServ**: différenciation de service

## Réservation de ressources

- IntServ (intégration de services)
- Protocole RSVP (Resource ReSerVation Protocol):
    - permet d'éviter la congestion et de garantir des QoS
    - Réservations de ressources (mémoire) dans les routeurs par des messages particuliers de signalisation depuis la source
    - Modification de QoS à la demande du (ou des) récepteur; c'est lui qui connaît la QoS dont il a besoin
    - QoS possible: garantie de débit minimal, latence maximale, gigue (variation de la latence) maximale
  - Complexe et coûteux
  - Ne tient pas à grande échelle; Nécessite la maîtrise de l'ensemble des routeurs.

## Différentiation de services (Groupe DiffServ)

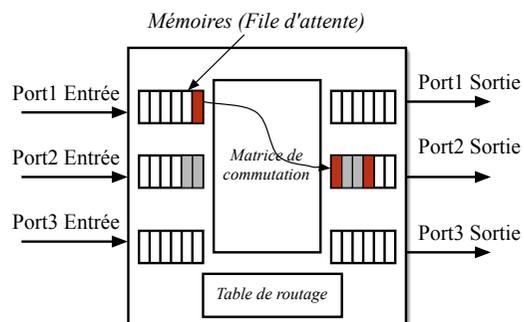
- Définition de classes de services de qualités différentes
- **Marquage des paquets et traitement prioritaire** dans les routeurs
  - Au départ, Champ dans l'entête IP permettant de donner des priorités de délai, débit, fiabilité, coût
  - Aujourd'hui remplacé par la désignation d'une classe de service
  - Prévu aussi dans IPV6
- Possibilité de domaine "Diffserv": ensemble des routeurs implémentant une politique commune (exemple : Renater, fournisseur d'accès)

## Classes de services

- Exemple dans RENATER 4 :
  - **LBE** (Less than Best Effort): Applications non sensibles à la gigue ou à d'éventuelles pertes de paquets
  - **BE** (Best Effort): Toutes applications non sensibles à la gigue
  - **BBE** (Better than Best Effort): Applications nécessitant un débit garanti et une faible gigue, et applications interactives
  - **IP Premium**: Applications nécessitant un débit garanti, une gigue nulle ou extrêmement faible, et aucune perte de paquets
  - **Supervision** / contrôle du réseau RENATER-4: Applications de supervision

## Principe de fonctionnement d'un routeur

- Un routeur possède plusieurs ports d'entrée et de sorties
- Chaque port dispose d'une mémoire (file d'attente)
- Une matrice de commutation permet le passage d'un paquet se trouvant dans une file d'attente en entrée à une autre en sortie
- La table de routage permet de décider de la commutation



## Temps de traversé et congestion d'un routeur

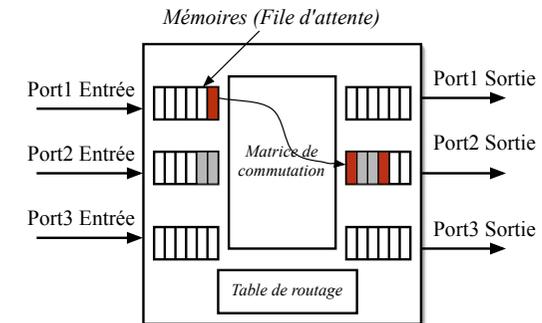
- Le temps de traversée d'un routeur dépend de sa charge : nombre de paquets contenu dans les files d'attentes
- Possibilité de saturation du routeur : files d'attente pleines
- Cas de congestion : destruction des paquets qui ne peuvent être mémorisés

## La commutation dans un routeur

- Un simple ordinateur peut être un routeur
- Le processeur prend les paquets dans une file d'entrée quand la carte "réseau" lui signale une arrivée (par interruption)
- Après traitement le processeur écrit dans une file de sortie d'une autre carte "réseau"
- Dans les vrais routeurs (modernes), un processeur est associé à chaque port d'entrée et les traitements sont faits en parallèle
- Un processeur central assure le maintien de la table de routage et l'administration du routeur
- La commutation peut ensuite se faire :
  - A l'aide d'une mémoire commune
  - A travers un bus partagé
  - A travers un réseau d'interconnexion (crossbar)

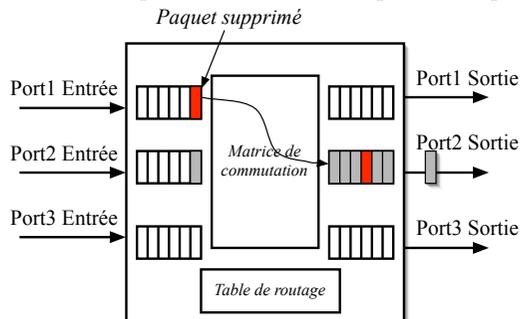
## L'ordonnancement dans un routeur

- Traitement "basique":
  - Chaque file d'attente est examinée successivement
  - Si un paquet est présent en entrée, il est placé dans la file de sortie décidée par le routage
  - Les paquets sont émis dans leur ordre d'arrivée dans la file de sortie (FIFO)



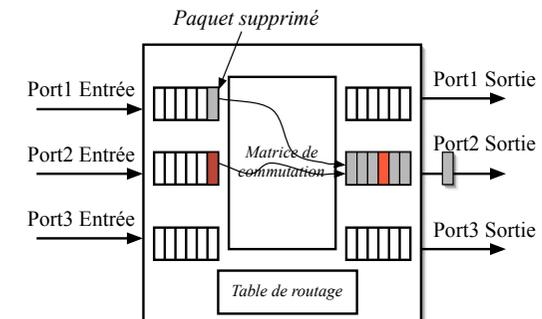
## Perte des paquets dans un routeur Gestion FIFO

- La plupart du temps:
  - Le routeur arrive à traiter les paquets en entrée à leur rythme d'arrivée
  - Les files d'entrée contiennent toujours au maximum un paquet
- Supposons 2 flux entrant sur deux ports dirigés vers un seul port de même débit
- La file de sortie peut se remplir très rapidement, les paquets sont supprimés au moment de la commutation quand la file de sortie est pleine ("drop tail")



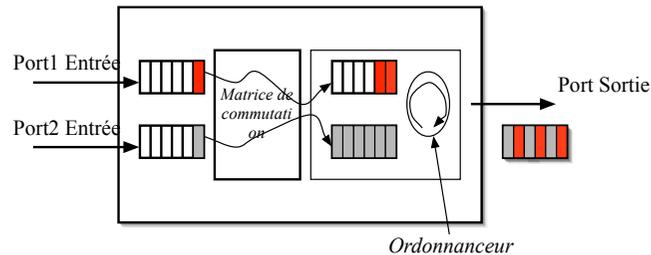
## Perte des paquets dans un routeur Gestion FIFO

- On peut arriver à des situations où plusieurs paquets d'une file d'entrée seront supprimés pendant que les paquets de l'autre trouveront une place libre (émission de paquet non synchronisé)
- Non équité entre différents flux, surtout dans le cas de flux de débits très différents
- Les flux à gros débits auront un pourcentage de perte beaucoup plus petit



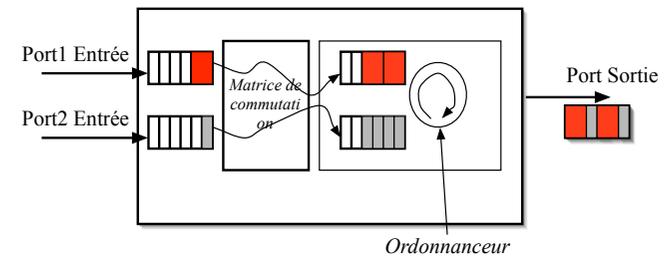
## Amélioration de l'équité dans un routeur

- Traitement d'attente équitable : FQ (Fair Queueing)
- Une file d'attente supplémentaire par flux d'entrée est défini sur l'interface de sortie
- Un ordonnanceur prend de façon cyclique un paquet dans chacune de ces files de sorties
- Le routeur partage ainsi de façon équitable les flux entrants (équité en nombre de paquet)
- En cas de saturation les flux à gros débits auront plus de pertes



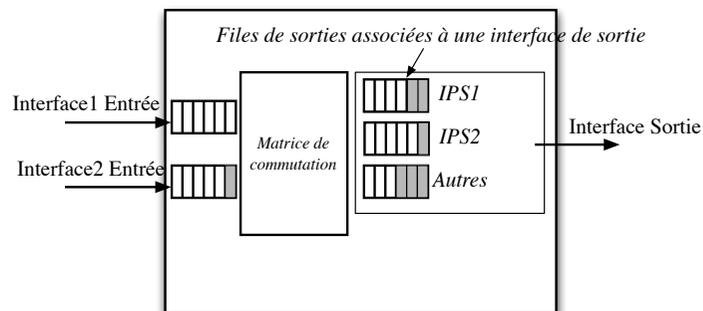
## Amélioration de l'équité dans un routeur

- Un flux à gros paquet est privilégié en terme de débit
- Il existe aussi des ordonnanceurs pour partager la ligne de sortie en terme de nombre d'octets (plus équitable en terme de débit)



## Traitement de la QoS dans un routeur

- Sur le même principe, on peut décider de différents critères pour la création des files d'attente sur une interface de sortie:
  - Adresse IP source ou destination (Machine)
  - Port destination (entête TCP) : suivant une application précise
  - Protocole de niveau supérieur
  - Marquage des paquets dans l'entête IP (classes de services associées "Diffserv")
  - ....



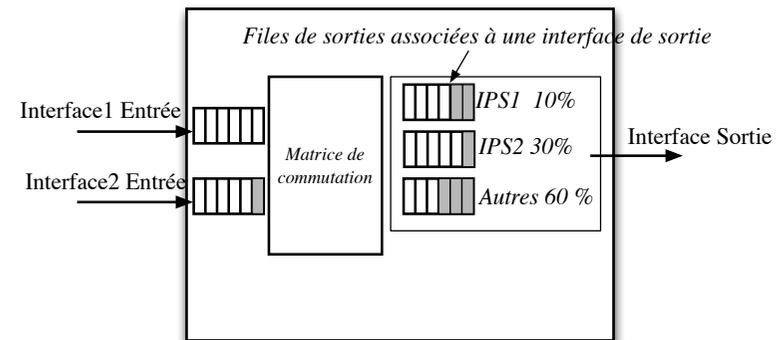
## Traitement de la QoS dans un routeur

- On peut ensuite donner des critères différents que "chacun son tour" (équité) à l'ordonnanceur pour privilégier certains flux (files d'attente):
  - Priorité absolue
  - Limitation du débit
  - Définition de flux pondérés: WFQ (Weighted Fair Queueing)
- Intéressant pour privilégier :
  - une application "Temps réel"
  - une machine source par rapport à une autre
  - un protocole par rapport à un autre
  - ....

## Le WFQ (flux pondérés)

- Chaque file d'attente  $F_i$  se voit attribuer un poids  $W_i$
- On veut partager entre plusieurs flux d'entrée la bande passante de sortie (suivant les poids affectés aux flux)
- L'ordonnanceur "donne" une partie de la bande passante de sortie à chaque file égale à  $W_i$  sur la somme des poids des files ayant des paquets en attente
  - Pour chaque file à traiter le nombre d'octet à émettre (avant de passer à la file suivante) est calculé en fonction de ces poids
- Ainsi si à un moment donné, un flux est seul à transiter, il peut utiliser 100 % de la bande passante

## Principe du WFQ (Weighted Fair Queueing)



## Exemple du WFQ

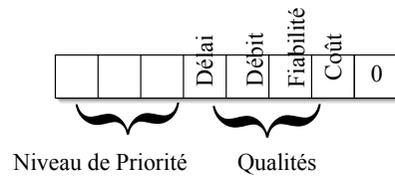
- Exemple: Trois flux (suivant par exemple le port d'entrée):
  - $F1: W1= 10, F2: W2= 30$  et  $F3: W3= 60$
- Les trois flux sont présents, le débit de sortie est partagé ainsi:
  - 10% pour le flux 1
  - 30% pour le flux 2
  - 60% pour le flux 3
- Le flux 3 disparaît (plus de paquet en attente), le débit de sortie est partagé ainsi:
  - $10/40=25\%$  pour le flux 1
  - $30/40=75\%$  pour le flux 2
  - 0% pour le flux 3

## Différentiation de services dans l'entête IP

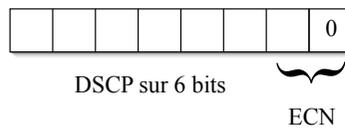
- Permet de garantir des QoS à certains flux
- Diffserv (différenciation de services):
  - Marquage dans l'entête IP
  - Au début
    - » Champs Type de service (ToS):
      - Priorité sur 3 bits
      - 4 bits (délai, débit, fiabilité, coût)
  - AUJOURD'HUI:
    - Champs ToS remplacé par le champ DSCP (Differentiated Service Code Point)
    - 6 bits permettent de différencier des classes de service et de donner des priorités de traitement dans chaque classe

## Champ TOS et DSCP dans IPV4

Champ Type Of Service de IP



Champ Differentiated Services CodePoint



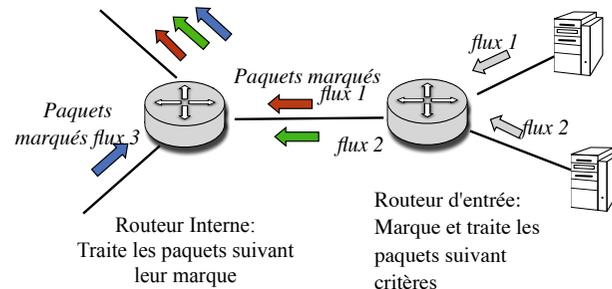
## Champ Class Service et Flow Label dans IPv6



- Champs Traffic Class: Equivalent au champ DSCP d'IPv4
- Flow Label:
  - Permet de marquer l'appartenance d'un paquet à un flux donné
  - Traitement de ce flux particulier par les routeurs
  - Choisi par la source (par exemple pour une connexion TCP donné)
  - Permet aussi de faciliter le travail des routeurs pour le routage

## Différentiation de services en pratique

- A la périphérie du réseau (**domaine Diffserv**), les routeurs sont configurés pour classifier (et marquer) les paquets entrants suivant différentes informations
  - L'adresse IP source ou destination
  - Le port destination
  - ...
- Les routeurs à l'intérieur du réseau sont configurés pour traiter les paquets différemment suivant leur marquage (par exemple par WFQ)



## Exemple de différenciation

- Box d'un fournisseur d'accès
- Accès libre via le WIFI
- Possibilité de donner une priorité suivant l'adresse IP (par exemple) dans le routeur de sortie

