

# ATM (*Asynchronous Transfer Mode*)

## Introduction

Un seul reseau pr transporter tous les medias (Voix RTC, Données en mode paquet, Television)

RNIS-LB

Meme si c'est tres inspiré du monde des Telecoms, c'est malgres tout une technologie qui ressemble à du **mode paquet** plutot que du circuit

Aucun mecanisme de reprise sur pertes

Note: X.25 est cool, c'est un outil tres puissant pour eviter les congestions puisque, avant de communiquer, tente de mettre en place une connexion (precise meme le debit de la connexion dans la trame de connexion).

Mais ça ne signifie pas que qu'il n'y aura pas de congestion.

Dans les années 80, les reseaux internet avait une faible garantie de QoS (Qualité de Service), aucune garantie sur le temps de traversée du reseau; alors que pour les reseaux telephoniques: gigue quasi nulle, le debit est constant et les temps de traversée faible (par contre: peu de gestion d'erreur)

## Principe de base

- **Communtation** en mode de transfert *asynchrone*\*
- **Fondé sur la notion de cellule** : taile constante, petite taille

48o de data + 5 de header

- **Mode connecté**, plus proche de Frame Relay ou X25
- **Pas de controle d'erreur** (pour plus d'efficacité)
- **Pas de controle de flux** (pour plus d'efficacité)
- **Prevention de congestion** par contrat

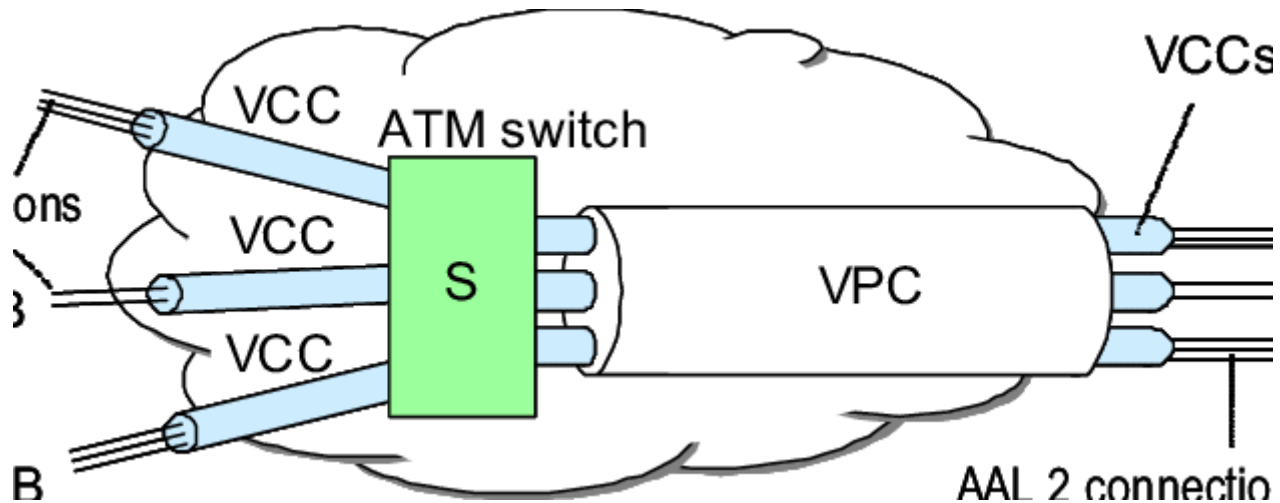
## Le differents niveaux d'ATM

Deux niveaux de granularité:

- **VPC** (*Virtual Path Connection*), ou conduit virtuel

Si on a un conduit virtuel, qu'est-ce qui nous empeche de definir des "sous-conduits"? Rien! D'où la naissancedu concept de *Channel*

- **VCC** (*Virtual Channel Connection*), ou voie virtuelle



Deux niveaux d'aiguillages:

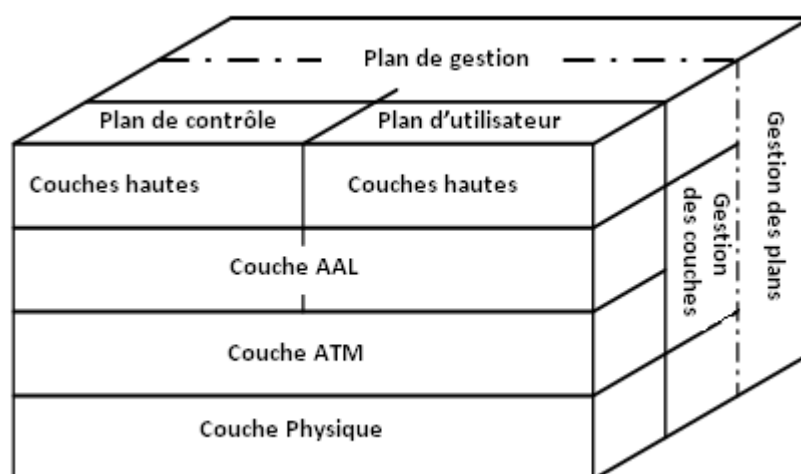
- **Commutateurs** pour les VCC: lit l'adresse au niveau **VCI** (*Virtual Channel Identifier*) et fait la commutation des Channels
- **Brasseurs** pour les VPC: ne regarde que le **VPI** (*Virtual Path Identifier*) de la cellule et fait la commutation de pls Channels en meme temps.

Attantion: une connexion virtuelle est **unidirectionnelle**!

Note: il y a une doc qui dit que la valeur VPI/VCI est associée à une connexion, et que donc une meme valeur est attribuée dans les 2 sens de transmission, ce qui impliquerait que le chemin est le meme dans les 2 sens!

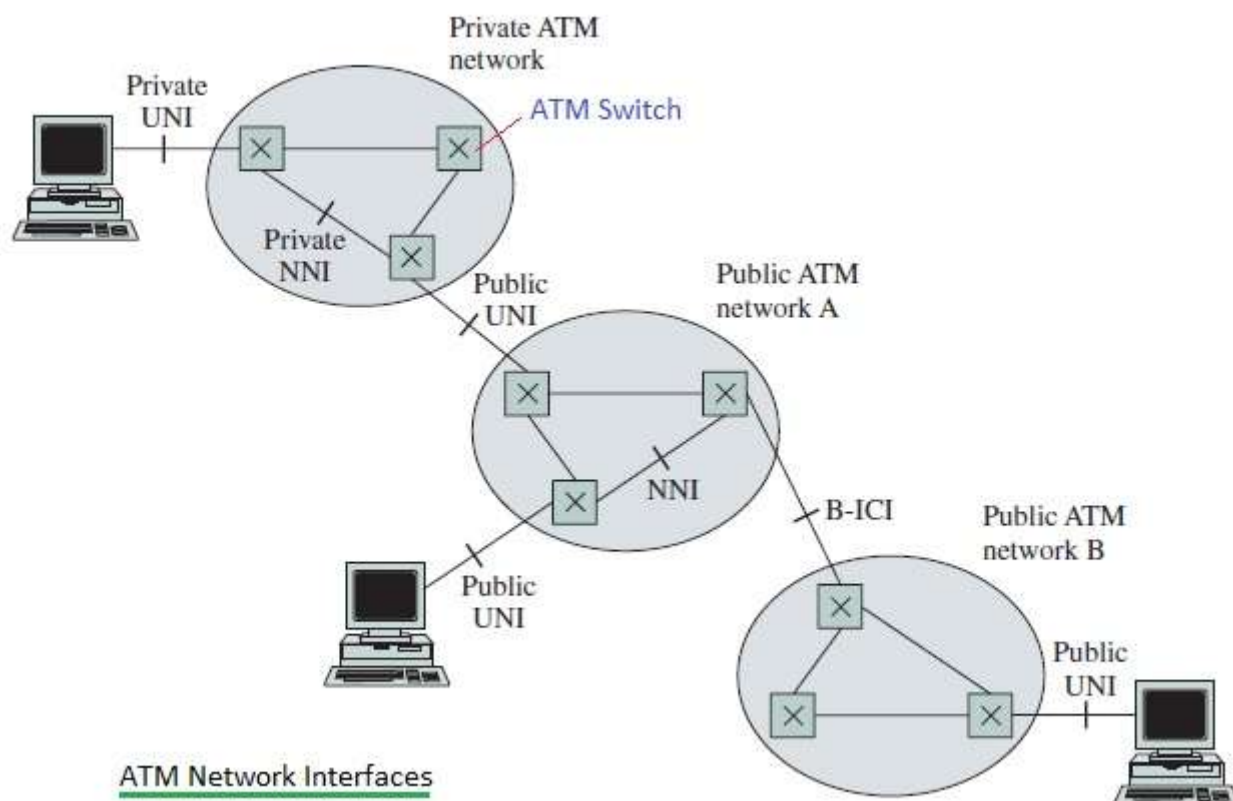
## Contrat

Un contrat de trafic est etabli entre l'usager et le reseau et est bilatéral. Les parametres de trafic et de QoS sont négociés par l'interface usager/reseau (*UNI - User Network interface*)



## UNI - l'interface Usager/Reseau

Des connexions *point a point*.



Les cellules ATM sont envoyées sur le réseau. Au besoin, la couche physique peut également envoyer des données sur le réseau (aussi sous forme de cellule). S'il n'y a pas de data à envoyer, on envoie des cellules vides.

#### Rapelle SDH/PDH

- Pour la synchro bit: c'est à la couche physique de se démener.
- Pour la synchro cellule: **Le 5e octet de l'entête d'une cellule est la somme des 4 premiers octets.**  
Donc si on reçoit périodiquement un octet, qui est la somme des 4 précédents, on se doute que nous sommes synchro niveau cellule

## Couche Physique

**Detection d'Erreur:** L'ATM est capable de détecter plusieurs erreurs et d'en corriger une.

#### HEC (Head Error Control)

Les cellules peuvent être encapsulées dans des trames de la couche physique

PDH, SDH & DSL peuvent encapsuler ATM

## Couche ATM

Un lien est un lien composite de cellules qui appartiennent à différents VP/VC, on peut donc faire du **multiplexage**.

L'entité ATM procure plusieurs primitives de service comme **ATM-DATA.demande** ou **ATM-DATA.indication**.

On rappelle que la taille des cellules **est fixe de 48 octets** (Les américains voulaient 64, les européens 32, du coup la question a été tranchée et le resultat ne satisfait personne 😊 )

- faible taille à un **bon impact sur les delais**
- **multiplexage plus efficace**
- **mais surcharge** (overhead) est **plus important** également

## Structure des cellules

### Cellules ATM UNI

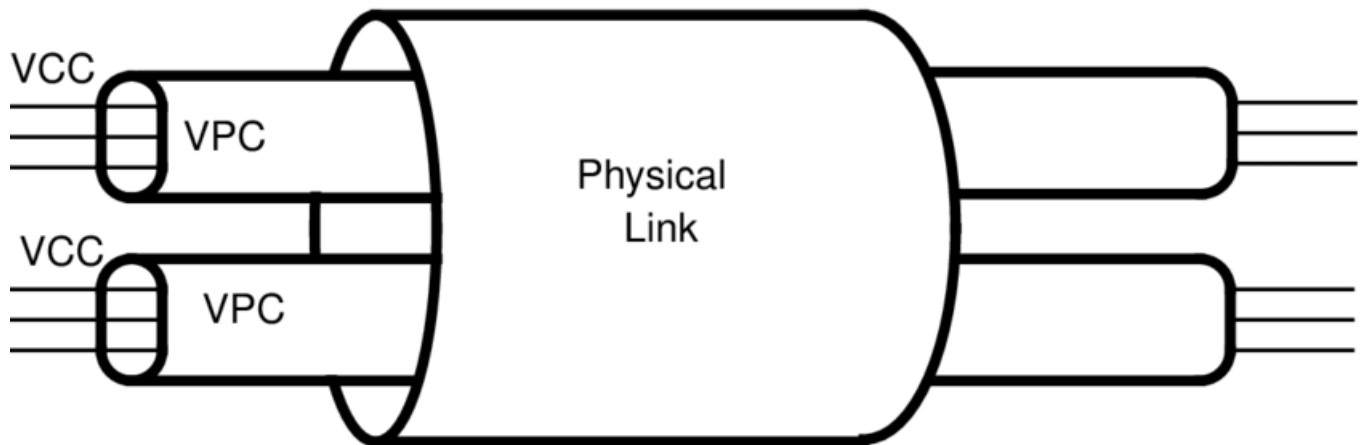
GFC | VPI | VCI | PT | CLP | HEC | Data..

### Cellules ATM NNI

VPI | VCI | PT | CLP | HEC | Data..

Champ	UNI	NNI
GFC	4 bits	0
VPI	8 bits	12
VCI	16 bits	16
PT	3 bits	3
CLP	1 bit	1
HEC	8 bits	8
<b>Total</b>	<b>40 bits</b>	<b>40 bits</b>

On remarque que le nombre de VP est plus petit dans UNI que NNI



## Debit

Les debits seront en nb de cellules/secondes. Pour ATM on facture au volume (Xeuros/minutes) et non au service (Xeuros/mois), donc il faut donc garantir une certaine QoS

Il faudra savoir dir "non" à un utilisateur si les ressources ne sont pas suffisantes pour garantir la communication

Mesure 2 debits:

- **Debit moyen**, debit à long terme
- **Debit crete**, si on depasse le debit moyen de temps en temps, une dextieme limite est definie pour limiter le debit (dans le temps & qte de debit). Aussi appelé **PCR** (*Peak Cell Rate*)

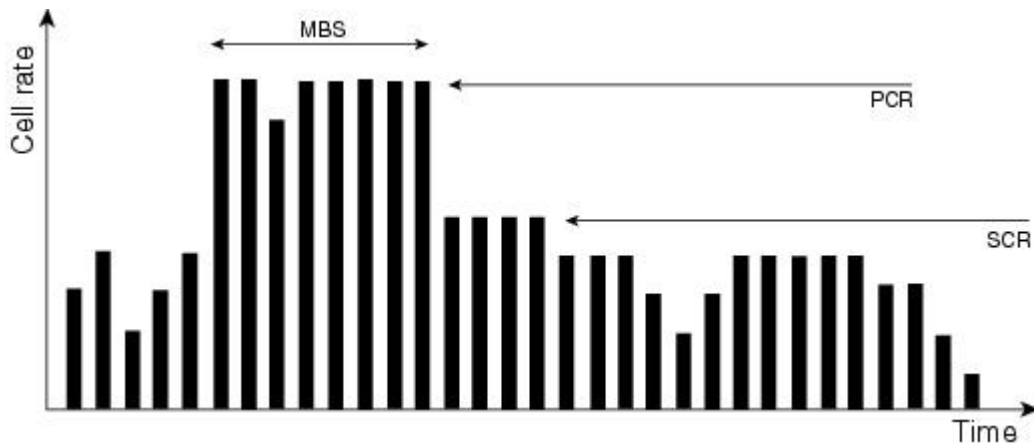
Il y a une certaine variation de debit qui est tolérée, pendant un laps de temps toleré

- **CDVT** (*Cell Delay Variation Tolerance*) = Gigue acceptale
- **MBS** (*Maximum Burst Size*) = le temps maximum de la gigue (donné en nombre de cellules)

## Parametres de QoS

Parametres negociés lors de la mise en place du reseau

- **CLR** (*Cell Loss Rate*)
- **CTD** (*Cell Transfer Delay*)
- **CDVT** (*Cell Delay Variation Tolerance*)



Le but est que ça ressemble autant que possible à du circuit

## Trucs intéressants à relever

Il y a la notion de **BECN/FECN** (*Backward/Forward Explicit Congestion Notification*)

Il est d'usage que, lorsqu'une cellule est perdue, tout un ensemble de cellules deviennent inutilisables. On donne donc des garanties à l'échelle d'une trame (contenu d'un ens. de cellules) et non à l'échelle d'une cellule.

Pour être sûr que le user respecte sa part du contrat, un **UPC** (*User Policing Contract*) est placé sur l'interface Utilisateur/Network (UNI).

Utilisation de l'algorithme GCRA(T,t) qui va nous donner la confirmité de chaque cellule

On fait tourner le GCRA sur le débit moyen et le débit crete en parallèle. Et si le commutateur commence à être submergé, on détruit des cellules (celles qui sont marquées "Non conformes" en priorité)

### Exemple application de GCRA

- $T = 10, t = 15$
- Tcells: 0, 12, 23, 30, 36, 42, 47, 53

Cel	X	t	Statut
0	0	0	OK
2	10	12	OK
3	22	23	OK
4	33	30	OK
5	43	36	OK
6	53	42	OK
7	63	47	X
8	63	53	OK

## Suite des notes pertinentes

Utilisation du bit AUU mis a 1 pour reperer une trame a detruire dans son integralité

Si on detruit une cellule qui transporte un bout d'une trame, il est inutile de transmettre les autres cellules de la trame puisque, une fois arrivé au destinataire cette trame sera incomplete, et sera dtruite.

On fait du controle de congestion preventif: si une connexion risque de provoquer de la congestion, on ne se risque pas a mettre en place la connexion.

- la signalisation permet d'etablir une connexion
- une fois la co etablie: on reserve les ressources pour la communication

Cette partie est gerée par le **CAC** (*Call Admission Control*) qui est chargé d'accepter ou refuser l'etablissement de connexions.

Note: le CAC n'est pas formalisé...du coup ca fait partie des plus-values des fabricants

## Couche AAL

---

Il existe differents types de services en fonction de quelques parametres.

Theoriquement, les AALs se decomposent en sous-couches

### AAL-1

**AAL-1**: adaptée au trafic CBR (utiliser pour du SDH)

### AAL-2

**AAL-2**: adaptée a la voix compressée (GSM), à la difference des autres AAL, les message sont tellement courts qu'on essaye de mettre pls msg dans la meme cellule (de diverses communications).

- Il y a une autre sous-couche CPS
- Encore plus de signalisation (definition d'un nouveau protocole)

### AAL-3/4

**AAL-3/4**: fusion de l'AAL-3 et de l'AAL-4

- Sous-couche SAR pour la protection des segments
- Sous-couche CPCS