

## Chapitre 6 : Le niveau Réseau

/udd/bcousin/Pages-web-Armor/Enseignement/Reseaux-generalites/Cours/6.recover.fm - 5 juin 2001 11:19

### Plan

- 1. Introduction p167
- 2. Segmentation-réassemblage p169
- 3. Adressage p171
- 4. Le routage p179
- 5. Le contrôle de congestion p195
- 6. Etude du protocole PLP p198
- 7. Conclusion p215

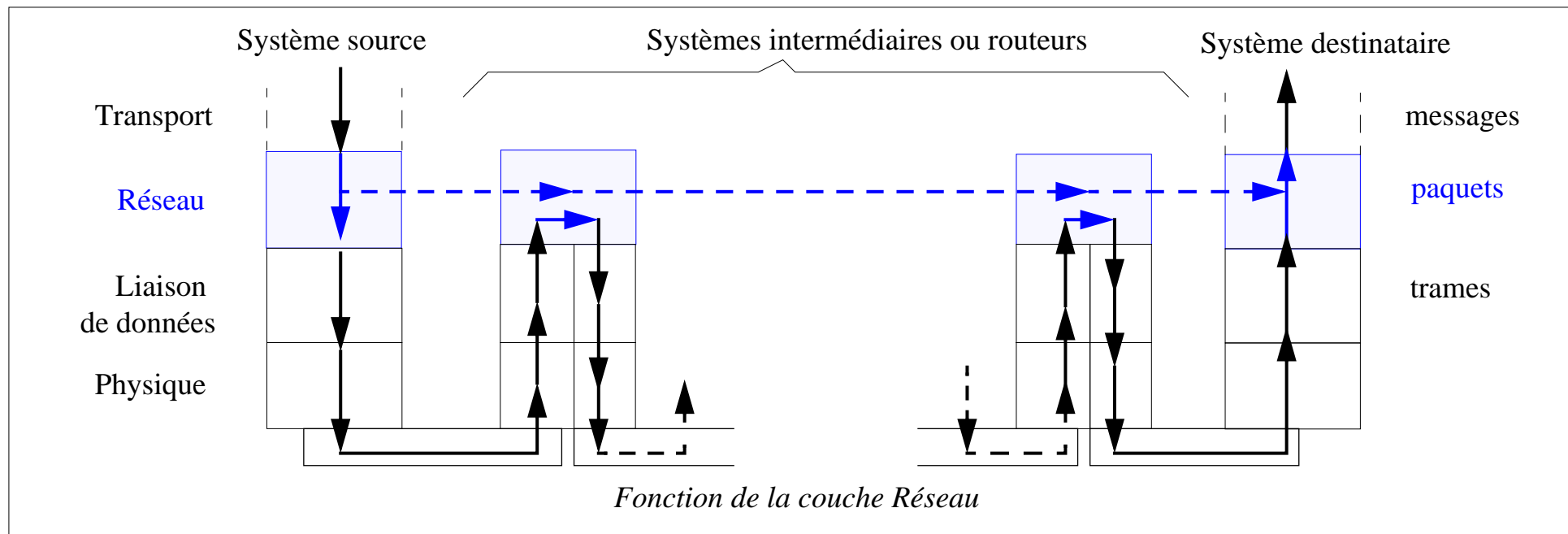
### Bibliographie

- Service et protocole d'accès au réseau, ISO 8348 et 8208 (CCITT X25 et X213)
- G. Pujolle, Les réseaux, Eyrolles, 1995. Chapitre 6.
- H. Nussbaumer, Téléinformatique, Presses polytechniques romandes, 1987. Tome 2, chapitre 2.
- A. Tanenbaum, Réseaux, InterEditions, 1997. Chapitre 5.
- U. Black, X.25 and related protocols, IEEE computer society press, 1991.

## 1. Introduction

### 1.1. Fonctions de la couche Réseau

Cette couche est chargée de transmettre les données de la couche supérieure (Transport) en utilisant un système de communication (un réseau) composé de systèmes intermédiaires (généralement homogènes) interconnectés par des liaisons de données.

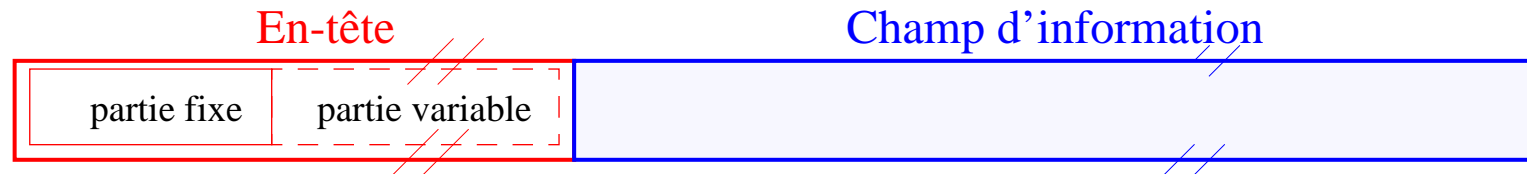


Remarque : le terme réseau est ambigu puisqu'on l'utilise à la fois pour nommer la couche et le système de communication. En plus, le réseau peut être constitué de plusieurs (sous-)réseaux.

## 1.2. Le paquet

C'est l'unité de données d'un protocole de niveau Réseau (**N-PDU** : "Network PDU")

- généralement de longueur variable (mais bornée)
- un **entête**
  - . composé d'une partie fixe (champs fixes en nombre et en taille)
  - . et parfois d'une partie optionnelle (champs optionnels)
- un **champ d'information** de longueur variable mais borné.
- Exemple : X25.3 ou IP



*Format général d'un paquet*

### Remarques :

- Suivant le rôle qui est attribué à un paquet, son champ d'information peut être vide.

## 2. Segmentation-réassemblage

Adaptation de la taille des unités de données à transporter à celle des unités de données transporteuses.

Les messages émis par les applications peuvent être de dimension très variable :

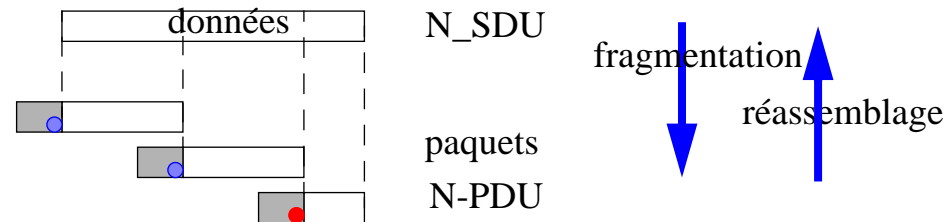
- commandes de quelques caractères
- fichiers de plusieurs Giga octets

Les paquets acheminés dans le réseau doivent avoir une taille adéquate :

- ni trop longue :
  - . difficulté de la gestion du stockage,
  - . inefficacité de la protection contre les erreurs,
  - . augmentation du temps de commutation
- ni trop courte :
  - . le surcoût de l'entête devient prépondérant

Le protocole de la couche Réseau met généralement en oeuvre un mécanisme de segmentation :

- identification des N-PDU appartenant aux différents N-SDU
- ordonnancement des N-PDU appartenant au même N-SDU
- délimitation des N-SDU entre eux (marquage du dernier N-PDU de chaque N-SDU)



Remarques :

- au sein d'une connexion, les deux premiers points sont naturellement rendus!
- la fragmentation peut être très coûteuse : manipulation de la totalité des données à chaque routeur

### 3. Adressage

#### 3.1. Introduction

La couche Réseau utilise des adresses pour **identifier** les systèmes (d'extrémité et intermédiaires).

Deux types de structuration des adresses existent :

L'adressage **global** (linéaire, plat) :

- chaque station se voit attribuer un numéro indépendamment de toute sémantique,
- des stations voisines peuvent avoir des adresses sans aucune relation entre elles.

L'adressage **hiérarchique** (structuré) :

- chaque adresse est décomposée en champs, sous-champs, etc.
- la sémantique associée à chaque champ peut être géographique ou fonctionnelle

Quelques critères pour le choix de la structure d'adresse :

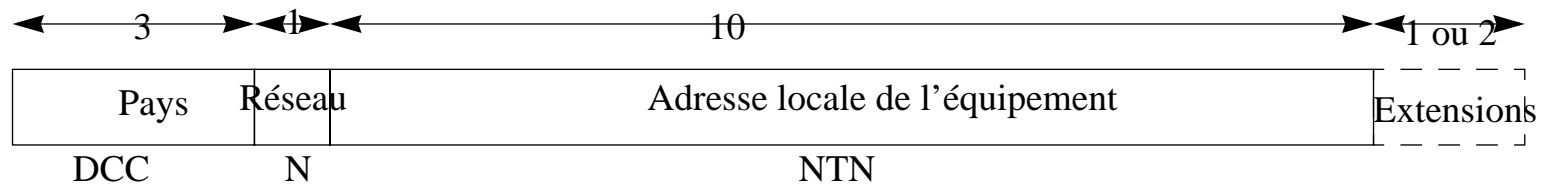
- mobilité : + adressage global
- routage : + adressage hiérarchique géographique
- diffusion (identification d'un groupe de stations) : + adressage global
- densité : + adressage global

## 3.2. Quelques adressages de l'ISO

### 3.2.1 L'adressage X.121 : réseau public de transmission de données numériques

Adressage hiérarchique

14 demi-octets (utilisant une représentation DCB)



Préfixes réservés : 0=E.163 (téléphone), 1=réservé, 8=F.69(télex), 9=E.164(RNIS)

Le champ DCC (“data country code”)

- Exemple : US = [310-329], France = [208-212]

Le champ N (“network code”) : 10 réseaux au plus par DCC!

- Exemple : Transpac = (208)0

Le champ NTN (“network terminal number”) : 10 chiffres (optionnellement 12).

- La structuration interne de ce champ d'adresse est choisie par l'administrateur local.

Exemple : (en DCB) 20 80 11 42 07 19 68

### 3.2.2 L'adressage E.164

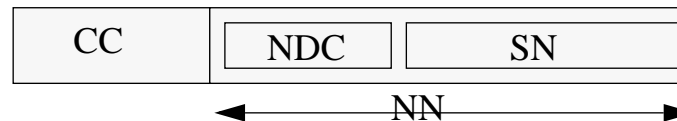
Plan d'adressage pour le RNIS (similaire au E.163 : plan du téléphone)

CC : “country code” (variable)

NN : “national number” (variable), consistant en 2 autres codes:

- NDC : “national destination code” (variable)
- SN : “subscriber number” (variable)

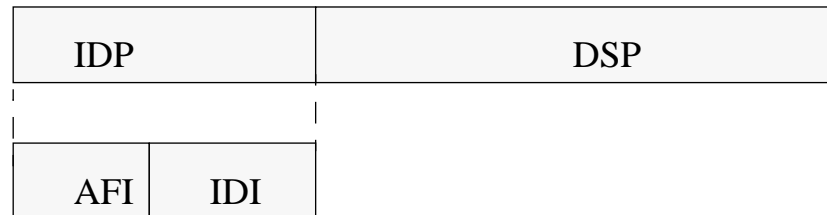
. La structuration interne de ce champ d'adresse est choisie par l'administrateur local.





### 3.2.3 L'adressage de la norme ISO 8348/2 ou CCITT X.213/A

Plan d'adressage hiérarchique :



- Initial domain part :
  - . Authority & format identifier (2 chiffres) : format de l'IDI et syntaxe du DSP,
  - . Initial domain identifier : identifie le domaine d'adressage donc l'autorité chargée d'interpréter le champ suivant.
- Domain specific part

Quelques codes du champ AFI :

- Adresse X.121 d'ETTD codée en décimal locale (52) ou globale (36), ou binaire locale (53) ou globale (37).
- Adresse X.121 d'ETCD codée en décimal (38) ou binaire (39).
- Adresse Télex codée en décimal locale (54) ou globale (40), ou binaire locale (55) ou globale (41)
- Adresse téléphonique codée en décimal locale (56) ou globale (42), ou binaire locale (57) ou globale (43)
- Adresse RNIS codée en décimal locale (58) ou globale (44), ou binaire locale (59) ou globale (45)

### 3.3. L'adressage d'Internet

Les adresses du protocole IPv4 : longueur fixe de 32 bits.

Adressage hiérarchique à 2/3 champs :

- (classe,) netid, hostid

3 classes d'adresse unicast :

- longueur variable des champs netid et hostid.

	31	24	23	16	15	8	7	0
classe A	0	netid		hostid				
classe B	10	netid			hostid			
classe C	110	netid				hostid		
classe D	1110	multicast						
		reserved						

Exemple : 146.84.32.07

classe B, netid = 146.84, hostid = 32.07

1 classe d'adresse multicast :

- identifie un groupe de stations

Nouvelle version IPv6 :

- longueur des adresses 128 bits!
- adresses "anycast" (+ unicast + multicast)

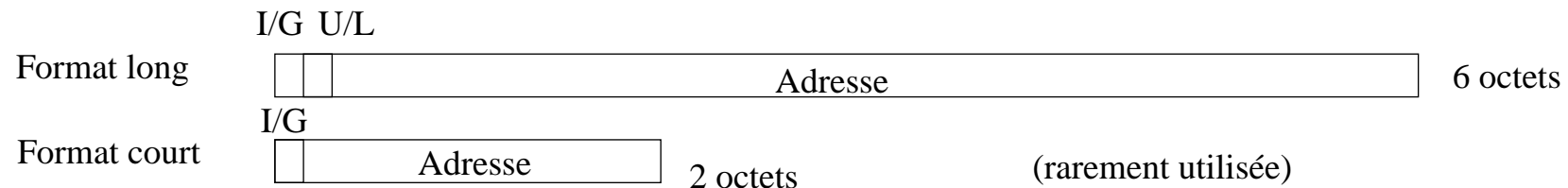
### 3.4. L'adressage IEEE 802

Adresse utilisée par les **réseaux locaux** (couche Liaison de données!) : norme IEEE 802

- Ethernet, Token Ring, Token Bus, FDDI.

Adressage global!

2 formats :



- bit I/G : adresse individuelle ou de groupe
- bit U/L : adresse universelle ou locale

Adresses réservées :

- exemple : Broadcast = 0xFFFFFFFFFFFF

Attribution des adresses (par les constructeurs des cartes) :

- 3 premiers octets : numéro de l'équipementier (exemples : Sun= 080020, 3COM= 02608C)
- 3 derniers octets : numéro de série de la carte

### 3.5. La gestion du plan d'adressage

Administration de la **structuration variable** des adresses (de **longueur variable**) :

- attribution des adresses (unicité)
- décodage des adresses (efficacité)

Chaque domaine possède une **autorité**. Chaque autorité connaît :

- les adresses des objets de son domaine
- l'autorité du domaine supérieure dont elle dépend
- les autorités des domaines inférieures (leur identification)
- les ensembles d'adresses qui dépendent de ses sous-autorités

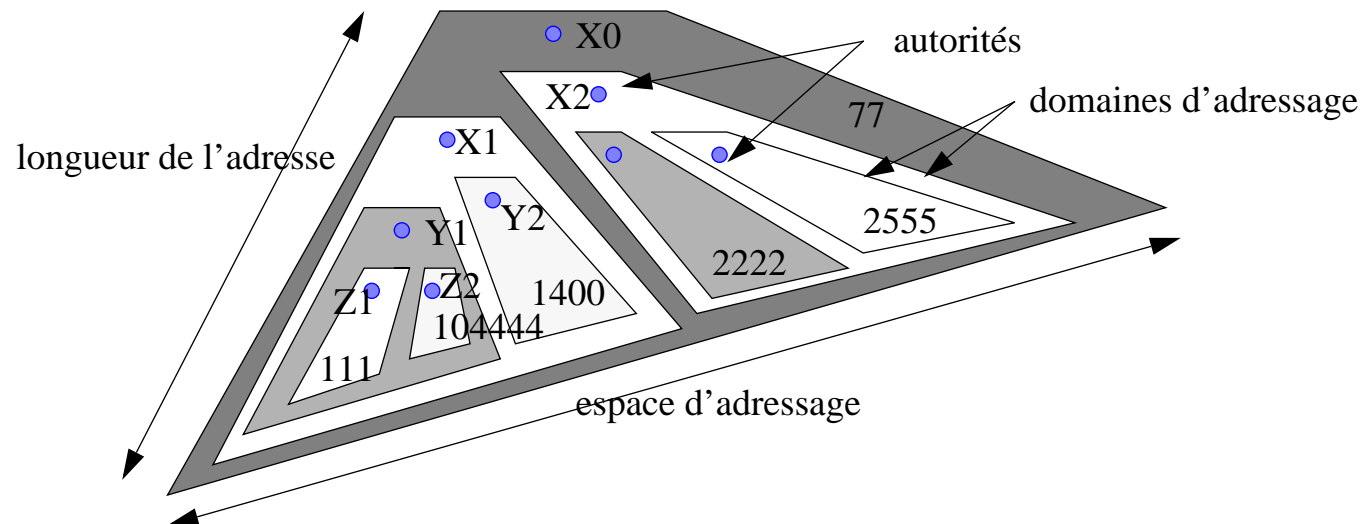
Généralement le décodage est basé sur les **préfixes** qui identifient de manière unique :

- les sous-domaines
- et les objets du domaine

Chaque autorité délègue la gestion des suffixes aux autorités de ses sous-domaines.

### Exemple :

- $x \in \{1.*\} \rightarrow$  autorité X1
  - .  $x \in \{1.[0-2].*\} \rightarrow$  autorité Y1
    - $x \in \{1.0.*\} \rightarrow$  autorité Z2
    - $x \in \{1.[1-2].*\} \rightarrow$  autorité Z1
  - .  $x \in \{1.[3-9].*\} \rightarrow$  autorité Y2
- $x \in \{2.*\} \rightarrow$  autorité X2
- $x \in \{[3.*- 9.*]\} \rightarrow$  objets du domaine de X0



### Autre exemple :

- la numérotation téléphonique!
  - . utilisation des préfixes, longueur variable des numéros, etc.

## 4. Le routage

### 4.1. Introduction

La fonctionnalité principale de la couche Réseau consiste à transmettre des paquets de données issu d'un émetteur à destination d'un (ou plusieurs) récepteurs, paquets qui doivent traverser un réseau composé de nombreux noeuds intermédiaires (routeurs).

A chaque réception d'un paquet, un routeur doit choisir vers quel prochain routeur il doit retransmettre le paquet entrant pour que celui-ci arrive à destination.

- En mode **datagramme**, le choix est effectué indépendamment pour chaque paquet.
- En mode **circuit virtuel**, le choix est fixé à l'établissement de la connexion et pour toute la durée de la connexion.

Dans chaque routeur (commutateur), ce choix est effectué en se servant d'informations contenues dans une **table de routage (commutation)**.

Les entrées d'une table de routage sont renseignées soit manuellement, soit automatiquement à l'aide d'**algorithme de mise à jour des tables de routage** en se basant sur différents critères (débit possible, disponibilité de la ligne, taux d'erreurs, nombre de noeuds intermédiaires, ...)

Fonctions d'un routeur :

- **acheminement** des paquets ("datagram forwarding"), c-à-d transmission des paquets.
- **mise à jour** des tables de routage - algorithme de routage.

## 4.2. L'acheminement par datagramme

Chaque routeur est muni d'une table de routage. Cette table reflète l'état (perçu par le routeur) de la topologie du réseau à un moment donné.

Actions effectuées lors de la réception d'un paquet:

- extraction de l'adresse de destination,
- recherche dans la table de routage,
- retransmission du paquet vers le prochain routeur

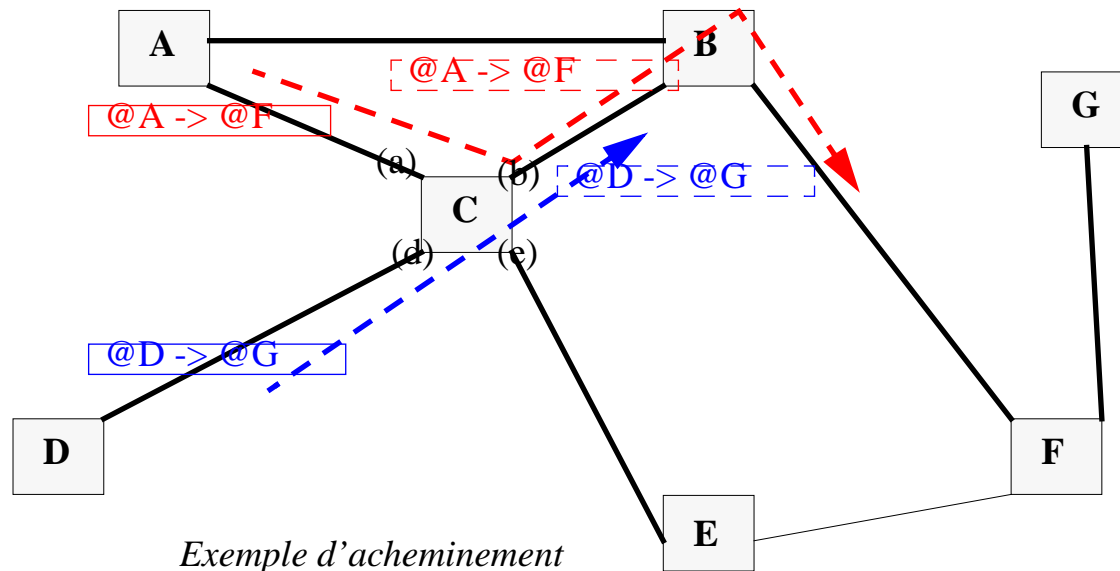


Table de routage de C

Destination	Prochain routeur	Informations complémentaires
@A	@A	(a)
@G	@B	(b)
...		
@D	@B	(b)
@F	@E	(e)
...		

### 4.3. Le procédé d'acheminement par circuit virtuel

Le circuit virtuel (CV) relie l'émetteur au destinataire. Chaque tronçon de CV est identifié par un n° de voie logique (NVL).

Les numéros de VL sont réservés lors de l'établissement de la connexion. Ils sont rendus lors de la libération de la connexion.

Actions effectuées lors de la réception d'un paquet :

- extraction du NVL, recherche dans la table de circuit virtuel
- échange du NVL, retransmission du paquet

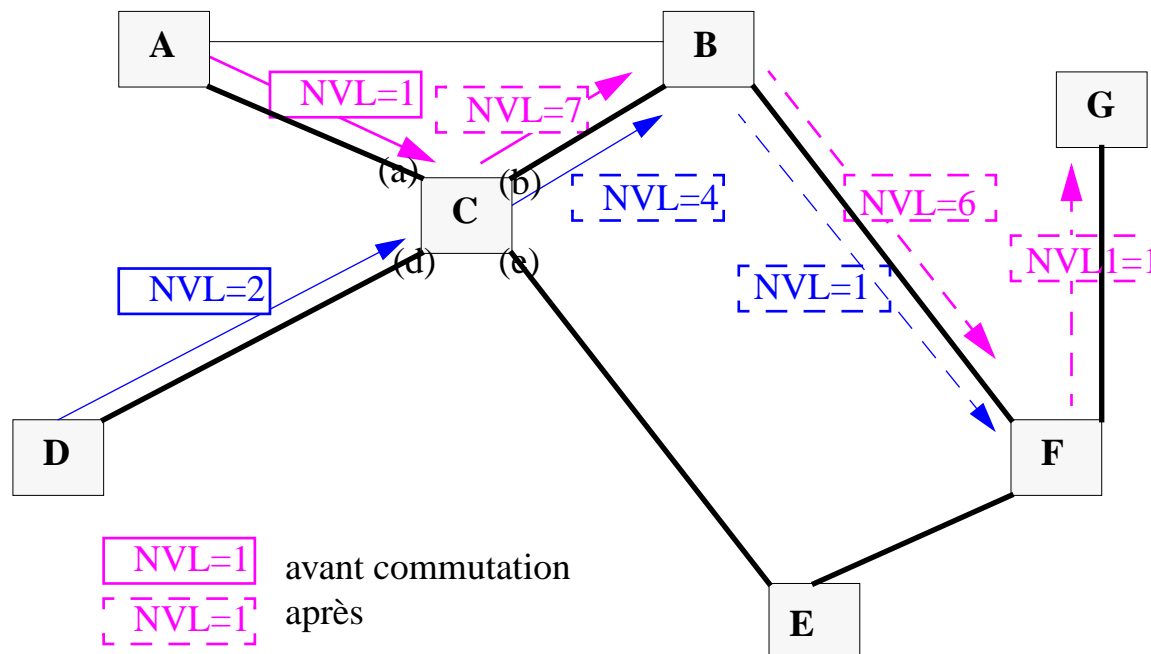


Table de commutation de C

Entrée		Sortie	
Origine	n°VL	Prochain	n°VL
(a)	1	(b)	7
(b)	1	(a)	1
(b)	2	(d)	1
(d)	1	(b)	2
(d)	2	(b)	4
(e)	4	(d)	2



#### 4.4. Services et procédés

On confond souvent services et procédés.

Deux **services** d'acheminement :

- en mode connecté ou en mode non connecté

Deux **procédés** d'acheminement :

- Par circuit virtuel ou par datagramme

##### 4.4.1 Le service **non connecté**

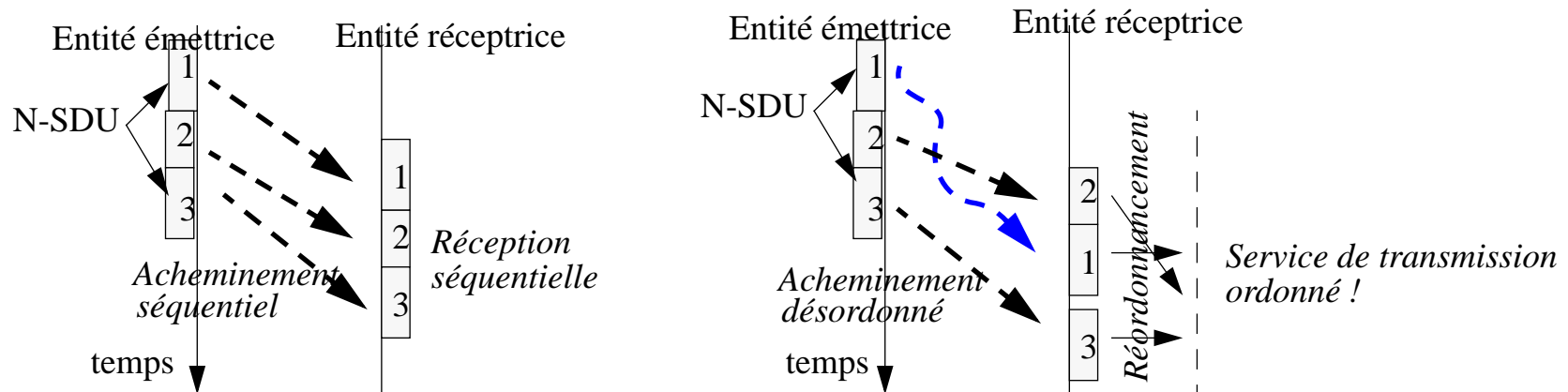
- Les paquets (N-SDU) sont reçus et délivrés par les entités Réseau indépendamment les uns des autres.
  - Utilise généralement le procédé d'acheminement **par datagramme**,
  - Généralement pas de service d'augmentation de la fiabilité.
    - . Les services de protection contre les erreurs, de contrôle de flux et le ré-ordonnement des paquets sont éventuellement reportés dans les couches supérieures.

##### 4.4.2 Le service **connecté**

- Les entités d'une même connexion partagent un même contexte :

- par exemple au sein d'une connexion une entité connaît le numéro du prochain paquet envoyé par l'autre entité.
- Les paquets (N-SDU) appartenant à la même connexion sont délivrés dans l'ordre où ils ont été émis. Et généralement, un contrôle de flux et d'erreur y est associé.
- Utilise généralement le procédé d'acheminement **par circuit virtuel**.

Le mode d'acheminement par datagramme et service ordonné ne sont pas contradictoires :

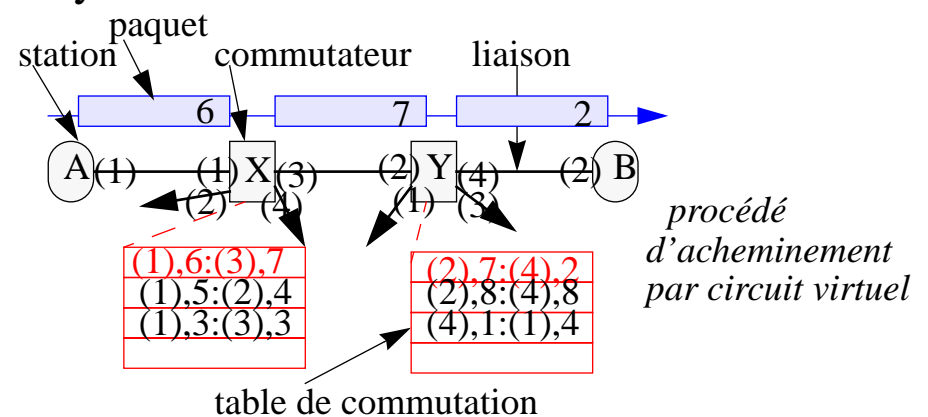
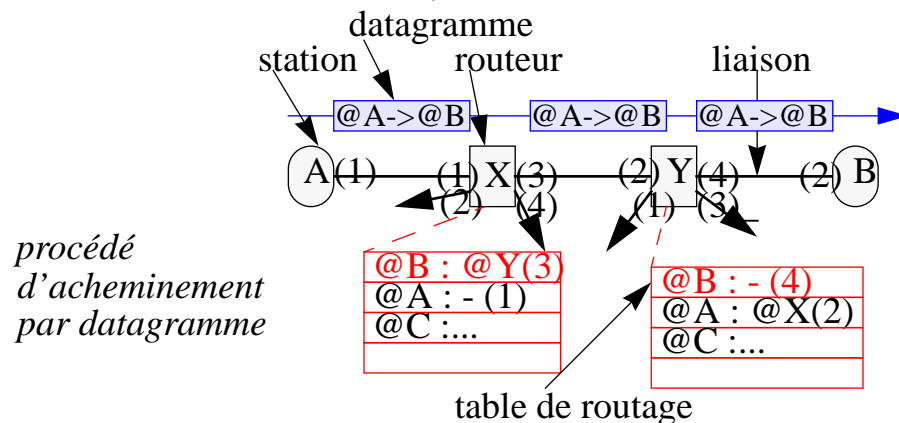


#### 4.4.3 Le procédé d'acheminement **par datagramme** :

- Les paquets (N-PDU) sont transmis **indépendamment** les uns des autres au sein du réseau.
  - Leur routage est effectué grâce à l'**adresse de destination** figurant dans l'entête du paquet,
  - Le routage des paquets (entre deux mêmes stations) est adaptatif,
  - Généralement, les routeurs effectuent peu ou pas de traitement de contrôle.

#### 4.4.4 Le procédé d'acheminement **par circuit virtuel** :

- Les paquets (N-PDU) sont souvent acheminés **séquentiellement** grâce à un circuit virtuel ouvert au sein du réseau.
  - Leur commutation est effectuée grâce à un **identificateur de circuit virtuel** figurant dans l'entête du paquet (c'est le numéro de voie logique : N°VL),
  - La phase d'établissement du circuit virtuel réserve les ressources (notamment le numéro de voie logique) au sein de chaque commutateur,
  - Tous les paquets utilisant un circuit virtuel suivent le même chemin,
  - Les paquets sont transmis sur ce circuit virtuel dans l'ordre,
  - Généralement, un contrôle de flux et d'erreur y est associé.



- Note : une table de routage est nécessaire lors de l'établissement de la connexion pour réserver le numéro de circuit virtuel!

## 4.5. Comparaisons

### 4.5.1 Circuit virtuel et datagramme, adresse et identificateur de CV

- Un numéro de voie logique identifie un tronçon de CV (pour une seule liaison entre 2 systèmes intermédiaires). Le chemin suivi par un CV est identifié par une suite de n° de VL.
- Le numéro de voie logique est généralement de plus faible taille qu'une adresse
- La table de commutation est plus petite que la table de routage.
- La durée de vie de l'association entre numéro de VL et circuit virtuel est celle de la connexion.
- La réservation d'un n° de VL est effectuée localement pour chacune des liaisons formant le circuit virtuel. Sinon :
  - . des réservations multiples de n° risqueraient d'apparaître,
  - . la résolution des conflits de numérotation serait complexe et longue.
- Grâce au circuit virtuel :
  - . la commutation est plus rapide,
  - . le contrôle du trafic est aisé,
  - . la taxation est plus facile.

## 4.5.2 Comparaison des deux types de service de transmission Réseau

**Tableau 1 : service en mode connecté ou non**

Fonctionnalités	Service en mode non connecté	Service en mode connecté
Etablissement/Initialisation	Inutile	Nécessaire
Adresse du destinataire	Dans chaque paquet	Dans le paquet d'initialisation
Séquencement des paquets	Non garanti, généralement	Garanti, généralement
Contrôle d'erreur/Contrôle de flux	Non fournis, effectués (si besoin) dans les couches supérieures	Fourni, généralement
Possibilité de négociation des options	Non	Oui
Utilisation d'identificateurs de connexion (NVL)	Non	Oui, dans <u>tous</u> les paquets

**Remarque** : La couche Réseau propose les deux types de service à travers 2 protocoles :

- PLP (X25.3),
- CNLP.

## 4.6. Les paquets errants

Certains paquets peuvent **errer** dans le réseau :

- notamment si le service Réseau est en mode datagramme
  - les paquets suivent des chemins indépendants
  - le mécanisme de retransmission produit des doubles
  - certaines portions du réseau peuvent se trouver isolées du reste :
    - . soit par panne de routeur ou par rupture de liaisons
    - . soit à cause de l'instabilité (vitesse de convergence) des algorithmes de routage
  - les routeurs doivent mémoriser les paquets qu'ils ne peuvent temporairement acheminer

Dans ce cas, la **durée de vie** des paquets est volontairement limitée.

Exemple IP :

- chaque datagramme est muni d'un champ spécifique : TTL ("Time to live"),
- initialisé à une certaine valeur : sa durée de vie maximale,
- décrémenté toutes les secondes de résidence dans un routeur et à chaque réception,
- le datagramme est détruit lorsque la valeur devient nulle.

## 4.7. Les algorithmes de routage

Mise à jour des tables de routage!

Deux classes d'algorithmes de routage :

- algorithmes de routage **fixes** (routage statique)
- algorithmes de routage **adaptatifs** (routage dynamique) :
  - . s'adapte aux modifications de la topologie et, plus rarement, à celles du trafic
  - . automatiquement

Les algorithmes de routage peuvent être :

- **centralisés** :  
les tables de routage de tous les noeuds sont calculées dans un seul noeud.
- **répartis** :  
les tables de routage sont calculées dans chaque noeud.

Les algorithmes de routage peuvent utiliser des informations :

- **locales** aux noeuds. Exemple : longueur de ses files d'attente.
- **collectées** globalement. Exemple : connaissance de la topologie totale

#### 4.7.1 Propriétés principales d'un algorithme de routage

- L'exactitude
- La robustesse
- La stabilité
- L'équité
- L'optimalité
- La complexité
  - Un bon algorithme de routage doit effectuer le meilleur compromis entre ces différents critères.

Les chemins proposés par le routage doivent :

- répondre aux exigences de qualité de chacune des **applications** (débit, délai, taux de perte, etc.)
- et simultanément offrir une utilisation optimale des ressources du **réseau**.
  - . critères contradictoires, apparemment!



## 4.8. Procédé de calcul des routes

### 4.8.1 Le routage multichemin (ou multiple)

La notion du plus “court” chemin dépend du **critère** d’évaluation (appelé métrique) choisi :

- distance, nombre de noeuds intermédiaires, débit offert, taux d’erreur, délai de transfert, coût financier,...
- plusieurs instances de la table de routage peuvent être nécessaires pour tenir compte de tous ces critères.

Plusieurs chemins peuvent mener au même destinataire :

- Chaque entrée de la table de routage d’un noeud peut comporter une liste de chemins possibles (triées par valeurs décroissantes).

### 4.8.2 Algorithmes centralisés de **calcul** du plus court chemin

- algorithme sur les graphes valués positivement [Dijkstra (1959), Moore (1957)],
- algorithme sur les graphes quelconques [Bellman (1958), Ford (1956)],
- algorithme matriciel [Floyd],
- etc.

### 4.8.3 Quelques procédés simples de routage

N'utilisent pas de table de routage : ne nécessitent aucune ou peu de connaissance locale et topologique

#### L'inondation

- simple, robuste, rapide (délai minimum),...
- ... mais risques de congestion et de copies multiples.
- exemple : “source routing” pour découvrir un chemin

#### Routage aléatoire

- sélection aléatoire du (*ou des*) prochains routeurs (*inondation sélective*)  
(tous les chemins mènent à Rome!)
- simple,... mais performances variables.

#### Le “hot potatoe” [Baran, 1964]

- sélection du prochain routeur en fonction de critères locaux :
  - . exemple : la longueur des files d'attente.
  - . optimisation locale (pas d'optimisation sur la totalité du chemin).

### Le “backward learning”

- apprentissage à travers des informations issues des messages de données précédents.
- durée de vie limitée des informations : nettoyage périodique nécessaire.
- exemple : “transparent bridging” et sa table de pontage
  - . utilisation de l’adresse source : telle station est accessible par ce port

#### 4.8.4 “Distance vector algorithm”

Procédé de routage adaptatif et réparti

Echange d’informations **synthétiques entre voisins**.

Principe :

- chaque noeud connaît la distance vers tous ses voisins immédiats,
- chaque noeud échange régulièrement avec tous ses voisins ses meilleurs chemins
- à chaque échange le noeud apprend de nouveaux chemins ou retient de meilleurs chemins.

Exemple :

- RIP (“Routing information protocol”) d’Internet

Simple, d’étendue limitée, mono-critère.

#### 4.8.5 “Link state algorithm”

Procédé de routage adaptatif et réparti

Echange l'état de **toutes les liaisons entre tous les routeurs**.

Principe :

- Diffusion de l'état de toutes les liaisons connues à tous les routeurs :
  - . tous les routeurs ont une même vision de la topologie du réseau -> même calcul
- Calcul local des meilleurs chemins entre le routeur et toutes les destinations :
  - . algorithme du “Shortest path first”

Exemple :

- OSPF : Open Shortest Path First (Internet), IS-IS (ISO)

## 4.8.6 Routage hiérarchique

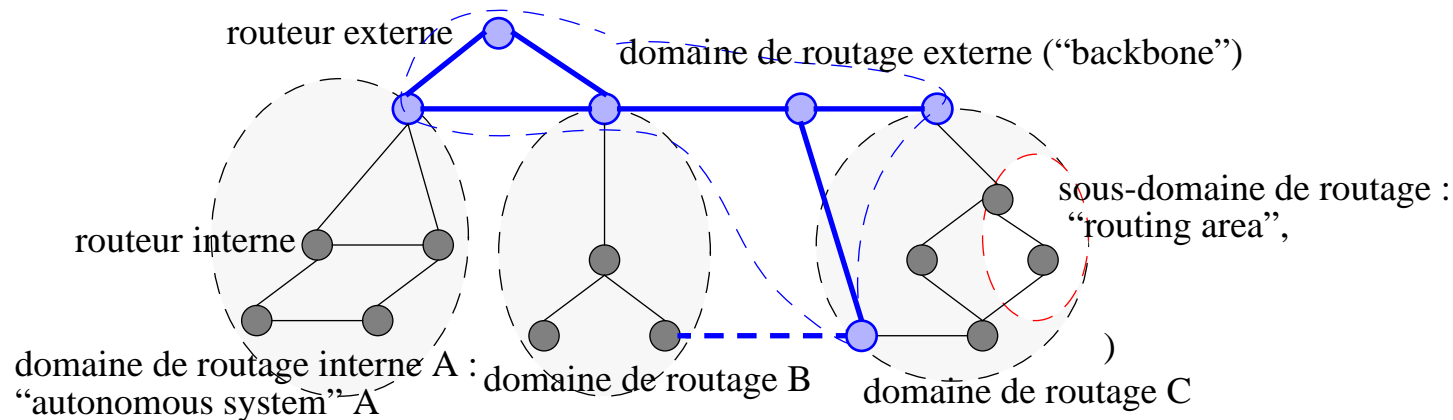
Lorsque la taille du réseau augmente :

- la taille des tables de routage s'accroît
- la complexité de calcul de la table de routage s'accroît
- le nombre de messages de routage échangés s'accroît



Le réseau est structuré hiérarchiquement en différents domaines (et sous-domaines) de routage :

- un nombre restreint de routeurs de chaque domaine assurent l'interconnexion entre domaines



## 5. Le contrôle de congestion

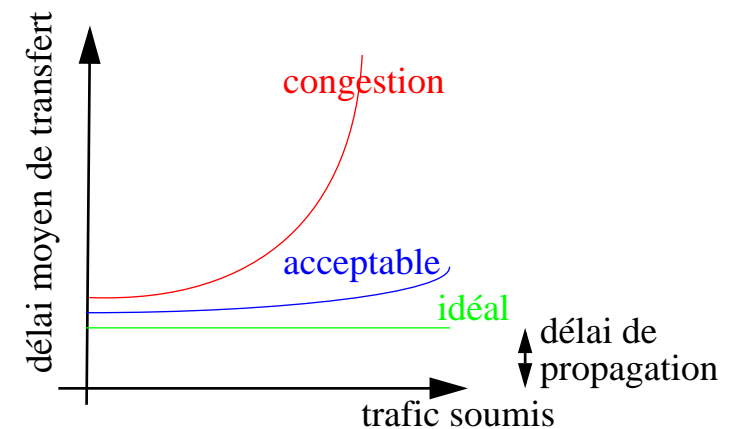
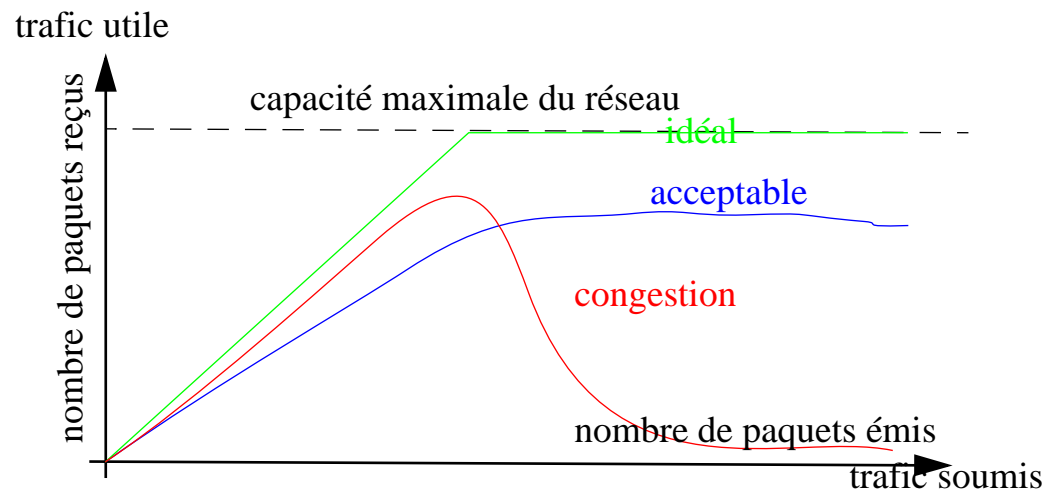
### 5.1. Introduction

La **congestion**, c'est le phénomène que l'on observe lorsqu'il y a trop de paquets présents dans le réseau.

Accumulation de paquets en attente d'émission dans les noeuds du réseau,

==> Retransmission des paquets : considérés à tort ou non comme perdus!

==> Augmentation du trafic et dégradation des performances.



Deux types de solution : la **prévention** (ou anticipation) ou la **régulation** (a posteriori).

**Attention** : le problème de contrôle de congestion est à distinguer du problème de contrôle de flux.

## 5.2. La préallocation des ressources

- Réserve des ressources : l'espace de stockage, portion de bande passante
  - Utilisable uniquement dans le cadre d'une connexion.
  - Efficace mais monopolisation inutile de ressources.

## 5.3. Le contrôle de congestion isarithmique

- Conservation d'un nombre constant de paquets en circulation dans le réseau : par utilisation de **jetons**.
  - Mauvaise répartition des jetons en fonction de la charge.

## 5.4. Rétro-contrôle

- Contrôle du débit des émetteurs
  - chaque noeud surveille le taux d'utilisation de ses liaisons
    - . explicitement : envoi de **paquet d'engorgement** vers les noeuds émetteurs de paquets devant être acheminés sur les liaisons chargées.
    - . implicitement : l'absence d'acquittement témoigne de congestion
  - délai de réaction

### 5.5. Priorité/destruction des paquets

- Sauf pré-réservation massive et donc inefficace, les congestions sont inévitables
  - nécessité de définir des **critères** pour choisir les paquets à détruire ou à acheminer en priorité.
  - inévitable mais en dernier ressort !



## 6. Etude du protocole PLP

### 6.1. Présentation

“Packet Layer Protocol”

#### 6.1.1 Normalisation (1974)

En mode connecté.

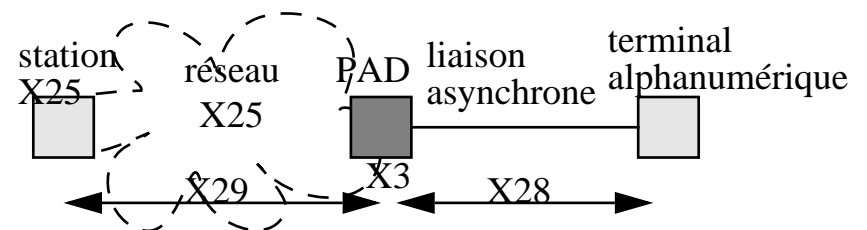
- **protocole** normalisé par le CCITT X.25 (ISO 8208)
- “Network **service** definition” X.213 (ISO 8348)

Mais il existe une variante en mode non connecté :

- CNLP : “Connectionless-mode network protocol” et service (ISO 8473, 9068)
- similaire à IP.

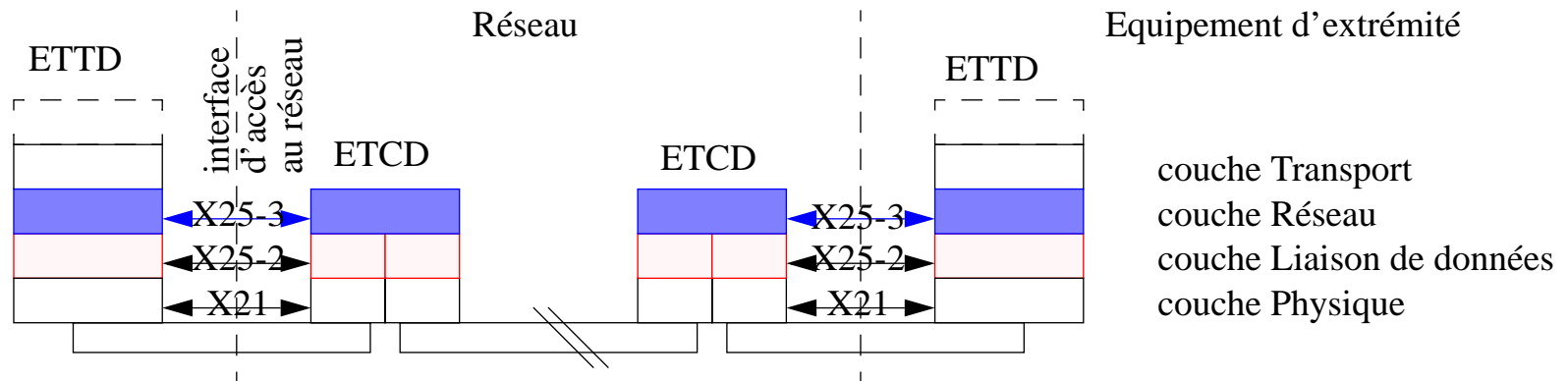
Adaptation de X.25 pour la transmission de caractères alphanumériques (historique) :

- assure la compatibilité
- PAD : “Packet assembler/desassembler”
- normes CCITT X.3, X.28, X.29



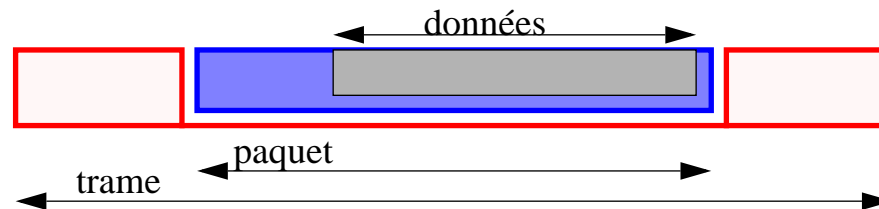
La norme X25 définit les protocoles des 3 premiers niveaux :

- X25.1 : Physique (X.21 ou X.21bis)
- X25.2 : Liaison de données (HDLC : LAP-B)
- X25.3 (PLP)



Encapsulation :

- `trame(paquet(données))`



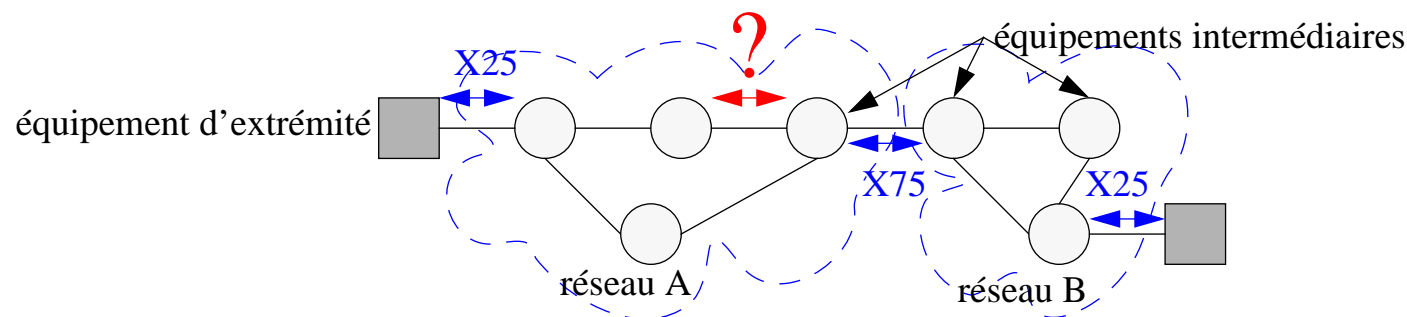
## 6.1.2 Fonctions

Le protocole X25.3, c'est un **protocole d'accès** au réseau :

- Entre un équipement d'extrémité (système hôte) et un équipement d'interconnexion (systèmes intermédiaires - routeurs).
- Il n'est pas forcément utilisé comme protocole d'échange des données au sein du réseau (entre équipements d'interconnexion).
  - . Ce serait imperceptible aux utilisateurs et donc c'est hors du champ de la normalisation.
  - . Mais par économie d'échelle, il est très souvent utilisé à l'intérieur du réseau.
  - . Exemple : Transpac.

En fait le même problème se pose entre réseaux de deux opérateurs :

- Définition d'un protocole d'interconnexion : X.75 (X.25 like)



### 6.1.3 Les circuits virtuels

Deux types de connexions possibles :

**Circuit virtuel permanent** (PVC : “Permanent Virtual Circuit”) :

- le circuit virtuel est toujours disponible : fonctionne comme une ligne spécialisée (louée/réservée). La connexion est établie en permanence!
- l'équipement d'extrémité n'a pas à implémenter les phases d'établissement et de rupture de la connexion (c-à-d le processus de signalisation ou de commande) :
  - . facilite la transition
  - . gain de temps
  - . sécurité ?

**Circuit virtuel** (VC : “Virtual Circuit”) :

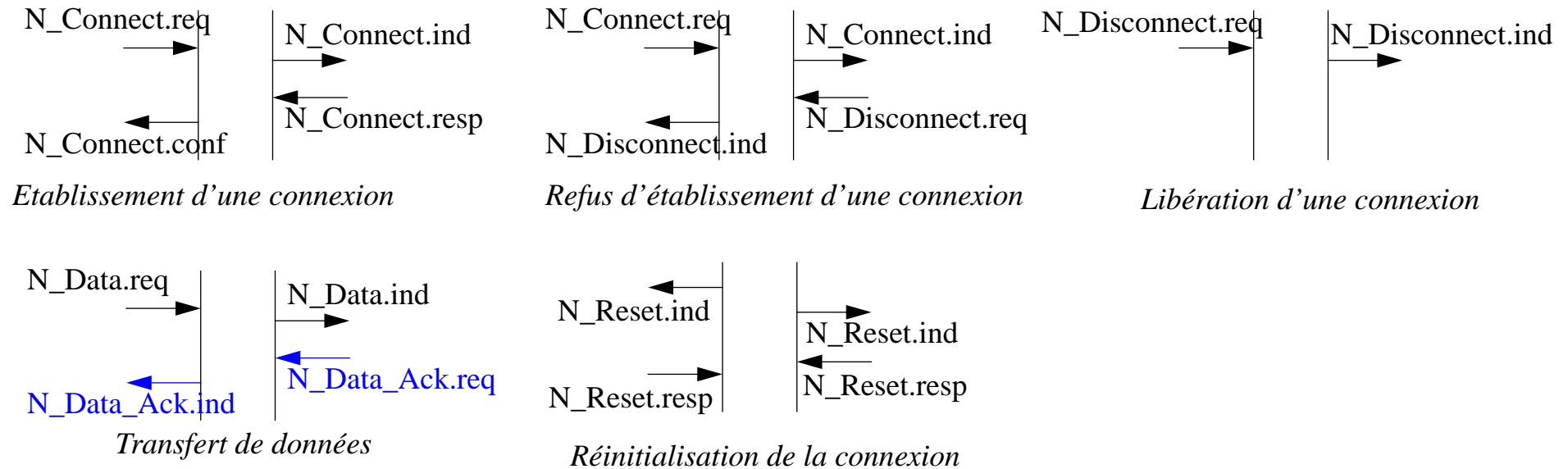
- c'est le fonctionnement normal
- en trois phases : établissement de la connexion, transfert de données, libération de la connexion.

## 6.2. Le service

### Mode connecté :

```
N_Connect.req(@ appelé, @ appelant, opt_conf_recept, opt_données_exprès, qos, données)
N_Connect.ind(@ appelé, @ appelant, opt_conf_recept, opt_données_exprès, qos, données)
N_Connect.resp(@ répondeur, opt_conf_recept, opt_données_exprès, qos, données)
N_Connect.conf(@ répondeur, opt_conf_recept, opt_données_exprès, qos, données)
N_Disconnect.req(raison, données, @ répondeur)
N_Disconnect.ind(origine, raison, données, @ répondeur)
N_Data.req(données, conf_demandée)
N_Data.ind(données, conf_demandée)
N_Data_ack.req()
N_Data_ack.ind()
N_Exp_data.req(données)
N_Exp_data.ind(données)
N_Reset.req(raison)
N_Reset.ind(origine, raison)
N_Reset.resp()
N_Reset.conf()
```

## Quelques enchaînements de primitives



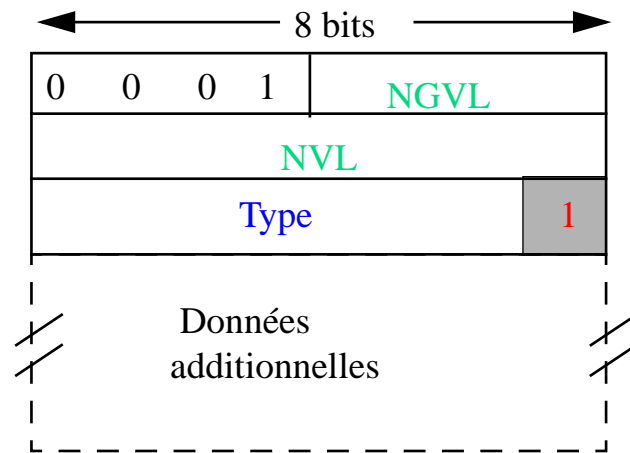
## Mode non connecté

`N_UniData.req(@ appelé, @ appelant, qos, données)`  
`N_UniData.ind(@ appelé, @ appelant, qos, données)`  
`N_Facility.req(qos)`  
`N_Facility.ind(origine, qos, raison)`  
`N_Report.ind(origine, qos, raison)`

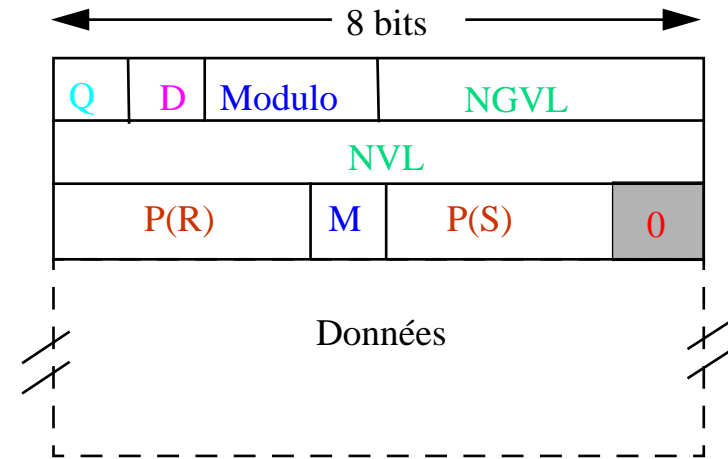
## 6.3. Les différents types de paquets

### 6.3.1 Le format général des paquets

- Paquets de données
- Paquets de contrôle



*Format général des paquets de contrôle*



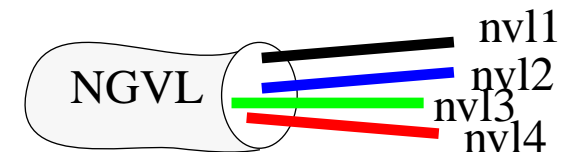
*Format général des paquets de données*

Identification hiérarchique du circuit virtuel :

- NVL : numéro de voie logique (= identificateur de tronçon de circuit virtuel)
- NGVL : numéro de groupe de voies logiques



brassage : aggrégation de plusieurs CV au sein d'un groupe de CV. Notion présente dans ATM (VP et VC)



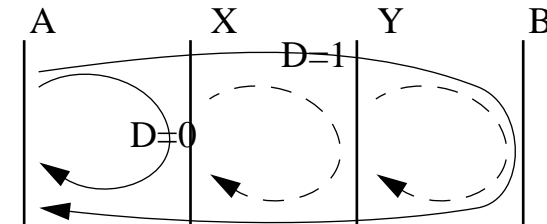
### 6.3.2 Le paquet de données

Le bit **Q** (“Qualified data”) :

- 1 : données de contrôle (ex: PAD)
- 0 : données issues des entités des couches supérieures

Le bit **D** (“Delivery”) :

- 1 : contrôle de flux de bout en bout
- 0 : contrôle de flux local (“back pressure”)



Le champ **Modulo** :

- dimension des champs de numérotation : P(S) et P(R),
- 01 : modulo 8 (entête de 3 octets) - format normal de paquet
- 11 : modulo 128 (entête de 4 octets) - format étendu de paquet

QD11	
P(S)	0
P(R)	M

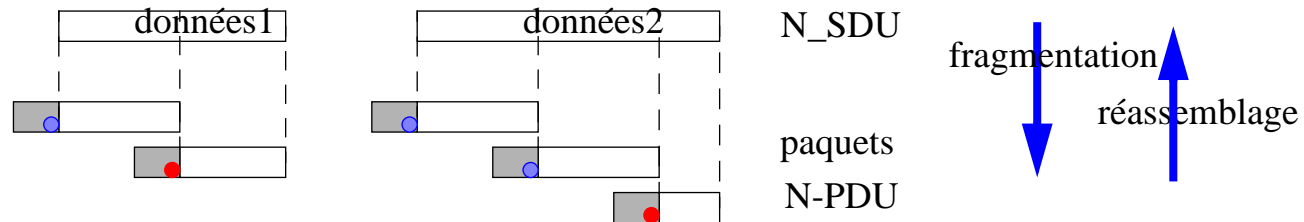
Les champs **P(S)** et **P(R)** :

- compteurs de la fenêtre coulissante utilisée par le protocole pour effectuer le contrôle de flux et le contrôle d'erreur
- P(S) : numéro du paquet (=N(S))
- P(R) : numéro du prochain paquet attendu et acquittement (=N(R))



Le bit **M** (“More”):

- **1** : ce paquet est le dernier fragment du N\_SDU
- **0** : ce paquet est un fragment intermédiaire du N\_SDU

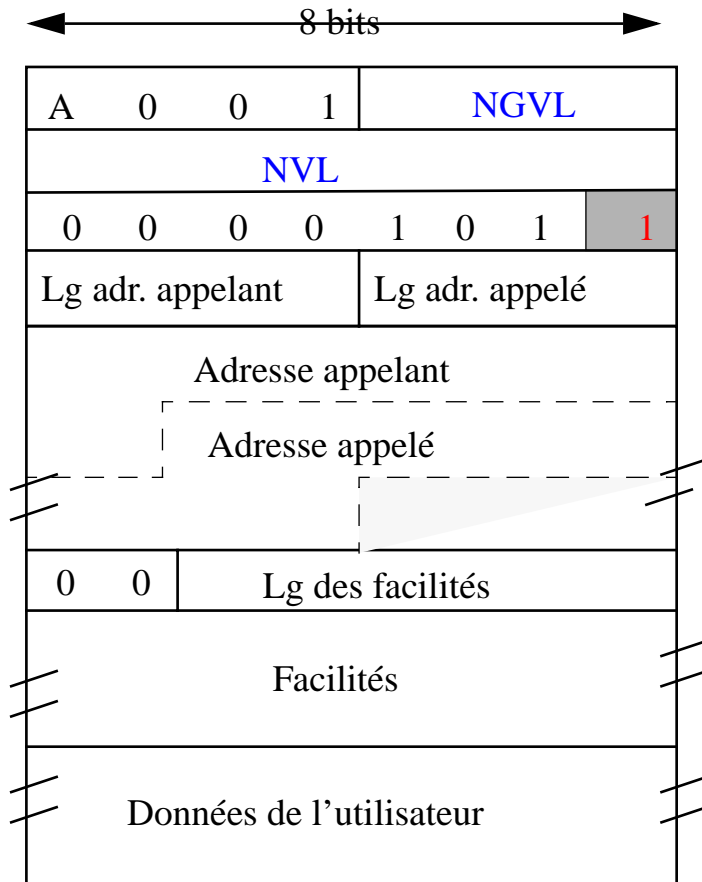


- La fragmentation est coûteuse : on préfère l’éviter et ne l’avoir qu’en frontière de réseau, si possible !

Le **champ de données** :

- longueur variable...
- mais bornée par la longueur maximale du paquet :
  - . négociée parmi les valeurs [16, 32, 64, 128, 256, 512, 1024, 2048, 4096],
  - . par défaut 128 octets.

### 6.3.3 Les paquets de contrôle



*Format du paquet d'appel (CALL REQUEST)*

Type de paquet	"type code"
CALL REQUEST (APP)	00001011
CALL ACCEPTED (COM)	00001111
RECEIVE READY (RR)	P(R)00001
RECEIVE NOT READY (RNR)	P(R)00101
REJECT (REJ)	P(R)01001
CLEAR REQUEST (LIB)	00010011
CLEAR CONFIRM (CONF)	00010111
INTERRUPT	00100011
INTERRUPT CONFIRM	00100111
RESET REQUEST	00011011
RESET CONFIRM	00011111
RESTART REQUEST	11111101
RESTART CONFIRM	11111111
DIAGNOSTIC	11110001

*Les paquets de contrôle*

## Le paquet d'appel

Le code correspondant au paquet d'appel : “00001011<sub>2</sub>”

Etablit le circuit virtuel dans chaque routeur :

- associe au sein de la table de CV en fonction de l'adresse de l'appelé et du lien à emprunter les NGVL+NVL entrant et sortant
- négocie les services complémentaires au travers des champs des facilités

Le bit **A** (“Address”) définit le type d'adresse utilisée :

- 1 : TOA+NPI address = “Type Of Address+Numbering Plan Identification+Address value”
- 0 : adresse X.121

Les **adresses** peuvent être de taille variable :

- le format peut s'adapter aux différents types d'adresse
- l'unité de longueur est le demi-octet
- les longueurs peuvent être différentes entre appelant et appelé !
- un sous-champ de bourrage, si nécessaire, permet l'alignement sur une frontière d'octet.

Le champ **Facilités** :

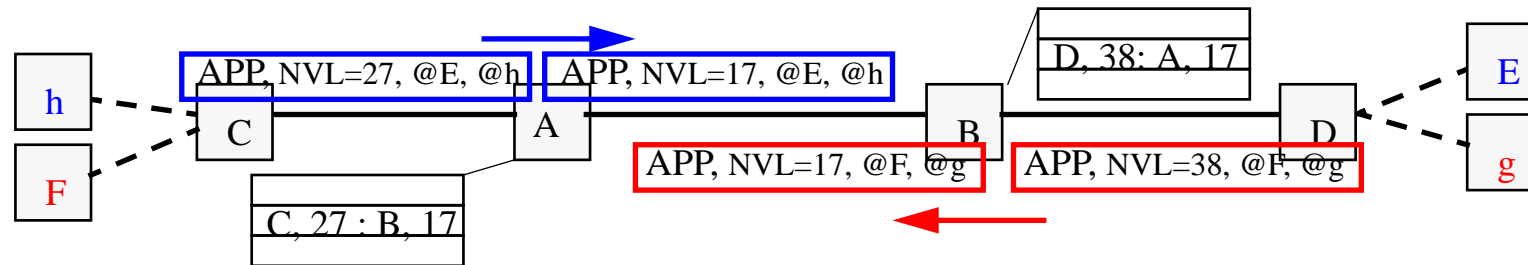
- c'est une liste d'options
- exemple d'options :
  - . taille maximum du paquet [16-4096 bits],
  - . largeur de la fenêtre [2-127 paquets],
  - . débit [75 à 64 Kbit/s], etc.

Le champ **Données** :

- données transmises lors de l'établissement de la connexion
- 10 octets par défaut, si "Fast select option" jusqu'à 128 octets

## Collision lors de la réservation des NVL

Si deux connexions passant par un même tronçon sont établies à partir de chacune des extrémités opposées simultanément alors les routeurs situés aux extrémités du tronçon partagé risquent de réserver le même numéro (NGVL+NVL) et les deux demandes d'établissement de connexion vont échouer.



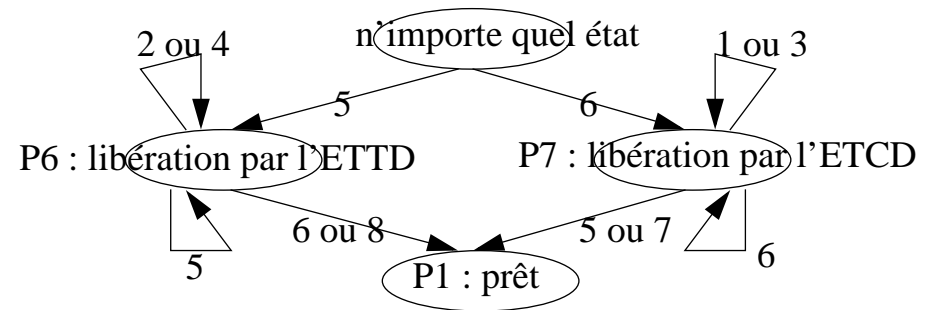
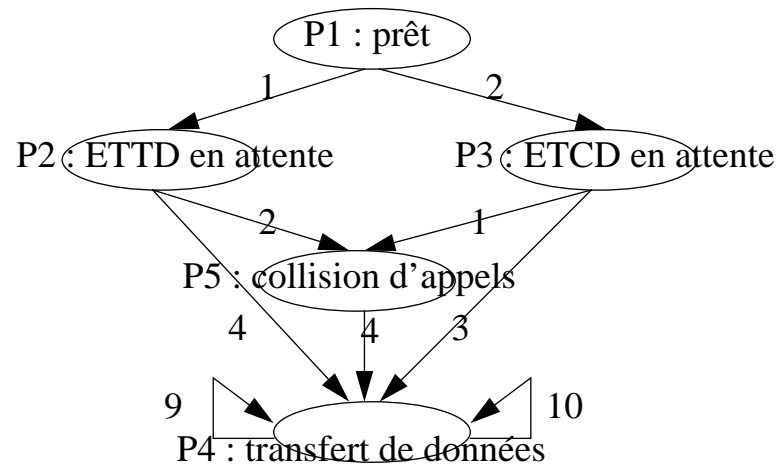
Remarque 1 : Le problème ne se pose que si le circuit virtuel est bidirectionnel et le même NVL est réservé pour les deux sens de transmission.

Remarque 2 : Au niveau des points d'accès au réseau pour diminuer la probabilité d'une telle collision, l'ETTD réserve les numéros à partir des valeurs basses, l'ETCD à partir des valeurs hautes.

## 6.4. Le protocole PLP

### 6.4.1 Spécification

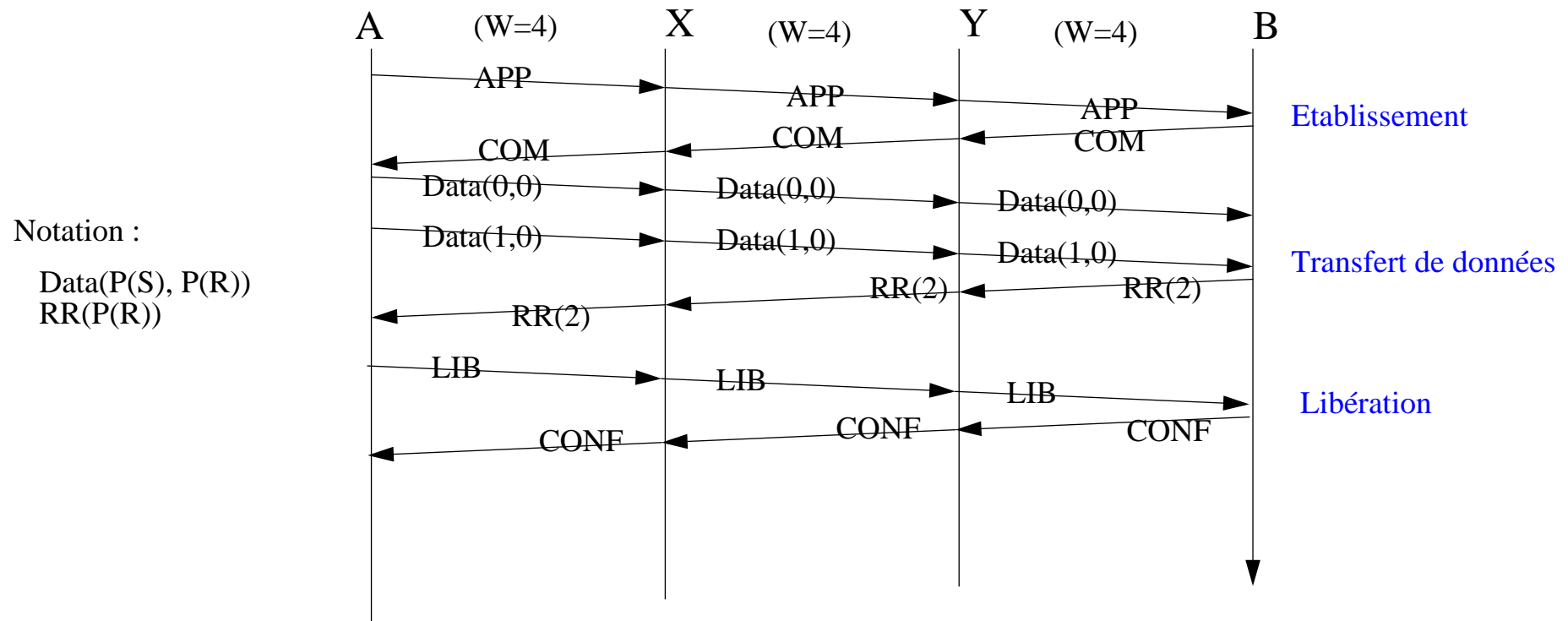
Extrait de l'automate décrivant le protocole :



- Paquet d'appel [APP] sortant (1: ETDD-> ETCD) ou entrant (2: ETCD->ETDD)
- Paquet de communication [COM] acceptée (3: ETDD-> ETCD) ou établie (4: ETCD->ETDD)
- Paquet de libération [LIB] demandée (5 : ETDD-> ETCD) ou indiquée (6: ETCD->ETDD)
- Paquet de confirmation de libération [CONF] (7 : ETDD-> ETCD) ou (8: ETCD->ETDD)
- Paquet de données [DATA] (9 : ETDD-> ETCD) ou (10: ETCD->ETDD)

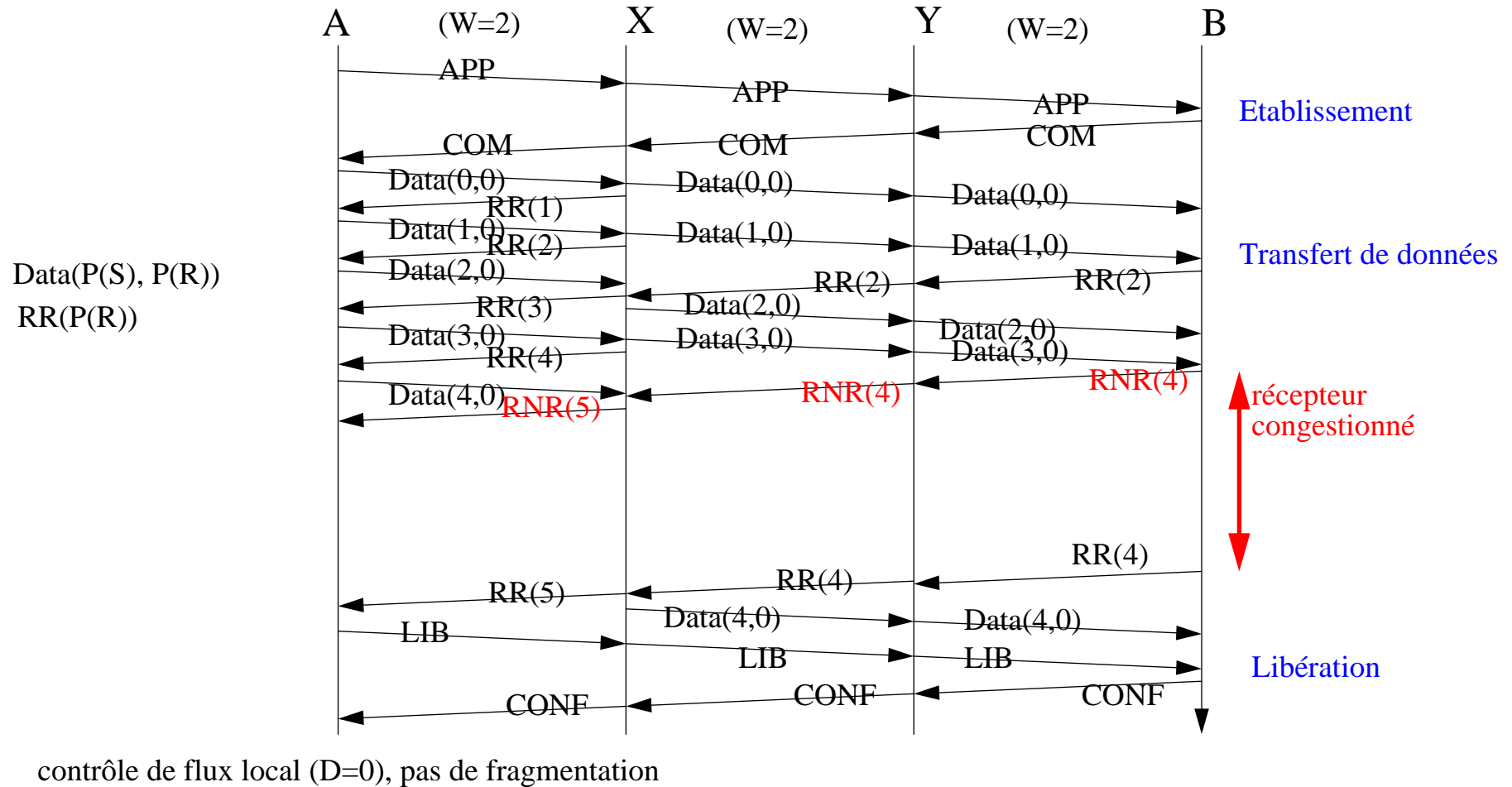
## 6.4.2 Exemples de scénarios

Etablissement de la connexion, transfert de données simple unidirectionnel puis libération de la connexion :



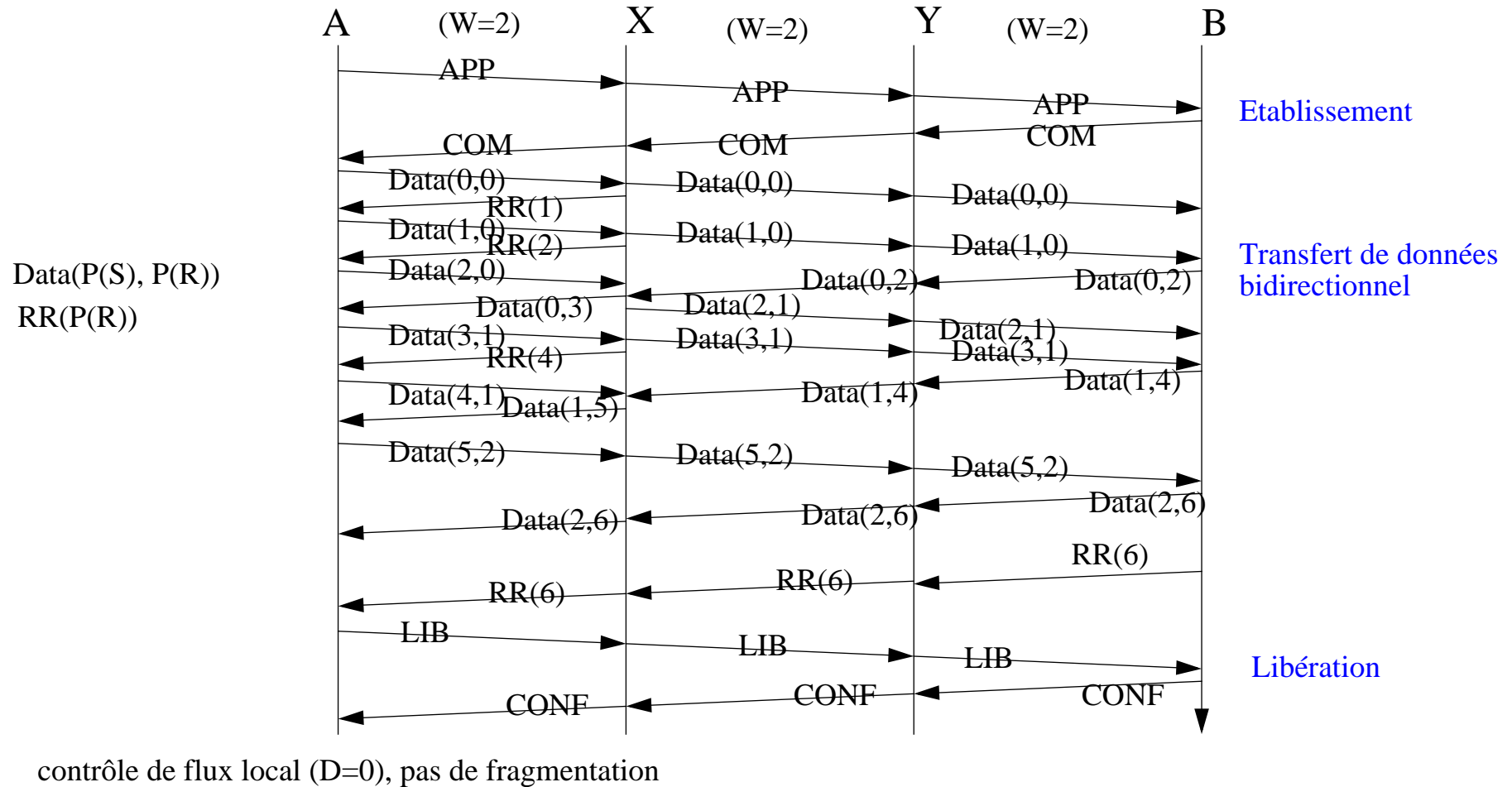
pas de fragmentation, contrôle de flux global (D=1)

## Transfert de données unidirectionnel avec congestion :





## Transfert de données bidirectionnel avec congestion :



## 7. Conclusion

Les fonctions essentielles de la couche Réseau sont :

- la **segmentation**, l'**adressage**, le **contrôle de flux** et le **routage**.

Le protocole **X25** est un protocole d'accès au réseau ...

- ... mais il peut être utilisé entre équipements intermédiaires.
- Il utilise une technique de commutation utilisant un **circuit virtuel** adapté au mode connecté.
- Il reprend le mécanisme de la **fenêtre coulissante** :
  - . contrôle de flux (et contrôle d'erreur).

D'autres services et protocoles existent :

- IP ou CNLP : leur service est non connecté, ils utilisent un procédé de transmission de type datagramme !

Les algorithmes de routage (mise à jour des tables de routage) sont déterminants pour de bonnes performances de transmission et une bonne utilisation du réseau.

La diffusion de données, la mobilité, la sécurité et le respect de la Qos sont les services qui devraient se développer dans le futur.