

Unité d'Enseignement L3 : Introduction aux télécommunications

Gilles Menez

Université de Nice – Sophia-Antipolis
Département d'Informatique
email : menez@unice.fr
www : [www : www.i3s.unice.fr/~menez](http://www.i3s.unice.fr/~menez)

20 février 2012: V 1.1

Techniques de commutation

- ⇒ Commutation de circuits
- ⇒ Commutation de messages
- ⇒ Commutation de paquets

Commutation : **switching**

La commutation est l'opération consistant à établir une liaison temporaire physique entre deux noeuds d'un réseau :

➤ Sur cette base, plusieurs mises en oeuvre sont possibles !

- ➡ Commutation de circuits
- ➡ Commutation de messages
- ➡ Commutation de paquets

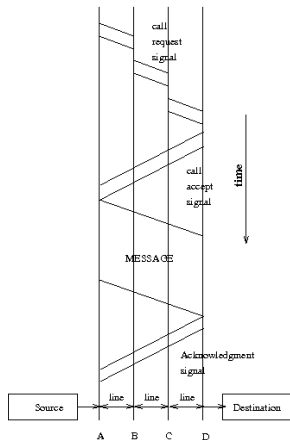
Commutation de circuits : **circuit-switching**

La **commutation de circuits** est la forme la plus ancienne (apparue avec le téléphone) :



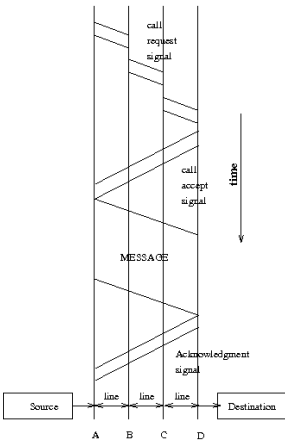
- Elle est aujourd'hui mise en oeuvre par des autocommutateurs (ici le DMS-100 de Nortel) dans les centraux téléphoniques.

Caractérisation de la commutation de circuits



- ① Un **chemin "dédié"** formé par une séquence de liens connectés, entre l'appelant et l'appelé et ceci durant toute la durée de la connection.
- ② Une phase de **connexion**, un signal spécial (dialing tone) transite à travers le réseau et « capture » les liens dans le chemin au fur et à mesure de l'établissement du **circuit**.
- ③ Le circuit forme un canal qui est **affecté en propre** à la communication.

Caractérisation de la commutation de circuits



”Au début” (le téléphone), cette commutation était une commutation spatiale puisque le support physique de transmission était ”matériellement et distinctement” alloué au circuit.

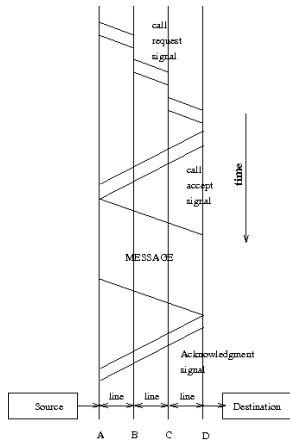
Aujourd’hui **il s’agit plus souvent de considérer que le canal est une ressource dont la capacité est affectée à un taux qui permet d’égaliser une qualité de service équivalente à celle de la commutation spatiale.**

L’affectation « en propre » des ressources

- le fil électrique
- l’intervalle de temps (si il y a multiplexage temporel)
- la bande fréquence (si il y a multiplexage fréquentiel)

doit paraître totale et sans partage en terme de performance.

Caractérisation de la commutation de circuits



- ④ La commutation de circuit ne suppose pas qu'il y ait de pertes d'informations induites par une gestion du réseau.

Une fois le circuit établi, d'un point de vue des utilisateurs finaux (end-users), **un tube est formé !**

D'un point de vue des données, ce qui caractérise ce type de commutation :

- c'est qu'elles ne savent pas « où elles vont » ... elles se laissent guider par le circuit.
 - Elles suivent **toutes** le même chemin !
- ⑤ Après l'établissement du circuit, un signal de retour est émis pour indiquer que la transmission de l'information peut commencer.
- ⑥ Le circuit dure jusqu'au moment où l'un des deux abonnés interrompt la communication.

Commutation de circuits : **circuit-switching**

Inconvénients :

Un des problèmes majeurs de ce type de commutation est la **sous-utilisation** potentiel de la ligne.

- Cette dernière peut en effet restée inutilisée dès lors que les correspondants n'ont plus de données à transmettre.
- Mais qu'ils n'ont pas "rompu" le circuit.

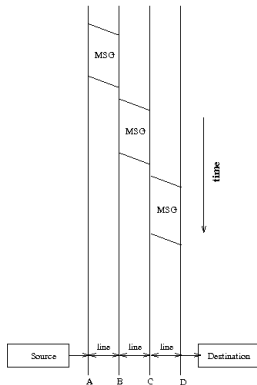
Avantages :

- Ce type de commutation permet de « garantir » la capacité du lien et le débit de transmission.
- C'est une technique assez simple, qui ne nécessite, de la part des commutateur, que de capacités mémoire restreinte.
- De plus, elle peut s'appliquer sur un réseau numérique ou analogique.

Commutation de messages : message-switching

L'idée principale est de pallier au problème de la sous-utilisation de la ligne.

- Elle s'applique aux seuls réseaux numériques.



On crée la **notion de message** :

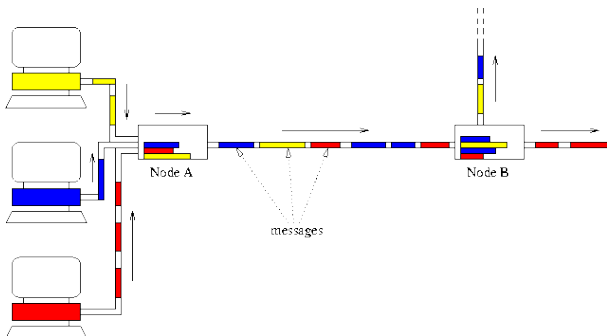
- une suite d'informations formant logiquement un tout pour l'expéditeur et le destinataire (un fichier complet, une ligne tapée sur un terminal ...)

La **commutation de messages** est typiquement utilisée pour la transmission de données

- où l'on n'a pas forcément besoin d'un chemin permanent
- et donc où le flux n'a pas être continu.

Noeud de commutation de messages : Store

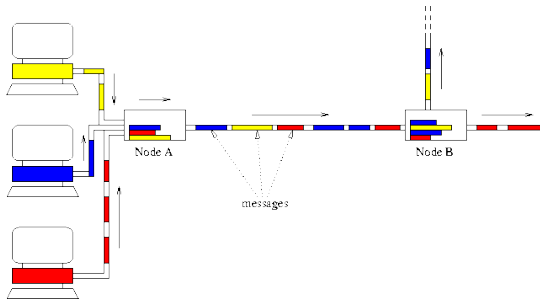
Le réseau à commutation de message se présente sous la forme d'un maillage de **noeuds contenant de la mémoire**.



- On peut y stocker le message à transmettre en attendant la disponibilité des moyens de transmission locaux.

Noeud de commutation de messages : Forward

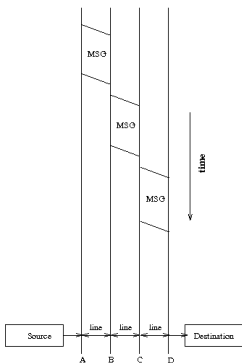
Lorsqu'un équipement veut transmettre un message, il lui ajoute l'**adresse du destinataire** et le transmet au commutateur « le plus proche ».



- Le message est envoyé de noeud en noeud jusqu'au destinataire.
- **Il ne peut pas être envoyé au noeud suivant tant qu'il n'est pas complètement et correctement reçu !**

En résumé : Store and Forward

Dans la commutation par message, **le message est envoyé dans sa totalité** d'un noeud à l'autre avec la possibilité d'être mis en attente (« notion de queuing ») lorsque les liens partant du noeud sont déjà tous occupés :



Ce procédé de progression (de noeud en noeud) est généralement appelée **store-and-forward switching** .

Il inclut :

- ① le « message-switching »
- ② le « packet-switching »

Ce type de commutation convient bien aux **données** dont les lois d'arrivée sont plus proches de la notion de rafales (« bursts ») entre-coupées par des temps morts (« idle »).

- On peut alors transmettre sur un même support de communication physique les paquets des différents canaux.

Commutation de messages : message-switching

Avantages :

Ce type de communication met en évidence le partage des ressources.

- Chaque noeud « s'occupe » tour à tour de messages de communications différentes.
- Ceci impose que chaque commutateur soit capable de stocker le message en entier ce qui peut induire des tailles mémoires importantes.

Inconvénients :

Par rapport à une commutation de circuit, certains paramètres de la transmission ne sont pas forcément garantis :

- ① La disponibilité "suffisante" de la ressource : il peut y avoir des **pertes** si les mémoires sont pleines !
- ② Il n'y a plus un chemin mais **des** chemins.
- ③ Le délai de transmission est, plus sensiblement, fonction du nombre de commutateurs traversés et de la taille (durée) du message : il y a une **gigue**.

Réseaux à message-switching

Ce mode de transmission suppose du système de gestion plusieurs choses qui n'existaient pas dans la commutation de circuits :

- ① Une mémoire dans chaque noeud intermédiaire,
- ② Un **mécanisme d'acquittement** éventuel pour les messages correctement reçus et de demande de retransmission pour les messages erronés.
- ③ De plus, comme la capacité des mémoires intermédiaires est limitée, il va falloir introduire un **contrôle sur le flux** des messages pour être sûr qu'il ne va pas y avoir débordement.
- ④ Il faut avoir des politiques de **routage** des messages pour aider et sécuriser les transmissions : défaillance d'une liaison

Limites de la commutation de messages

La commutation de messages se heurte à la difficulté de transmettre correctement de très longs messages :

- Le Taux d'Erreur Binaire (TEB = BER = Bit Error Rate) exprime un taux d'erreur par bit.

- ➔ Il caractérise la qualité d'une liaison.

- ➔ Si $TEB = 10^{-5}$ alors en moyenne un bit sur 10^5 est erroné.

- Le Taux d'Erreur Paquet (TEP = PER)

$$= 1 - (1 - TEB)^{\#bits \text{ du paquets}}$$

- ➔ Pour un message de 100 000 octets, sa probabilité d'arriver correctement si le BER = 10^{-5} vaut 0.6

- ➔ Pour un fort TEB (TEB = 10^{-3} à 10^{-4}), la probabilité de succès est proche de 0.

Evolution de la commutation de messages

Le réseau **Telex** est un réseau à commutation de messages.



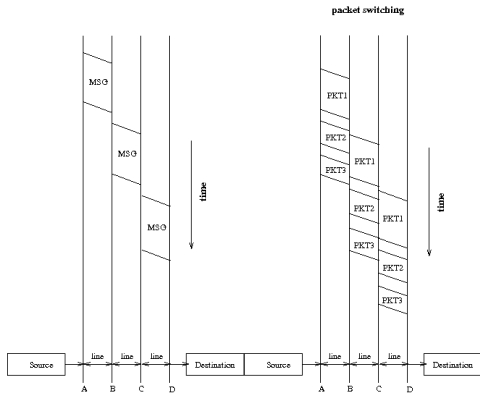
➤ Mais la commutation de messages n'est plus d'actualité !

Du message au paquet !

Pour accélérer la vitesse de transmission et rendre beaucoup plus simples les reprises sur erreur, on a vu apparaître, au début des années 70, **le concept de réseau à commutation de paquets**.

Commutation de paquets : packets-switching

La **commutation de paquets** s'appuie sur le même principe que celui utilisé pour la commutation de messages ...



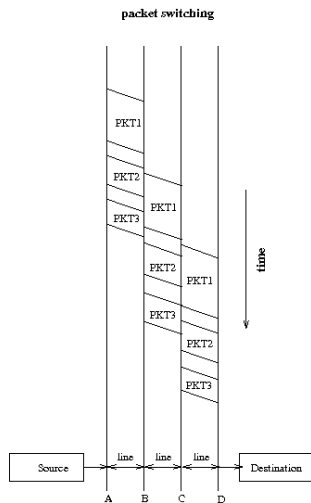
- avec la différence qu'un message est divisé en une série de **paquets** de longueurs finies (max 1000 à 2000 bits) que l'on appelle **segments / datagrams**.

La segmentation

Cette opération de division du message initial s'appelle la **segmentation** .

- Les paquets sont considérés comme totalement indépendants les uns des autres.
- Chaque paquet transite à travers le réseau avec l'ensemble des informations nécessaires à son acheminement et notamment les adresses de l'expéditeur et du destinataire.
- Chaque segment est adressé séparément et numéroté séquentiellement.
- Les paquets composant le message sont alors routés vers leur destination séparément.

La commutation de paquets : **packets-switching**



Plusieurs paquets peuvent être en phase de transmission simultanément (puisque chaque paquet contient toutes les informations nécessaires à son acheminement).

- Cet effet de **pipeline** permet de réduire les temps de transmission.

Avantages :

Par rapport à la commutation de messages, la gestion de blocs d'informations de petite taille est plus simple, **surtout au niveau des reprises sur erreur.**

- L'avantage d'un tel réseau est sa simplicité de réalisation.

Inconvénients :

En revanche, un paquet ne forme pas (sauf si il est court), un tout logique.

Le routage est effectué pour chaque paquet :

- deux paquets successifs échangés entre les mêmes équipements peuvent donc suivre des chemins différents
- et être reçus par le destinataire dans un ordre différent de l'ordre d'émission.

Un problème surgit donc lorsqu'il s'agit de réassembler des paquets pour reformer le message original.

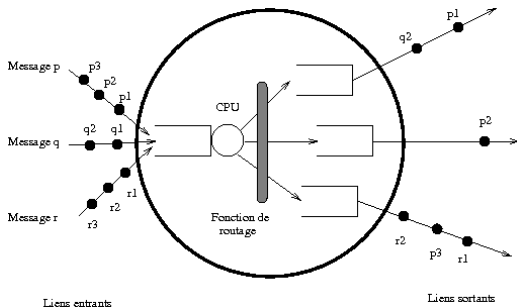
- **Il faut rajouter une couche de protocole supplémentaire.**

Bilan sur les commutations

Commutation	Circuits	Paquets
Circuit dédié	Oui	Non
Bande Passante Disponible	Fixe	Dynamique
Gaspillage potentiel	Oui	Non
Store and Forward	Non	Oui
Chaque paquet suit la même route	Oui	Oui ou Non (selon le réseau)
Etablissement d'un circuit par séquence d'appel	Oui	Oui ou Non (selon le réseau)
Apparition d'un congestion potentielle	A l'établissement du circuit	A chaque paquet
Principe de facturation	A la distance et à la durée	Au volume d'informations transmises

Noeuds de commutation

La commutation de paquets induit l'apparition dans le réseau de **noeuds de commutation** :



Les paquets sont envoyés indépendamment les uns des autres :

- Les paquets de plusieurs flux peuvent donc être **multiplexés temporellement** sur une même liaison.

Noeuds de commutation : Fonction de routage

Comme chaque paquet doit traverser le réseau, il est nécessaire qu'il contienne une **en-tête** comportant des informations de contrôle.

- d'où il vient,
- où il va, ...

Fonction de routage

Ces informations sont utilisées par les commutateurs pour un aiguillage correct et optimal.

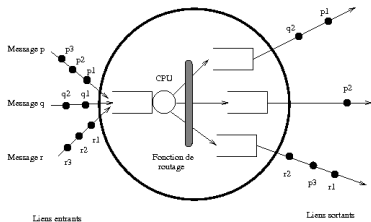
- Chaque paquet reçu par un commutateur est stocké puis **son en-tête est analysé**.

En fonction des informations de contrôle, le paquet est aiguillé vers un autre commutateur (à travers éventuellement un autre réseau) ou le cas échéant vers l'équipement terminal.

Le format de l'en-tête est défini par l'opérateur du réseau.

Noeuds de commutation : Multiplexage

Une liaison entre 2 commutateurs n'est pas monopolisée par un équipement mais supporte la transmission de paquets de multiples communications.



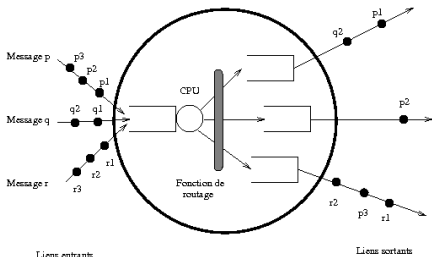
En ce sens on retrouve l'intérêt du multiplexage :

- Si le débit de la liaison est supérieur au flux transmis par l'ensemble des utilisateurs, elle peut supporter de multiples dialogues simultanés tout en donnant l'impression à chacun d'être seul sur le réseau.

Noeuds de commutation : Bufferisation

Si la liaison de données vers le commutateur concerné est occupée, le paquet est conservé en mémoire :

- Chaque commutateur se comporte donc comme une mémoire tampon.



Le dimensionnement de la mémoire des commutateurs est un des éléments déterminant la capacité et les performances d'un réseau à commutation par paquets.

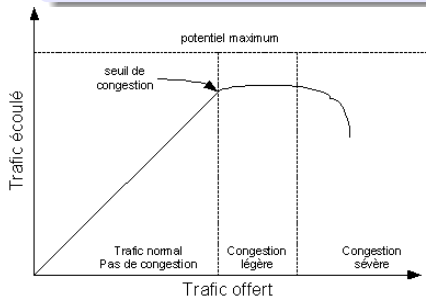
- Si la mémoire d'un commutateur est entièrement utilisée, celui-ci n'est plus en mesure de recevoir de nouveaux paquets. Il peut alors **détruire des paquets** et dégrader les performances du réseau.

Contrôle de congestion

La congestion est statistiquement inévitable sur un réseau.

- Elle intervient lorsque trop de sources tentent d'envoyer trop de données trop vite pour que le réseau soit capable de les transmettre.
- Ceci entraîne la perte de nombreux paquets et de longs délais.

L'ensemble des techniques mises en oeuvre pour éviter la saturation de la mémoire des commutateurs s'appelle le **contrôle de congestion**.



Elle résulte de l'augmentation des délais d'acheminement.

- Si la taille des files d'attente augmente dans les commutateurs, les blocs ne sont pas acheminés dans les délais et sont donc retransmis, **ce qui augmente encore le trafic.**

Contrôle de congestion

Pour prévenir la congestion, il faut mettre en place un contrôle d'admission

- ne pas admettre dans le réseau plus de trafic que celui-ci ne peut en supporter.
- ① Par exemple, en X25, les demandes de connexion seront donc accompagnées par des informations comme le débit moyen et le débit de pointe nécessaire.
 - afin que le réseaux puisse mettre en place un circuit adapté
 - ou puisse refuser cette demande de connexion.

Ceci n'est pas adapté à un trafic en rafale.

- ② Dans le contexte d'Internet, seul TCP (en fait il existe plusieurs versions de ce protocole) dispose de mécanismes permettant d'aborder cette problématique :
 - La source va observer le réseau et son état, via l'occurrence des pertes de paquets, afin d'adapter son débit gérant ainsi la congestion et le flux.

Contrôle de flux

Le contrôle de flux est le processus de gestion du débit de transmission entre deux noeuds :

- Afin d'éviter qu'un émetteur ultra rapide ne sature un récepteur trop lent et donc débordé ...

Le contrôle de flux fournit un mécanisme au récepteur afin qu'il puisse asservir le débit de l'émetteur.

- Le contrôle de flux doit être distingué du contrôle de congestion puisqu'un récepteur peut être saturé sans que le réseau ne soit congestionné.
- TCP propose un mécanismes hybride gérant à la fois la congestion du réseau et le contrôle de flux.

Nature des informations à transporter

L'ère du multimédia est exemple typique de ce que peut produire la rencontre de l'ordinateur, du son et de l'image.

- Texte et hypertexte, voix, image, vidéo et données se côtoient pour offrir toutes les formes possibles de communication.

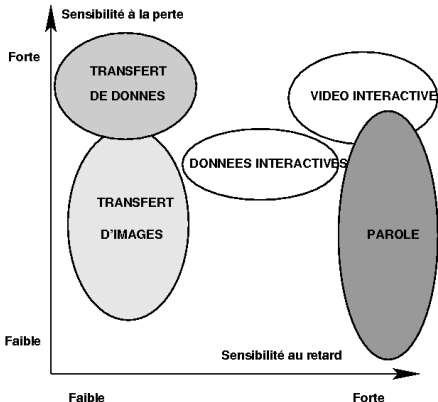
Perception et transport

L'idée de regrouper télévision numérique distribuée, bases de données vidéo, informatique et téléphonie sur un même support de transmission a conduit naturellement au concept d'un réseau « universel »

S'est alors posé la question du mode de transfert qui permettrait d'une part, de véhiculer « idéalement » tous ces types de données, c'est à dire, de façon transparente vis-à-vis de l'information et avec une grande flexibilité au regard des débits possibles (de quelques Kilobits à quelques centaines de Mégabits), mais surtout, de transmettre ces informations à grande vitesse.

Les trafics

En particulier, il est clair qu'une des **difficultés majeurs dans l'intégration des différents services est le respect des contraintes variées liées à chaque type de trafic.**



- ① Données informatiques : trafic asynchrone et sporadique (ou par rafale).
- ② Voix interactive : temps réel (temps de transfert < 150ms pour être indécélable)

La voix interactive nécessite un intervalle strict entre chaque échantillon (transfert isochrone).

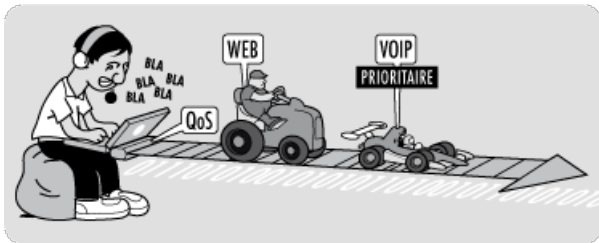
Par contre, un paquet (un petit morceau de signal) perdu n'a aucune incidence sur la conversation et n'est pas détectable par l'oreille. Ces paquets sont soit ignorés, soit reconstitués (en recopiant le précédent).

- ③ Vidéo : même problème que la voix (transfert isochrone) pour l'aspect temps réel.

Mais beaucoup plus d'information à fournir !

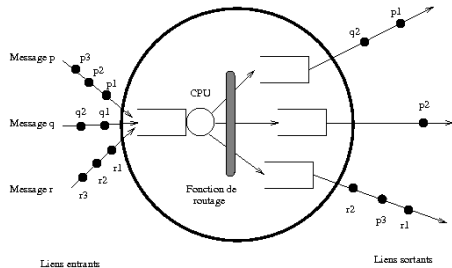
Qualités de services

La **qualité de service (QoS)** est la capacité à véhiculer dans de bonnes conditions un type de trafic donné, en termes de disponibilité, débit, délais de transmission, gigue, taux de perte de paquets ...



<http://www.awt.be/>

Politiques de service



Un routeur basique est composé de processus asynchrones ayant pour fonction :

- récupérer les paquets reçus (file d'attente en entrée),
- contrôler l'intégrité du paquet (calcul du checksum),
- déterminer l'interface de destination,
- transmettre les paquets (files d'attente en sortie).

La **stratégie de gestion des diverses files d'attente** sur un routeur joue un rôle essentiel dans la différenciation des services, à travers le choix de l'algorithme qui place les paquets dans la queue de sortie, et le choix de la taille maximale de la queue.

Plusieurs algorithmes ont été développés !

Politique de service : First In, First Out (FIFO)

C'est la méthode standard de gestion de trafic entre une interface d'entrée et une interface de sortie.

- Les paquets sont **placés dans la file de sortie dans l'ordre dans lequel ils sont reçus.**
- Compte tenu des optimisations logicielles effectuées depuis le début, cette technologie peut-être considérée comme la plus rapide du point de vue de la transmission en paquets par seconde alors que des techniques plus élaborées risquent de dégrader ces performances.

Dans un **environnement réseau avec capacité suffisante**, cette technique est efficace, le délai de mise en file d'attente des paquets étant alors insignifiant par rapport au temps de transmission de bout en bout.

En revanche, **en situation de rafales**, la file d'attente déborde et les paquets suivants sont jetés.

- Des stratégies de mise en **file d'attente différenciée** peuvent limiter la dégradation du service, en permettant à certains trafics d'être traités.

Politique de service : Priority queuing

C'est la forme primitive de différenciation des services.

- Un trafic particulier peut être identifié et réordonné dans la file de sortie suivant un critère fourni par l'utilisateur.
- La granularité de la classification est flexible : par protocoles ou par services au sein d'un protocole.

Mais en contrepartie, cela peut induire une dégradation des performances pour le trafic "normal" !

- Lorsque le trafic classé prioritaire est anormalement élevé, le trafic normal peut-être rejeté par manque de buffers.

Enfin, il est difficile de calculer précisément la gigue induite dans le chemin de bout en bout dans les techniques non-FIFO, le trafic pouvant rester dans la queue pour une période indéterminée.

Politique de service : Class-Based Queuing (CBQ)

Ce mécanisme, utilisé pour éviter qu'une seule classe de trafic ne monopolise les ressources, **définit plusieurs files de sortie avec une priorité et un total de trafic autorisé.**

- Le trafic est extrait de chaque queue suivant une rotation.

Le principe de ce mécanisme est que l'absence de ressources est pire qu'une réduction des ressources.

- Il convient à des liens bas débit, car il induit un surcroît de traitement et une réorganisation des files pouvant dégrader les performances du routeur à haut débit.

Politique de service : Weighted Fair Queing (WFQ)

Cet algorithme donne **un traitement prioritaire aux flux de faible volume et permet aux flux de volume important d'utiliser la place qui reste.**

- Pour cela, il trie et regroupe les paquets par flux, puis met ceux-ci en file d'attente suivant le volume de trafic dans chaque flux.

L'implémentation de la politique de caractérisation du flux est dépendante du constructeur : elle peut utiliser les bits « IP Precedence » dans le champ **TOS** de l'en-tête du paquet pour trier les paquets.

- Dans ce cas, le traitement rapide par le matériel peut ne pas affecter les performances.
- Mais le contrôle du mécanisme de tri, dépendant du constructeur, est figé et peu satisfaisant.

La commutation de paquets

➤ à la mode internet ...

Services transport : "Connectionless"

L'exploitation d'un réseau à commutation de paquets, peut offrir deux types différents de services à un utilisateur (end-user) :

① **service sans connexion** :

Ce type de service est **structuré comme le système postal**.

- Chaque paquet est considéré comme totalement indépendant des précédents par l'équipement expéditeur.
- Il doit donc comporter l'adresse complète du destinataire et éventuellement celle de l'expéditeur.
En peu comme un message ... mais ca reste du paquet !

Il n'y a :

- pas de phase de connexion,
- pas de garantie d'un chemin unique,
- aucune garantie de conservation de l'ordre,
- aucune garantie sur le débit et le délai, ...

Mais on gagne en coût, car il n'y a pas de réservation de ressources.

Services transport : "WithConnection"

② **service avec connexion** :

Dans ce type de service, l'équipement terminal doit d'abord indiquer le correspondant avec lequel il veut dialoguer : le réseau établit un lien **logique** entre les deux équipements et constitue un **tube de dialogue**.

- Cette procédure est logiquement appelée « ouverture de la connexion ».
- Tout paquet délivré par un équipement au réseau est alors transmis jusqu'au destinataire sans qu'il soit besoin de préciser une quelconque adresse.
- Lorsque le dialogue est terminé un des deux équipements peut indiquer au réseau qu'il souhaite fermer la connexion.

Ca ressemble à de la commutation de circuit : on parle de **circuit virtuel**.

- phase de connexion,
- conservation de l'ordre (simulant un circuit),
- débit garanti, ...

Services Fiables

Des services sont **fiables** quand ils ne perdent **jamais** de données.

- Habituellement, la mise en œuvre d'un service fiable se fait par acquittement (accusé de réception).

Les services fiables en mode connexion présentent deux variantes mineures :

- ① par message :
 - Les frontières du message sont préservées entre l'émission et la réception.
- ② par flots d'octets :
 - La connexion est simplement un flot d'octets, sans frontières de messages
 - Le réseau ne respecte pas la structuration en parties de l'émetteur.

Cette variante est celle exploitée par TCP !

Différents services avec différentes qualités

Toutes les applications ne nécessitent pas le mode connexion, ni la fiabilité.

- ① Par exemple la diffusion électronique de prospectus.
 - Ce type de connexion **sans connexion non fiable sans acquittement** est souvent appelé **service datagramme** .
- ② Dans d'autres cas, on ne veut pas établir de connexion pour envoyer un court message, mais on exige une bonne fiabilité.
 - On peut alors utiliser le service **datagramme avec acquittement**.
- ③ Le service **demande-réponse** est encore un autre type de service dans lequel
 - l'émetteur transmet un seul datagramme contenant une demande ;
 - le récepteur transmet alors la réponse.

Il faut enfin remarquer que tous les réseaux ne proposent pas forcément tous les services !

Différents services/exemples

Connexion	Services	Exemples
Avec	Transfert fiable de messages	Suite de pages
Avec	Transfert fiable d'octets	Connexion à un ordinateur
Avec	Connexion non fiable	Voix numérisée
Sans	Datagramme non fiable	Diffusion électronique de prospectus
Sans	Datagramme avec acquittement	Messagerie avec ACK
Sans	Demandes-réponses	Consultation de bases de données

Index :